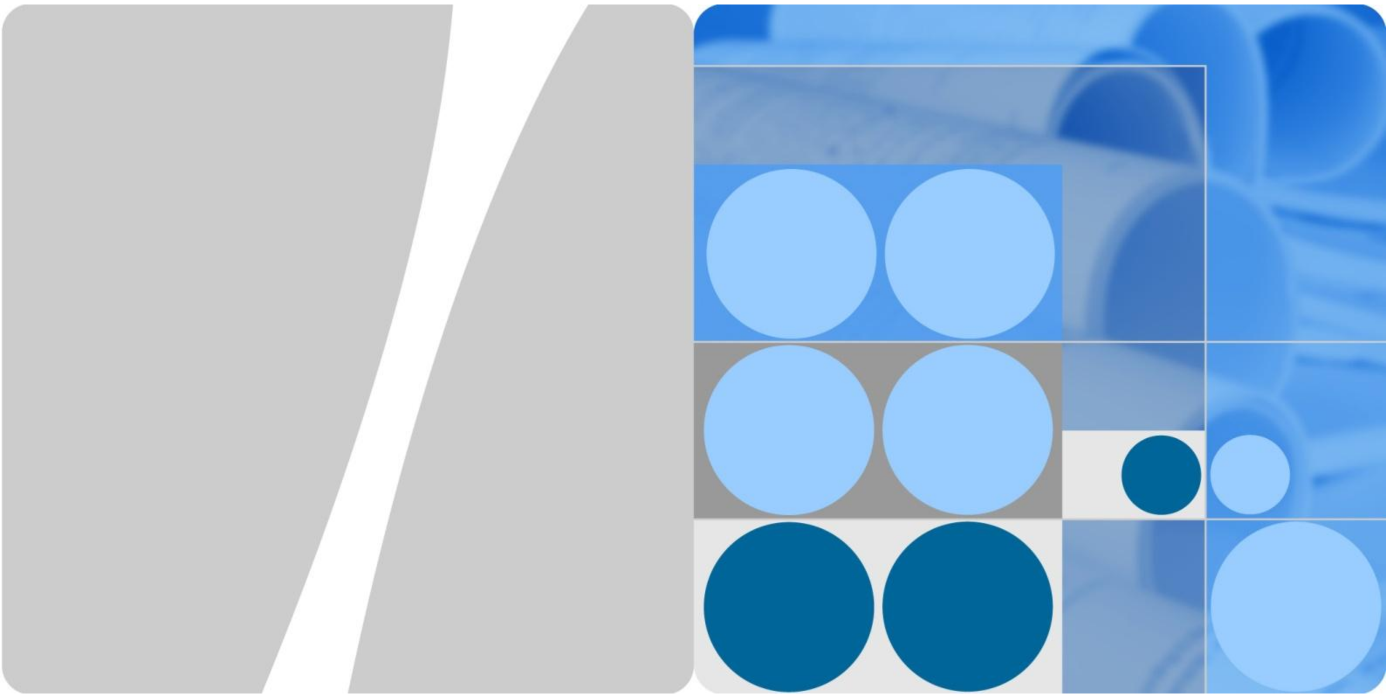


Huawei OceanStor 18500 and 18800 V5 Mission-Critical Hybrid Flash Storage Systems

Technical White Paper

Issue	01
Date	2018-07-31

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: http://e.huawei.com

Contents

1 Executive Summary

Huawei OceanStor 18000 V5 mission-critical hybrid flash storage series (OceanStor 18000 V5 series for short) are designed for enterprise-class applications.

This document describes and highlights the key technologies, unique advantages, and customer benefits of OceanStor 18000 V5 series in terms of product positioning, hardware architecture, software architecture, and features.

2 Overview

2.1 OceanStor 18000 V5 Mission-Critical Series

OceanStor 18000 V5 mission-critical series consists of OceanStor 18500 V5 and 18800 V5.

Figure 2-1 Exterior of OceanStor 18000 V5 mission-critical series

The specifications of OceanStor 18000 V5 mission-critical series are as follows:

- Each storage system provides a system bay that houses a 4-controller controller enclosure and a maximum of 16 x 2 U 25-slot 2.5-inch disk enclosures or 8 x 4 U 24-slot 3.5-inch disk enclosures as the standard configuration.
- Each system bay supports a maximum of 5 disk bays connected to it. Each disk bay can house a maximum of 16 x 2 U 25-slot 2.5-inch disk enclosures or 8 x 4 U 24-slot 3.5-inch disk enclosures.

- An OceanStor 18500 V5 storage system supports a maximum of 4 system bays, 12 disk bays, 16 controllers, and 6400 x 2.5-inch disks or 3072 x 3.5-inch disks.
- An OceanStor 18800 V5 storage system supports a maximum of 4 system bays, 20 disk bays, 16 controllers, and 9600 x 2.5-inch disks or 4608 x 3.5-inch disks.
- The system cache is integrated into controllers. Cache capacity can be non-disruptively expanded up to 16 TB (1 TB per controller) by adding controllers.
- Each storage system supports up to 384 front-end ports, including 8 Gbit/s Fibre Channel, 16 Gbit/s Fibre Channel, GE, 10GE, 10 Gbit/s FCoE, and 56 Gbit/s InfiniBand ports.
- The storage series uses the latest SAS 3.0 technology and supports a maximum of 192 x 4-lane 12 Gbit/s SAS 3.0 ports.

For detailed product specifications, visit:

http://e.huawei.com/en/products/cloud-computing-dc/storage/massive-storage/18500-18800-v5

2.2 Customer Benefits

OceanStor 18000 V5 mission-critical storage series is dedicated to providing the highest level of data services for enterprises' key services.

Leveraging a best-in-class SmartMatrix 2.0 system architecture, gateway-free HyperMetro active-active feature, flash-oriented optimization technology, cutting-edge hardware platform, and a full range of software used for efficiency improvement and data protection, OceanStor 18000 V5 series delivers world-leading reliability, performance, and solutions that meet the storage needs of various applications such as large-database Online Transaction Processing (OLTP), Online Analytical Processing (OLAP), and cloud computing. Applicable to sectors and industries such as government, finance, telecommunications, energy, transportation, and manufacturing, OceanStor 18000 V5 series is the best choice for mission-critical applications.

Converged: Accelerated Data Service Efficiency

Powered by the latest OceanStor OS, OceanStor 18000 V5 series provides converged and unified resource pools with the agility of resource scheduling, enabling free data mobility and helping enterprise IT architectures evolve to cloud-based architectures.

- Convergence of all types of flash storage

  Huawei has the most complete flash product portfolio and supports interconnection between different types, levels, and generations of flash storage. Convergence of data, management and O&M empowers high-performance (6 million IOPS) and low-latency (1 ms) flash storage arrays, while ensuring the long-term reliability of SSDs.
- Convergence of SAN and NAS

  SAN and NAS are converged to provide elastic storage, improve storage resource utilization, and reduce the total cost of ownership (TCO). OceanStor 18000 V5 series not only converges SAN and NAS to support multiple types of services, but also provides industry-leading SAN and NAS performance and functions.
- Convergence of storage resource pools

  The built-in heterogeneous virtualization function enables OceanStor 18000 V5 to take over the storage arrays of different levels, types, and models from other mainstream vendors, and integrate them into a unified resource pool. This eliminates data silos, achieves unified resource management, and enables automated service orchestration. In addition, data can be automatically migrated from third-party storage to Huawei storage without interrupting services. Huawei's automatic migration tool reduces the migration time by 60% on average.

- Convergence of multiple data centers

  The converged SAN and NAS active-active solution provides cross-data center service continuity assurance and makes the networking simpler. Active-active data center deployment can be smoothly upgraded to the geo-redundant 3DC layout to achieve the highest level of service continuity protection. Customers can also deploy hierarchical data centers for the purpose of centralized disaster recovery. Currently, Huawei storage supports the backup of data from 64 subordinate data centers to a central data center.

Stable and Reliable: 99.9999% High Availability from Products to Solutions

- 4-controller symmetric controller enclosure

  With the SmartMatrix architecture, OceanStor 18000 V5 series integrates four controllers into the 6 U space of a controller enclosure. The controllers are interconnected through a passive backplane. In addition, continuous cache mirroring and back-end disk controller interconnection techniques are incorporated, offering industry-leading 4-controller redundancy. The four controllers act as a hot backup for each other. Even if three controllers fail to work at the same time, service stability is protected to maximize the continuity of mission-critical applications, preventing a single-point running status that can be seen in scenarios where traditional high-end storage systems are upgraded or a controller is faulty.

- Load balancing

  Load balancing is implemented among controllers, thereby accelerating application access and eliminating performance bottlenecks.

- Full hardware redundancy

  All components and channels are redundant to prevent single points of failure. Fault detection, recovery, and isolation can be independently implemented for each component and channel, ensuring stable system running.

- Unique rapid data restoration technology

  Innovative block-level virtualization is employed to reduce the time needed to reconstruct 1 TB of data from 10 hours to 30 minutes. Compared with traditional storage systems, OceanStor 18000 V5 series reduces the risk of data damage caused by disk failures by 95%.

- DIX+PI end-to-end data protection

  Based on PI and DIX, OceanStor 18000 V5 series provides solutions to protect data integrity all the way from application systems to HBAs, storage systems, and disks. Such protection prevents damages to data, further protecting services.

- A wide range of data protection software

  The Hyper series of data protection software includes snapshot, clone, all-in-one backup, remote replication, and other data protection technologies. They protect user data locally, remotely, inside systems, and across different regions, and achieve 99.9999% availability, maximizing business continuity and data availability.

- Leading converged SAN and NAS active-active solution

  One storage array can support active-active deployment of both SAN and NAS, ensuring high availability for databases and file services. The gateway-free HyperMetro solution enables load balancing of active-active mirrors and non-disruptive cross-site takeover, ensuring zero loss of core application data and zero service interruption. Gateway-free design reduces customers' procurement spending and simplifies deployment. In addition, HyperMetro can be effortlessly upgraded to the geo-redundant layout with three data centers.

Fast: Outstanding Performance Achieved to Meet Requirements of Ever-Increasing Enterprise Services

- Flash-oriented storage architecture

  OceanStor 18000 V5 series employs a flash-oriented system architecture based on the flash convergence technology, CPU scheduling, cache, redundant array of independent disks (RAID), and interworking between the OceanStor OS and disk drives that are specially designed to suit flash memory. OceanStor 18000 V5 series can intelligently sense HDDs and SSDs, automatically distinguish between media types, and dynamically select the optimal algorithms to provide a stable I/O response time that is shorter than 1 ms in the event of massive service access requests, thereby ensuring the optimal performance of critical applications.

- Industry-leading performance and specifications

  OceanStor 18000 V5 series uses next-generation Intel multi-core processors, cutting-edge PCIe 3.0 buses, 12 Gbit/s SAS 3.0 high-speed disk ports, and a variety of host ports such as 16 Gbit/s Fibre Channel, and 10 Gbit/s FCoE as well as supports up to 384 front-end host ports. It fully meets requirements for bandwidth- and latency-sensitive applications.

- Flexible scalability

  OceanStor 18000 V5 series supports high-speed enterprise-class SSDs. A single storage system can be equipped with a maximum of 16 TB cache and 9600 disk drives, providing up to 55 PB of capacity, 6 million IOPS, as well as industry-leading performance and specifications.

3 System Architecture

3.1 Hardware Architecture

OceanStor 18000 V5 series uses the intelligent matrix multi-controller architecture. An OceanStor 18000 V5 can scale out by adding controller enclosures to achieve linear increase in both performance and capacity. A controller enclosure uses a four-controller redundancy architecture. The four controllers use the onboard PCIe3.0 to implement inter-controller cache mirroring. Multiple controller enclosures are scaled out through dedicated switches. Disks in a disk enclosure are connected to SAS 3.0 back-end full interconnection interface modules in the controller enclosure through two ports. In this way, all of the four controllers can access these disks. SSDs, SAS disks, and NL-SAS disks are supported. Backup battery units (BBUs) are used to ensure that data in the cache can be written to coffer disks in a timely manner when an OceanStor 18000 V5 encounters a power failure, protecting cache data and achieving zero data loss.

3.1.1 Back-End Full Interconnection

Each controller enclosure of a storage system contains four controllers that are interconnected through PCIe 3.0 high-speed channels provided by the passive backplane in the enclosure. When no more than four controllers are used, external cables and switches are not required, simplifying deployment and increasing reliability and performance. Figure 3-1 shows the front view of a 4-controller storage system, where four BBUs reside in the upper part, two controllers reside in the middle, and another two controllers reside in the lower part. Figure 3-2 shows the internal connections of a controller enclosure.

Figure 3-1 Front view of a 4-controller OceanStor 18000 V5 controller enclosure

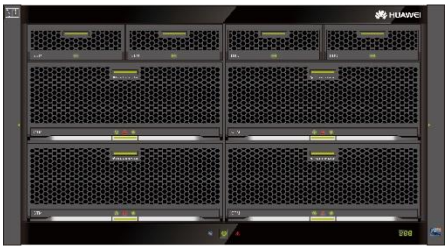
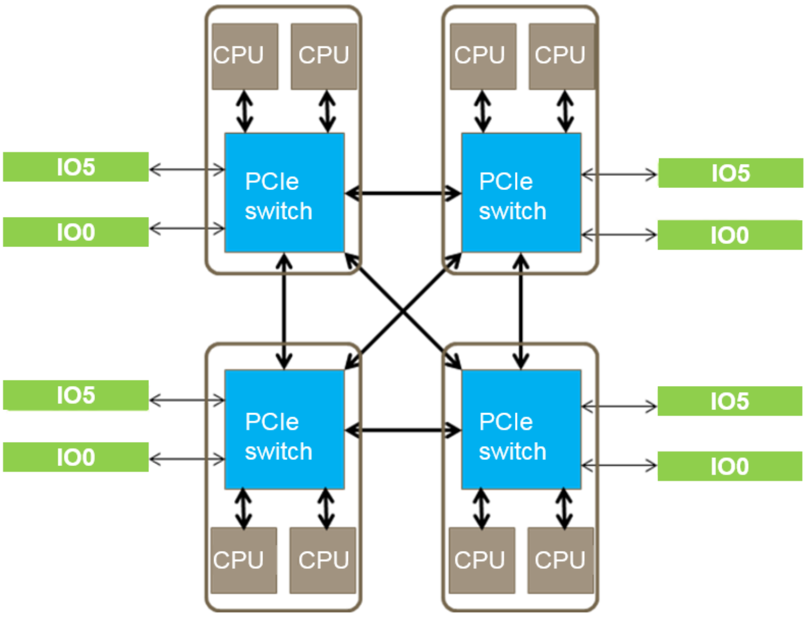


Figure 3-2 Internal connections of a controller enclosure



OceanStor 18000 V5 series uses SAS 3.0 back-end full-interconnection interface modules. Such an interface module is shared with and connected to the four controllers in a controller enclosure. If a controller is faulty, the connections between the controller enclosure and the disk enclosure remain redundant, not reducing the back-end connection reliability. Figure 3-3 shows the rear view of a controller enclosure, where the four slots in the middle house 4 U back-end full-interconnection interface modules. Figure 3-4 shows back-end connection of a storage system.

Figure 3-3 Rear view of a controller enclosure

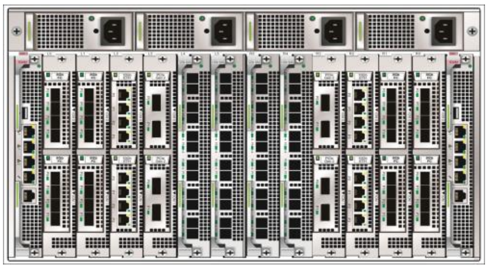
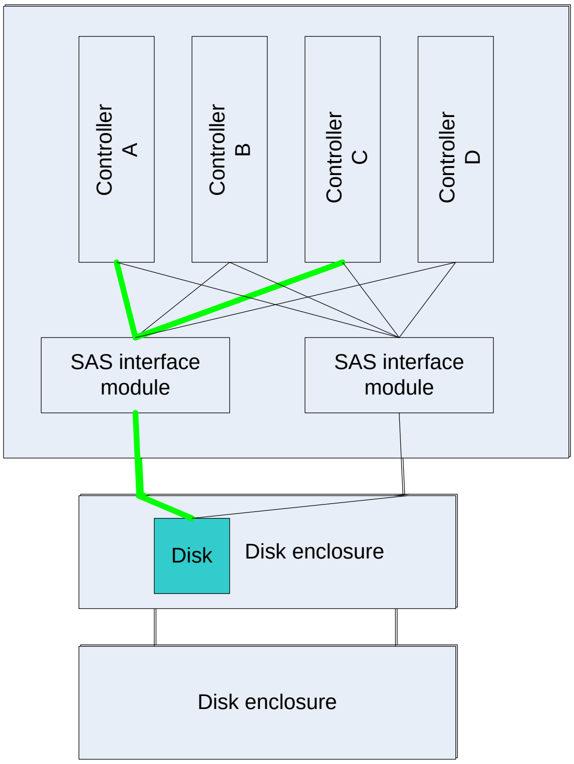


Figure 3-4 Back-end connections of a storage system

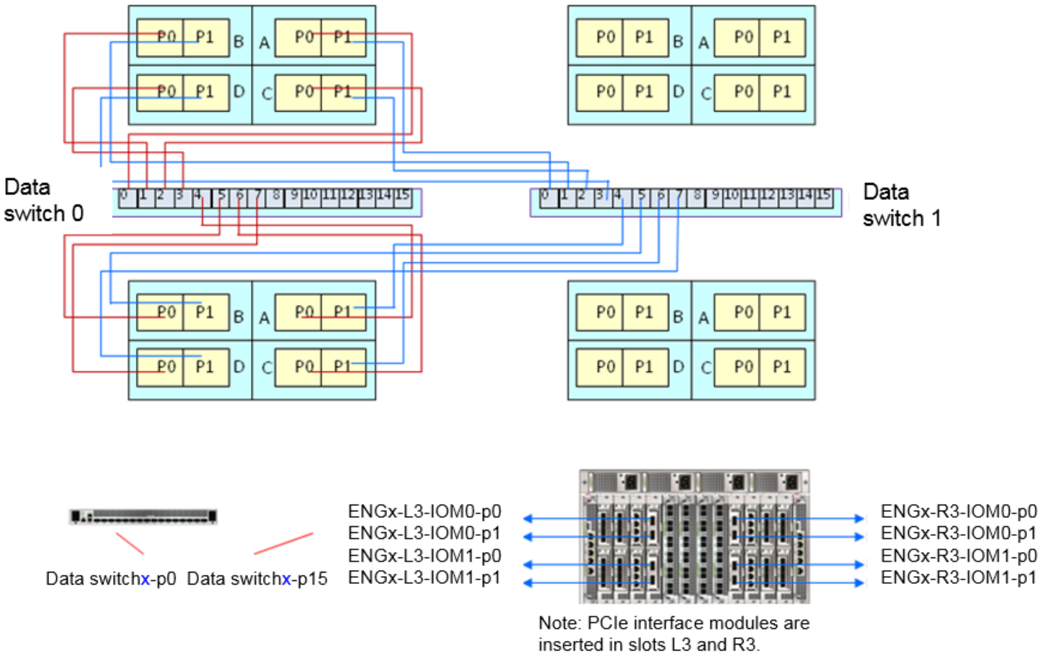


OceanStor 18000 V5 series supports persistent cache, a persistent mirroring technology. Typically, within a controller enclosure the cache of controller A and that of controller B act as a mirror for each other, as do that of controller C and controller D. In a traditional architecture, if a controller (say, controller B) becomes faulty, the cache of controller A enters the write through mode to ensure data reliability, leading to a significant decrease in performance. In the same scenario, however, persistent mirroring technology mirrors data from the cache of controller A to that of controller C or D (dynamically selected based on internal algorithms), ensuring cache data reliability and preventing significant performance deterioration.

3.1.2 PCIe Scale-out

OceanStor 18000 V5 series uses SmartMatrix 2.0, a multi-controller smart matrix architecture in which controllers exchange data through PCIe optical interconnection. A storage system supports a maximum of four controller enclosures. Each controller enclosure contains four controllers. The entire system supports a maximum of 16 controllers. Each controller is connected to two switching planes through PCIe optical cables for data forwarding. In addition, four controllers in a controller enclosure can forward data to each other through PCIe interconnection channels provided by the backplane on the controller enclosure. As shown in the following figure, slot 3 on each controller is equipped with a PCIe interface module. Each PCIe interface module has two PCIe optical ports, each of which is connected to a PCIe switch through an optical cable. (In the following figure, the cable connections of two controller enclosures are demonstrated.)

Figure 3-5 Scale out of an OceanStor mission-critical storage system



Note: PCIe interface modules are inserted in slots L3 and R3.

16 controllers are connected to two data switches through quad small form-factor pluggable (QSFP) cables to exchange data. The full-PCIe interconnection architecture reduces the latency caused by protocol conversion, accelerating data exchange. The full optical-interconnection design improves the reliability of long-distance data transmission and makes the distance between bays more flexible, facilitating the design of bay layout in an equipment room.

Each PCIe switch provides 16 PCIe switching ports, each of which delivers 8 GB/s of switching bandwidth. Two PCIe switches provide 256 GB/s (2 x 16 x 8 GB/s) of switching bandwidth in total. In each controller enclosure, 128 GB/s (PCIe 3.0 x 64) of backplane PCIe interconnection bandwidth is provided among four controllers. The total system switching bandwidth provided by a 16-controller mission-critical storage system is 768 GB/s (128 GB/s x 4 as the switching bandwidth within a controller enclosure + 256 GB/s as the cross–controller enclosure switching bandwidth).

NOTE

Controllers in each controller enclosure are connected by PCIe Gen3 x 8 links (that is, 8-lane PCIe 3.0 links).

The number of links connecting controllers is as follows:

- 2 for controllers A to B, and C to D
- 1 for controllers A to C, A to D, B to C, and B to D

A total of eight links are available between the four controllers, and each link provides a bidirectional bandwidth of 2 x 8 x 8 Gbit/s. Therefore, the total bidirectional PCIe interconnection bandwidth is 8 x 2 x 8 x 8 Gbit/s = 1024 Gbit/s, which is 128 GB/s.

3.1.3 Full Hardware Redundancy

All components and channels of the OceanStor 18000 V5 series are fully redundant, eliminating single points of failure. Components and channels can detect, repair, and isolate faults independently to ensure stable system running.

Table 3-1 Fully redundant hardware components

Hardware	Component	Redundancy	Fault Impact
Bay	PDU	1+1	No impact
Controller enclosure	Controller	1+3	Performance deteriorates accordingly.
	Power module	2+2	No impact
	Fan module	11+1	No impact
	BBU module	3+1	No impact
	Interface module	1+1	No impact
	Management module	1+1	No impact
Switch	Data switch	1+1	No impact
2 U disk enclosure	Expansion module	1+1	No impact
	Power module	1+1	No impact
	Fan module	1+1	No impact
4 U disk enclosure	Expansion module	1+1	No impact
	Power module	2+2	No impact
	Fan module	5+1	No impact

3.1.4 SED Data Encryption

OceanStor 18000 V5 series can work with self-encrypting drives (SEDs) and Internal Key Manager (built-in key management system) or External Key Manager (an independent key management system) to implement static data encryption. The data encryption feature uses the AES 256 algorithm to encrypt user data on storage to ensure the confidentiality, integrity, and availability of user data.

Internal Key Manager

Internal Key Manager is a key management application built in OceanStor 18000 V5 series. It uses the best practice design of NIST SP 800-57 to manage the authentication key (AK) life cycle of SEDs and supports the Trusted Platform Module (TPM) that meets FIPS 140-2 Level 2 requirements to protect keys. Internal Key Manager supports key generation, update, destruction, backup, and restoration.

Internal Key Manager is easy to deploy, configure, and manage. It is recommended if there is no requirement for FIPS 140-2 and key management is only used by storage systems in a data center. There is no need to deploy an independent key management system.

External Key Manager

OceanStor 18000 V5 series supports the External Key Manager (an independent key management system) that uses the Key Manager Server (KMS) of a third-party system to manage keys.

External Key Manager is a SafeNet Key Secure system that uses the standard KMIP + TLS protocols and complies with FIPS 140-2. It is recommended if FIPS 140-2 is required or multiple systems in a data center require centralized key management.

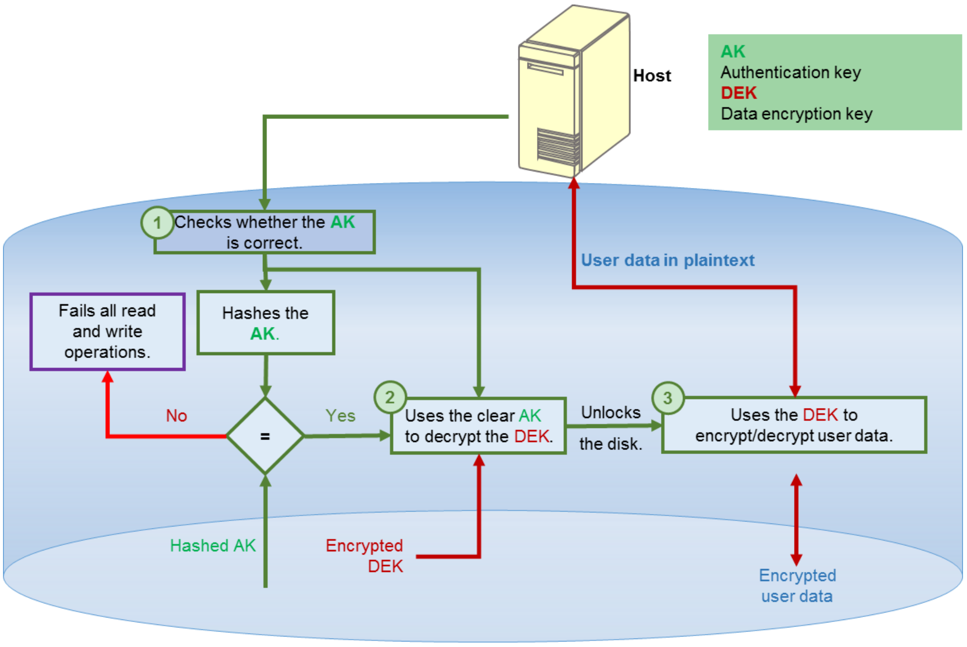
External Key Manager supports key generation, update, destruction, backup, and restoration. Two External Key Managers can be deployed to synchronize keys in real time for enhanced reliability.

SEDs

SEDs use AKs and data encryption keys (DEKs) to implement two layers of security protection.

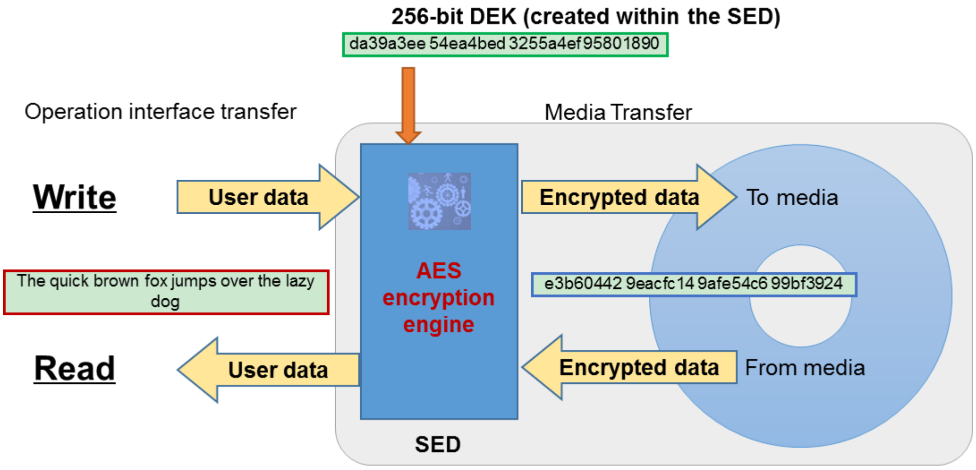
- AK mechanism: After data encryption has been enabled on a storage system, the storage system activates the AutoLock function for an SED, applies an AK from the key manager, and stores the AK on the SED. AutoLock protects the SED and allows only the storage system itself to access the SED. When the storage system accesses an SED, it acquires an AK from the key manager and compares it with the AK stored on the SED. If they are the same, the SED decrypts the DEK for data encryption or decryption. If they are different, all read and write operations will fail.

Figure 3-6 AK mechanism



- DEK mechanism: After the AutoLock authentication succeeds, the SED uses its hardware circuits and internal DEK to encrypt or decrypt the data that is written or read. DEK encrypts data after it is written to disks. The DEK cannot be acquired separately, that is, the original information on an SED cannot be recovered in a mechanical way after it is removed from the storage system.

Figure 3-7 Data encryption

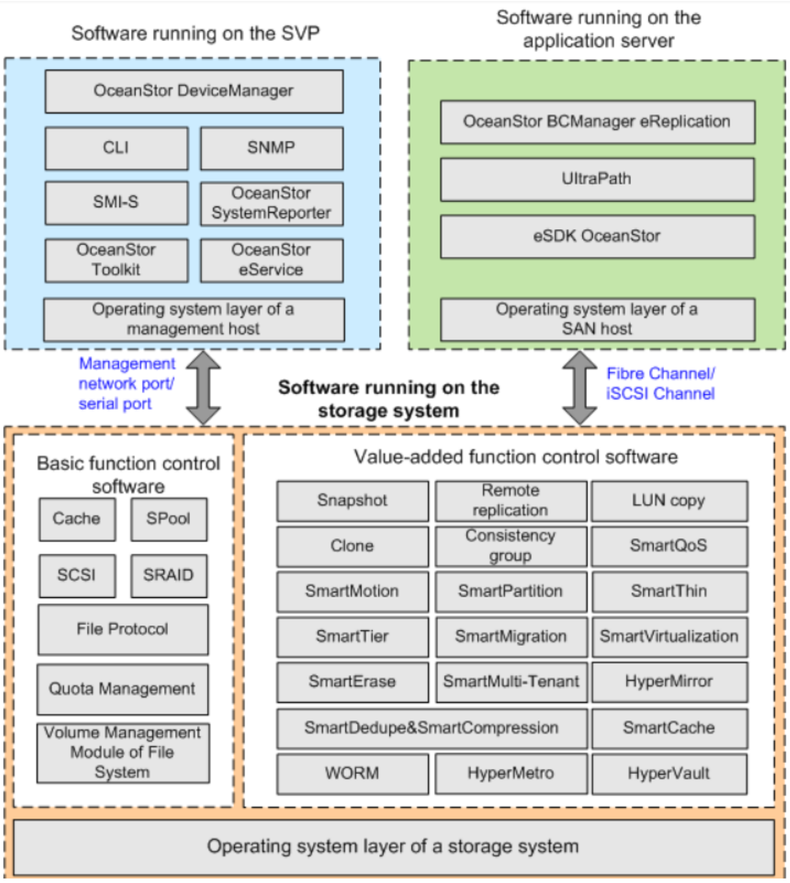

256-bit DEK (created within the SED)
da39a3ee 54ea4bed 3255a4ef 95801890
Operation interface transfer
Media Transfer
Write
User data
Encrypted data
To media
The quick brown fox jumps over the lazy dog
AES encryption engine
e3b60442 9eacfc14 9afe54c6 99bf3924
Read
User data
Encrypted data
From media
SED

3.2 Software Architecture

The software suite provided by the OceanStor 18000 series consists of software deployed on storage systems, management and maintenance software on the service processor (SVP), and software on application servers. These three types of software work jointly to deliver storage, backup, and disaster recovery services in a smart, efficient, and cost-effective manner.

Figure 3-8 shows the software architecture.

Figure 3-8 Software architecture


Software running on the SVP
OceanStor DeviceManager
CLI
SNMP
SMI-S
OceanStor SystemReporter
OceanStor Toolkit
OceanStor eService
Operating system layer of a management host
Software running on the application server
OceanStor BCManager eReplication
UltraPath
eSDK OceanStor
Operating system layer of a SAN host
Management network port/ serial port
Software running on the storage system
Fibre Channel/ iSCSI Channel
Basic function control software
Cache
SPool
SCSI
SRAID
File Protocol
Quota Management
Volume Management Module of File System
Value-added function control software
Snapshot
Remote replication
LUN copy
Clone
Consistency group
SmartQoS
SmartMotion
SmartPartition
SmartThin
SmartTier
SmartMigration
SmartVirtualization
SmartErase
SmartMulti-Tenant
HyperMirror
SmartDedupe&SmartCompression
SmartCache
WORM
HyperMetro
HyperVault
Operating system layer of a storage system

OceanStor 18000 V5 series uses the dedicated OceanStor OS operating system to manage hardware and support the running of storage system software. The basic function control software is used to provide basic data storage and access services. Value-added features are used to provide advanced functions, such as backup, disaster recovery, and performance optimization. Storage systems can be managed by management function control software.

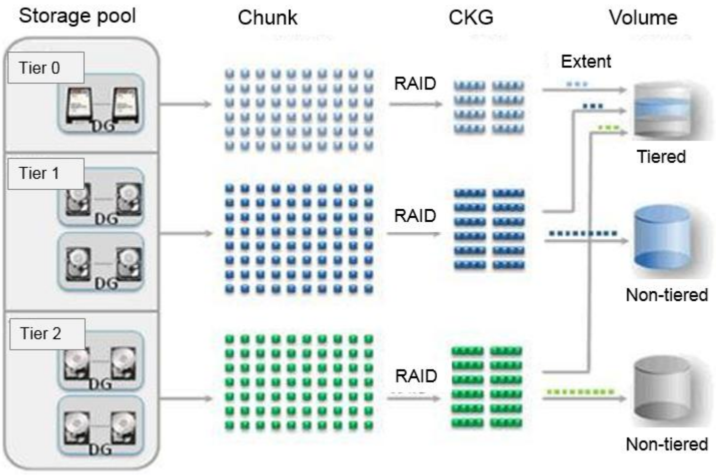
The following describes key technologies in terms of block-level virtualization, SAN and NAS integration, load balancing, data cache, end-to-end data integrity protection, and software features.

3.2.1 Block Virtualization

Working Principle

OceanStor 18000 V5 series uses the RAID 2.0+ block virtualization technology. Different from traditional RAID that has fixed member disks, RAID 2.0+ enables block virtualization of data on disks. All disks in a storage system are divided into fixed chunks (CKs) at a fixed size. Multiple chunks from disks are automatically selected at random to form a chunk group (CKG) based on the RAID algorithm. A CKG is further divided into extents at a fixed size. These extents are allocated to different volumes. Volumes are presented as LUNs or file systems. Figure 3-9 shows RAID 2.0+.

Figure 3-9 RAID 2.0+ block virtualization



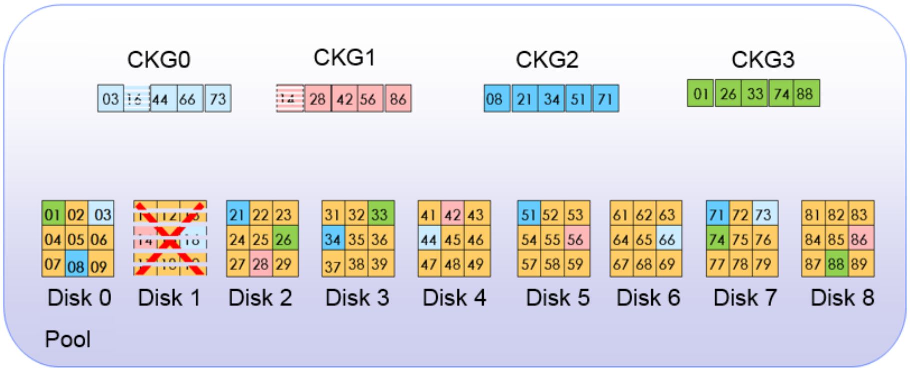
Fast Reconstruction

A RAID group consists of multiple chunks from several physical disks. Once a disk fails, other disks participate in data construction of the faulty disk. More disks are involved in data reconstruction and data reconstruction is accelerated. 1 TB data can be reconstructed within 30 minutes.

For example, in a RAID 5 group with nine member disks, if disk 1 becomes faulty, data in CKG0 and CKG1 is damaged. The storage system randomly selects chunks to reconstruct data on disk 1.

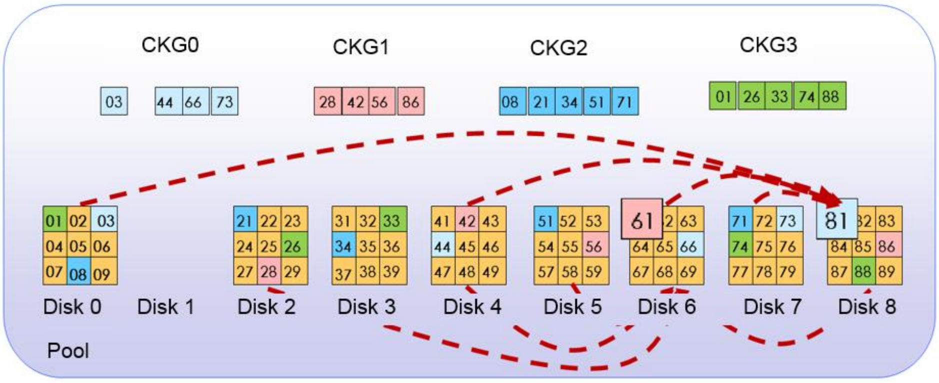
As shown in Figure 3-10, chunks 14 and 16 are damaged. In this case, idle chunks (yellow ones) are randomly selected from the pool to reconstruct data. The system attempts to select chunks from different disks.

Figure 3-10 RAID 2.0+ fast reconstruction (1)



As shown in Figure 3-11, chunk 61 on disk 6 and chunk 81 on disk 8 are selected randomly. Data will be reconstructed to the two chunks.

Figure 3-11 RAID 2.0+ fast reconstruction (2)



The bottleneck of traditional data reconstruction typically lies in the target disk (a hot spare disk). Data on all member disks is written to a target disk for reconstruction. As a result, the write bandwidth becomes the key factor that determines the reconstruction speed. For example, if 2 TB data on a disk is reconstructed and the write bandwidth is 30 MB/s, it will take 18 hours to complete data reconstruction.

RAID 2.0+ improves data reconstruction in the following two aspects:

1. Multiple target disks: In the preceding example, if two target disks are used, the reconstruction time will be shortened from 18 hours to 9 hours. If more chunks and member disks are involved, the number of target disks will be equal to that of member disks. As a result, the reconstruction speed linearly increases.
2. Chunk-specific reconstruction: If fewer chunks are allocated to a faulty disk, less data needs to be reconstructed, further accelerating reconstruction.

RAID 2.0+ shortens the reconstruction time per TB to 30 minutes, greatly reducing the probability of a dual-disk failure.

Load Balancing Among Disks

RAID 2.0+ automatically balances workloads on disks and evenly distributes data from volumes to all disks of a storage system, preventing individual disks from being overloaded and enhancing reliability. More disks participate in data reads and writes, improving storage system performance.

Maximized Disk Utilization

1. Performance

   In a RAID 2.0+ environment, LUNs or file systems are created using storage space from a storage resource pool and are no longer subject to the number of disks in a RAID group, greatly boosting the performance of a single LUN or file system.

2. Capacity

   The number of disks in a storage resource pool is not subject to the RAID level. This eliminates the possibility of usage difference of different RAID groups in traditional volume management environments. Coupled with dynamic LUN or file system capacity expansion, disk space usage is remarkably improved.

Enhanced Storage Management Efficiency

1. Easy planning

   It is not necessary to spend much time in planning storage. All that customers need to do is to create a storage pool using multiple disks, set the tiering policies of the storage pool, and allocate space (volumes) from the storage pool.

2. Easy expansion of storage pools

   To expand the capacity of a storage pool, customers only need to insert new disks, and the system will automatically distribute data evenly to all disks.

3. Easy expansion of volumes

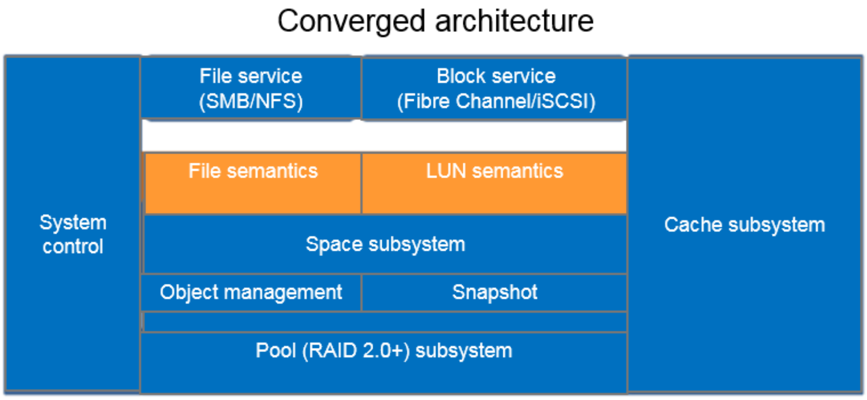
   When you need to expand the capacity of a volume, you only need to specify the size of the volume to be expanded. The system automatically allocates the required space from the storage pool and adjusts the data distribution of the volume to evenly distribute the volume data to all disks.

3.2.2 SAN and NAS Convergence

OceanStor 18000 series uses a SAN and NAS convergence design. NAS gateways are no longer needed. One set of hardware and software supports both SAN and NAS as well as file access protocols such as Network File System (NFS), Common Internet File System (CIFS), FTP, and HTTP, and file backup protocol Network Data Management Protocol (NDMP). Like SAN, NAS supports Scale-out of 16 controllers. Hosts can access any LUN or file system from a front-end host port on any controller.

Figure 3-12 shows the converged architecture of the storage systems. File systems and LUNs directly interact with the space subsystem. The file system architecture is based on objects. Each file or folder acts as an object, and each file system is an object set. LUNs are classified into thin LUNs and thick LUNs. The two types of LUNs come from the storage pool system and space system, instead of file systems. In this way, this converged architecture delivers a simplified software stack and provides a higher storage efficiency than the traditional unified storage architecture. In addition, LUNs and file systems are independent from each other.

Figure 3-12 OceanStor OS software architecture



3.2.3 Load Balancing

SAN Load Balancing

By default, OceanStor 18000 V5 series evenly allocates LUNs to controllers and evenly distributes LUN space to all disks in the system.

If there is an I/O path between a host and each controller of the storage system, UltraPath, Huawei proprietary multipathing software, preferably selects the path to the owning controller of the target LUN. If no optimum path is available, the system automatically determines the corresponding controller of the LUN service after I/O requests are delivered to the storage system. Then the Smart Matrix architecture transfers the I/O requests to the corresponding controller.

Even allocation of LUNs to controllers and distribution of LUN space among disks balance workloads of controllers and disks. SmartMatrix selects the optimum path to deliver I/O requests using UltraPath. In this way, the system can reach its optimum performance.

NAS Load Balancing

By default, OceanStor 18000 V5 series automatically allocates file systems to controllers. The file system space is evenly distributed to all disks in the system to balance the service loads and disk pressure.

OceanStor 18000 V5 series also provides the DNS load balancing feature to intelligently distribute host NFS/CIFS/FTP client connections to service IP addresses configured on different nodes and ports based on service loads, improving system performance and reliability.

When a host uses a domain name to access the NAS service of a storage system, the host sends a DNS request to the built-in DNS server of the storage system to obtain an IP address based on the domain name. If the domain name contains multiple IP addresses, the built-in DNS server selects an IP address with a light load to respond to the host based on the CPU usage, port bandwidth usage, and number of NAS connections of the controllers where IP addresses reside. After receiving the DNS response, the host sends a service request to the destination IP address.

The DNS load balancing feature supports the following load balancing policies: round robin, node CPU usage, node connection quantity, node bandwidth usage, and comprehensive node load.

3.2.4 Data Caching

- Cache distribution

  The physical memory usage of OceanStor 18000 V5 series is as follows:

  Physical memory = Cache occupied by the operating system + Read cache + Local write cache + Mirroring write cache + Cache occupied by service features

- Cache types

  There are two types of cache in OceanStor 18000 V5 series, namely, read cache and write cache.

  - Read cache: The data that has been read is saved in memory (read cache). These is no need to read the same data from disks again next time, accelerating read efficiency.
  - Write cache: The data that is about to be written into disks is saved in memory (write cache). When the amount of data that is saved in the write cache reaches a threshold,

the data will be saved to disks. Read cache and write cache reduce disk-related operations, improve read and write performance of storage systems, and protect disks from being damaged due to repeated read and write operations.

If the write cache is not used, all cache can be used as the read cache. Each storage system reserves the minimum read cache to ensure that read cache resources are still available even if the write workload is heavy.

- Cache prefetch

  In the event of a large number of random I/Os, OceanStor 18000 V5 series implements the multi-channel sequential I/O identification algorithm to identify sequential I/Os. For the sequential I/Os, the storage systems employ prefetch and merge algorithms to optimize system performance in various application scenarios.

  The prefetch algorithm supports intelligent prefetch, constant prefetch, and variable prefetch. By automatically identifying I/O characteristics, intelligent prefetch determines whether data is prefetched and determines the prefetch length, ensuring that the system performance meets requirements of different scenarios.

  By default, the storage systems adopt the intelligent prefetch algorithm. In application scenarios with definite I/O models, users can also configure constant prefetch or variable prefetch. These two algorithms allow users to define a prefetch length.

- Cache eviction

  When the cache usage reaches a threshold, the cache eviction algorithm calculates the access frequency of each data block based on historical and current data access frequencies, and works together with the multi-channel sequential I/O identification algorithm to evict unnecessary cached data. In addition, you can configure the cache priority of a volume and adjust the priority of each I/O for a specific service. Data with a low priority is eliminated first. High-priority data is cached to ensure the data hit rate.

3.2.5 End-to-End Data Integrity Protection

The ANSI T10 Protection Information (PI) standard provides a way to check data integrity during access to a storage system. The check is implemented based on the PI field defined in the T10 standard. This standard adds an 8-byte PI field to the end of each sector to check data integrity. In most cases, the T10 PI is used to ensure the integrity of data in a storage system.

Data Integrity Extensions (DIX) provided by vendors such as Oracle and Emulex further extends the protection scope of T10 PI from Oracle databases to HBAs. Therefore, DIX+T10 PI can achieve complete end-to-end data protection from applications to disks.

In addition to using T10 PI to ensure the integrity of data in a storage system, OceanStor 18000 V5 also adopts DIX + T10 PI to implement end-to-end data integration protection from Oracle databases to disks. A storage system verifies and delivers the PI fields of data in real time. If a host does not support PI, the storage system adds the PI fields to the host interface and delivers the fields. In a storage system, PI fields are forwarded, transmitted, and stored together with user data. Before user data is read by a host again, the storage system uses PI fields to check the accuracy and integrity of user data.

3.2.6 Various Software Features

OceanStor 18000 V5 series provides the Smart software series for accelerating system efficiency and the Hyper series software for protecting data.

- The Smart software series includes SmartDedupe, SmartCompression, SmartThin, SmartVirtualization, SmartMigration, SmartTier, SmartQoS, SmartPartition, SmartErase, SmartMulti-Tenant, SmartCache, SmartQuota, and SmartMotion. These software features help users improve storage efficiency and reduce the total cost of ownership (TCO).

- The Hyper software series includes HyperSnap, HyperClone, HyperReplication, HyperMetro, HyperVault, HyperCopy, HyperMirror, and HyperLock. These software features help users implement data backup and disaster recovery. In addition, the storage systems can be used in various disaster recovery solutions in which three data centers are deployed.

The storage systems can also be deployed in solutions that are integrated with common IT systems. The following is an example of integration with some VM environments.

- VMware vSphere

  The storage systems support VMware vStorage APIs for Array Integration (VAAI), vStorage APIs for Software Awareness (VASA), and Site Recovery Manager (SRM). The vCenter plugin is provided, enabling unified management in vCenter.

- Windows Hyper-V

  The storage systems support the Windows Thin space reclamation technology and Windows Offload Data Transfer (ODX). The System Center plug-in is provided and can be managed by System Center Operations Manager (SCOM) and System Center Virtual Machine Manager (SCVMM).

3.2.7 Flash-Oriented System Optimization

SSDs deliver high performance in random I/O accesses with a low latency but their erase times are limited. HDDs deliver high performance in sequential I/O accesses and their erase times are not restricted. Huawei has optimized SSDs and hybrid storage of SSDs and HDDs used in OceanStor 18000 V5 series to achieve better performance and reliability.

- Seamless collaboration between OceanStor OS and Huawei SSD (HSSD) firmware

  SSDs use flash chips that involve erasure operations. When erasure operations are being performed, other data in the channels involved in the erasure operations are inaccessible. As a result, a latency of 1 ms to 2 ms exists, leading to performance fluctuations.

  Huawei storage systems use HSSDs. OceanStor OS collaborates with SSDs to ensure that erasure operations are performed on multiple HSSDs in turn. OceanStor OS does not read data from the HSSDs where erasure is being performed. Instead, data is read from other HSSDs based on a RAID redundancy mechanism, thereby ensuring a stable latency.

- Intelligent SSD perception by cache

  The storage systems employ different dirty data flushing policies for SSDs and HDDs. When Huawei-certified disks are connected, the storage systems automatically identify the media types. For SSDs, the storage systems delay the flushing of active data, reduce the flushing times, and decreases write amplification based on the Least Recently Used (LRU) algorithm to boost system performance and prolong SSDs' service life.

- Performance optimized using multiple cores

  In terms of multi-core scheduling mechanism, system performance is optimized for the NUMA architecture. For example, messages related to a single I/O are dispatched to the same CPU to reduce cross-CPU access overheads and increase the CPU cache hit ratio.

  With regard to multi-thread operating efficiency, a proper data structure design is employed to prevent multiple threads from concurrently accessing data on a cache line of CPU L1 cache, thereby eliminating the pseudo-sharing of CPU L1 cache, improving CPU L1 cache efficiency, and reducing the CPU overhead in memory-based data access.

4 Smart Series Features

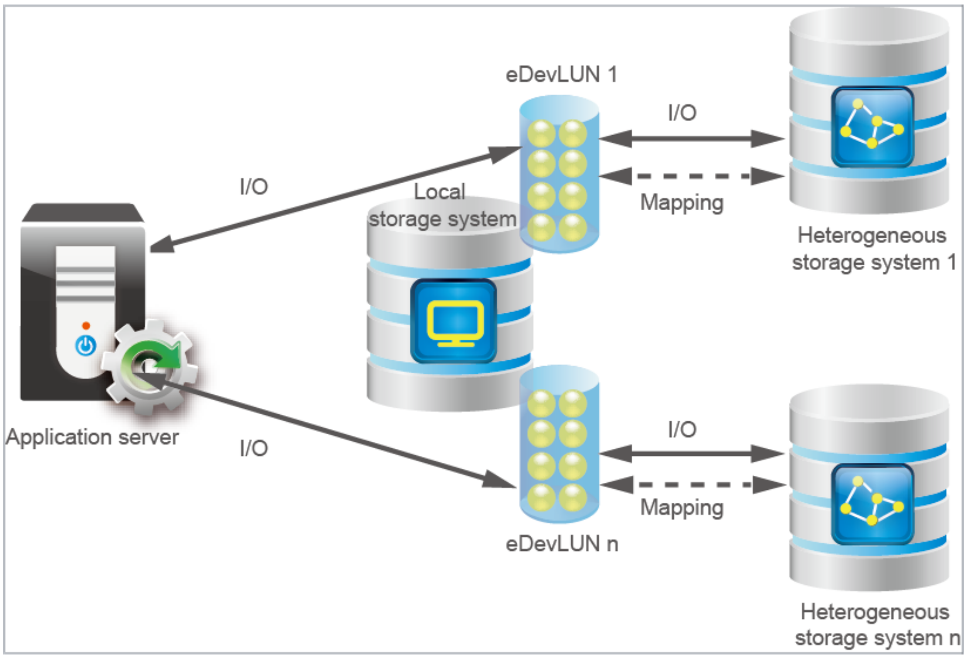
4.1 SmartVirtualiztaion

OceanStor 18000 V5 mission-critical series uses SmartVirtualization to take over heterogeneous storage systems (including other Huawei storage systems and third-party storage systems), protecting customer investments. SmartVirtualization conceals the software and hardware differences between the local and heterogeneous storage systems, allowing the local system to use and manage the heterogeneous storage resources as its local resources. In addition, SmartVirtualization can work with SmartMigration to migrate data from heterogeneous storage systems in online mode, facilitating device replacement.

Working Principle

SmartVirtualization maps the heterogeneous storage system to the local storage system which then uses external device LUNs (eDevLUNs) to take over and manage the heterogeneous resources. eDevLUNs consist of metadata volumes and data volumes. The metadata volumes manage the data storage locations of eDevLUNs and use physical space provided by the local storage system. The data volumes are logical presentations of external LUNs and use physical space provided by the heterogeneous storage system. An eDevLUN on the local storage system matches an external LUN on the heterogeneous storage system. Application servers access data on the external LUNs via the eDevLUNs.

Figure 4-1 Heterogeneous storage virtualization



SmartVirtualization uses LUN masquerading to set the world wide names (WWNs) and Host LUN IDs of eDevLUNs on a storage system to the same values as those on heterogeneous storage system. After data migration is complete, the host's multipathing software switches over the LUNs online without interrupting services.

Application Scenarios

- Heterogeneous array takeover

  As customers build data centers over time, the storage arrays they use may come from different vendors. Storage administrators can leverage SmartVirtualization to manage and configure existing devices, protecting investments.

- Heterogeneous data migration

  The customer may need to replace storage systems whose warranty periods are about to expire or whose performance does not meet service requirements. SmartVirtualization and SmartMigration can migrate customer data to OceanStor V5 mid-range series in online mode without interrupting host services.

- Heterogeneous disaster recovery

  If service data is stored at two sites having heterogeneous storage systems and robust service continuity is required, SmartVirtualization can work with HyperReplication to enable data on LUNs in heterogeneous storage systems to be backed up mutually. When a disaster occurs, a functional service site takes over services from the failed service site and recovers data.

- Heterogeneous data protection

Data on LUNs that reside in heterogeneous storage systems may be attacked by viruses or corrupted. SmartVirtualization can work with HyperSnap to instantly create snapshots for LUNs that reside in heterogeneous storage systems, and rapidly restores data at a specific point in time using the snapshots if the data is corrupted.

4.2 SmartMigration

OceanStor 18000 V5 series uses SmartMigration for intelligent data migration based on LUNs. Data on a source LUN can be completely migrated to a target LUN without interrupting ongoing services. SmartMigration supports data migration within a Huawei storage system or between a Huawei storage system and a compatible heterogeneous storage system.

When the system receives new data during migration, it writes the new data to both the source and target LUNs simultaneously and records data change logs (DCLs) to ensure data consistency. After the migration is complete, the source and target LUNs exchange information to allow the target LUN to take over services.

SmartMigration involves data synchronization and LUN information exchange.

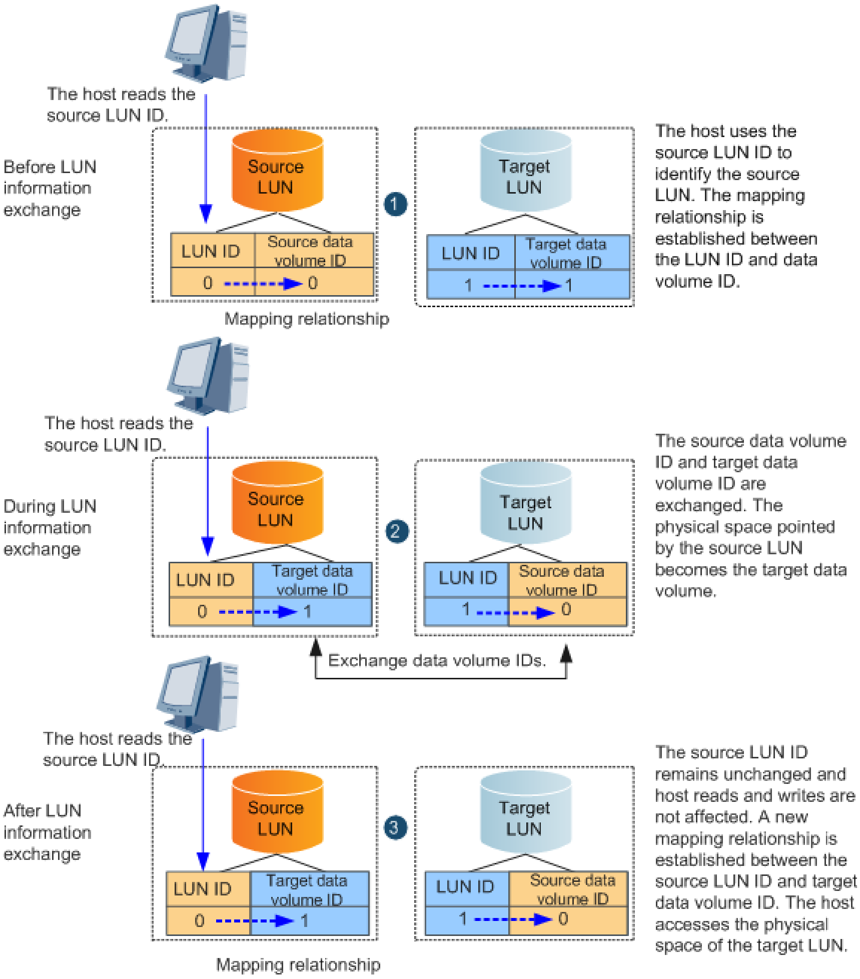
Data Synchronization

1. Before migration, you must configure the source and target LUNs.
2. When migration starts, the source LUN replicates data to the target LUN.
3. During migration, the host can still access the source LUN. When the host writes data to the source LUN, the system records the DCL.
4. The system writes the incoming data to both the source and target LUNs.
   - If writing to both LUNs is successful, the system clears the record in the DCL.
   - If writing to the target LUN fails, the storage system identifies the data that failed to be synchronized according to the DCL and then copies the data to the target LUN. After the data is copied, the storage system returns a write success to the host.
   - If writing to the source LUN fails, the system returns a write failure to notify the host to re-send the data. Upon reception, the system only writes the data to the source LUN.

LUN Information Exchange

After data replication is complete, host I/Os are suspended temporarily, and the source and target LUNs exchange information, as shown in Figure 4-2.

Figure 4-2 LUN information exchange


The host reads the source LUN ID.
Before LUN information exchange
Source LUN
LUN ID
Source data volume ID
0
0
Mapping relationship
1
Target LUN
LUN ID
Target data volume ID
1
1
The host uses the source LUN ID to identify the source LUN. The mapping relationship is established between the LUN ID and data volume ID.
The host reads the source LUN ID.
During LUN information exchange
Source LUN
LUN ID
Target data volume ID
0
1
2
Target LUN
LUN ID
Source data volume ID
1
0
The source data volume ID and target data volume ID are exchanged. The physical space pointed by the source LUN becomes the target data volume.
Exchange data volume IDs.
The host reads the source LUN ID.
After LUN information exchange
Source LUN
LUN ID
Target data volume ID
0
1
3
Target LUN
LUN ID
Source data volume ID
1
0
The source LUN ID remains unchanged and host reads and writes are not affected. A new mapping relationship is established between the source LUN ID and target data volume ID. The host accesses the physical space of the target LUN.
Mapping relationship

LUN information exchange is completed instantaneously, which does not interrupt services.

Application Scenarios

- Storage system upgrade with SmartVirtualization

  SmartMigration works with SmartVirtualization to migrate data from legacy storage systems (from Huawei or other vendors) to new Huawei storage systems to improve service performance and data reliability.

- Data migration for capacity, performance, and reliability adjustments

4.3 SmartDedupe and SmartCompression

SmartDedupe and SmartCompression provide data deduplication and compression functions to shrink data for file systems and thin LUNs, saving space while reducing the TCO of the enterprise IT architecture.

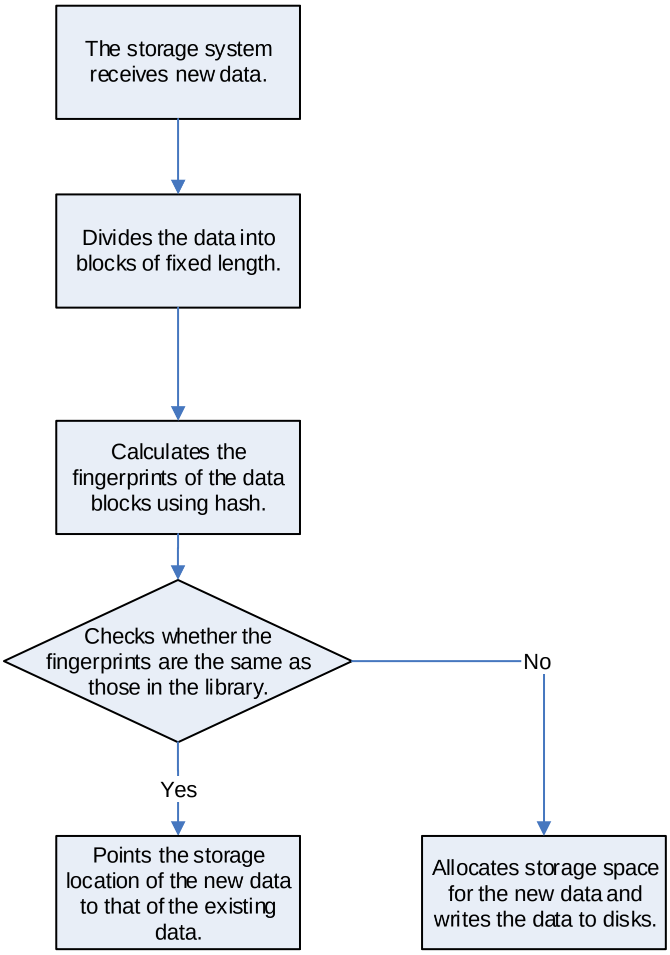
SmartDedupe

OceanStor 18000 V5 series uses SmartDedupe to implement inline deduplication for file systems and thin LUNs. In inline deduplication mode, the storage system deduplicates new data before writing it to disks.

The data deduplication granularity is consistent with the minimum data read and write unit (grain) of file systems or thin LUNs. As users can specify the grain size (4 KB to 64 KB) when creating file systems or thin LUNs, OceanStor 18000 V5 series can implement data deduplication based on different granularities.

Figure 4-3 shows how a storage system deduplicates data.

Figure 4-3 Deduplication process


The storage system receives new data.
Divides the data into blocks of fixed length.
Calculates the fingerprints of the data blocks using hash.
Checks whether the fingerprints are the same as those in the library.
No
Yes
Points the storage location of the new data to that of the existing data.
Allocates storage space for the new data and writes the data to disks.

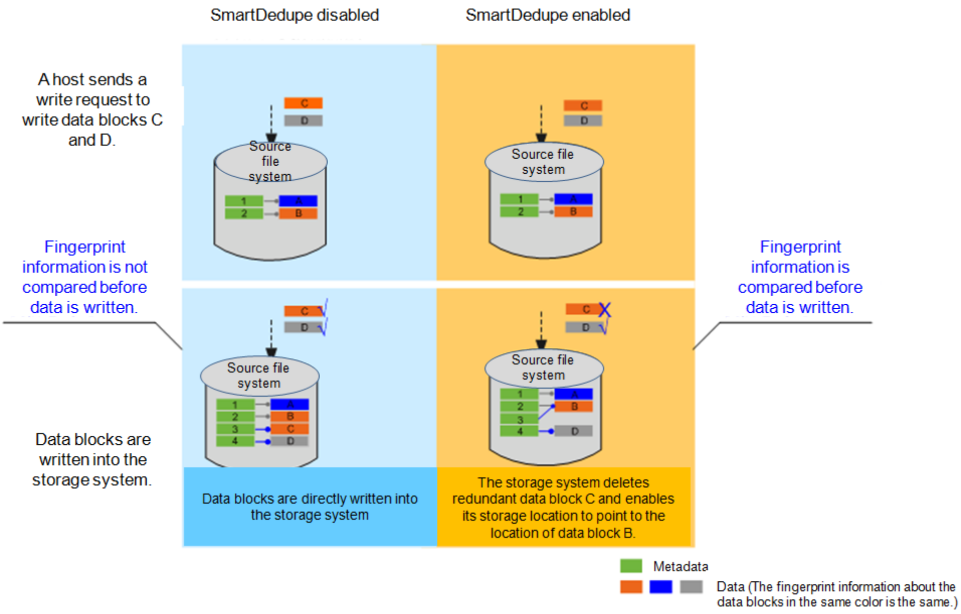
The process is described as follows:

1. The storage system divides new data into blocks based on the deduplication granularity.
2. The storage system compares the fingerprints of new data blocks with those of existing data blocks in the fingerprint library. If no identical fingerprints are found, the storage system writes the new data blocks. If the same fingerprints are found:
   - With byte-by-byte comparison disabled (default), the system identifies the data blocks as duplicate ones. It will not allocate storage space for these duplicate blocks, but only points their storage locations to those of the existing data blocks.

- With byte-by-byte comparison enabled, the storage system will compare the new data blocks with the existing data blocks byte by byte. If they are the same, the system identifies duplicate data blocks. If they are different, the system writes the new data blocks.

The following describes an example. A file system has data blocks A and B. An application server writes data blocks C and D to the file system. C has the same fingerprint as B, while D has a different fingerprint from A and B. Figure 4-4 shows how the data blocks are processed when different data deduplication policies are used.

Figure 4-4 Data processing with SmartDedupe enabled and disabled


SmartDedupe disabled
SmartDedupe enabled
A host sends a write request to write data blocks C and D.
C
D
Source file system
1
2
A
B
Fingerprint information is not compared before data is written.
Fingerprint information is compared before data is written.
3
4
Data blocks are written into the storage system.
Data blocks are directly written into the storage system
The storage system deletes redundant data block C and enables its storage location to point to the location of data block B.
Metadata
Data (The fingerprint information about the data blocks in the same color is the same.)

SmartCompression

Inline and post-process compression is available in the industry. OceanStor 18000 V5 series uses inline compression which compresses new data before it is written to disks. It has the following advantages in comparison to post-process compression:

- Requires less initial storage space, lowering the initial investment of customers.
- Generates fewer I/Os, applicable to SSDs that have restrictions on the number of reads and writes.
- Compresses data blocks after snapshots are created, saving space.

SmartCompression compresses data blocks based on the user-configured compression policy. The storage system supports the following two compression polices:

- Fast policy

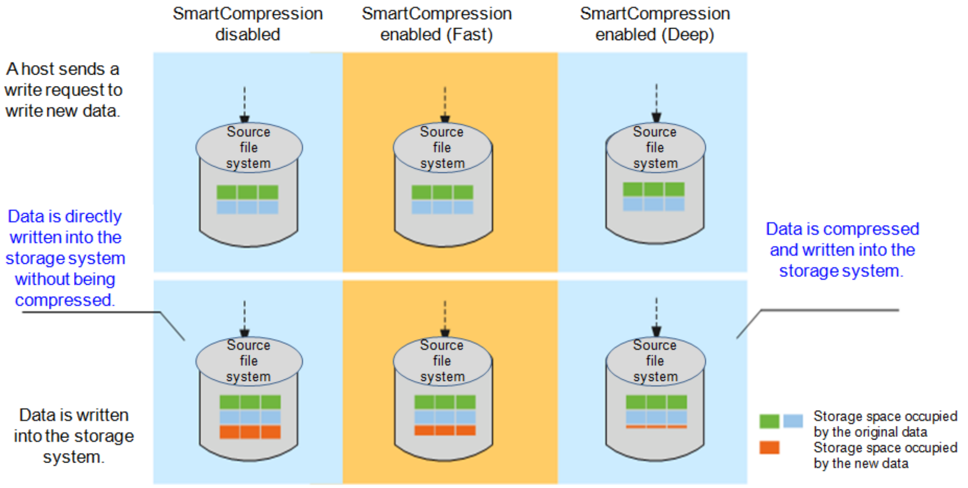
  It is the default compression policy. This policy has higher speed in compression but lower efficiency in capacity saving.

- Deep policy

  It significantly improves the efficiency in capacity saving but takes longer time in compression and decompression.

Figure 4-5 shows how the data blocks are processed when different data compression policies are used.

Figure 4-5 Data processing with SmartCompression enabled and disabled


SmartCompression disabled
SmartCompression enabled (Fast)
SmartCompression enabled (Deep)
A host sends a write request to write new data.
Source file system
Data is directly written into the storage system without being compressed.
Data is compressed and written into the storage system.
Data is written into the storage system.
Storage space occupied by the original data
Storage space occupied by the new data

Interworking of SmartDedupe and SmartCompression

SmartDedupe and SmartCompression can work together. When both of them are enabled, data is deduplicated and then compressed, saving more storage space.

SmartDedupe and SmartCompression provided by the OceanStor 18000 V5 series work in inline mode. When the functions are enabled, new data is deduplicated and compressed. When the functions are disabled, deduplicated data cannot be restored.

4.4 SmartTier

SmartTier for Block

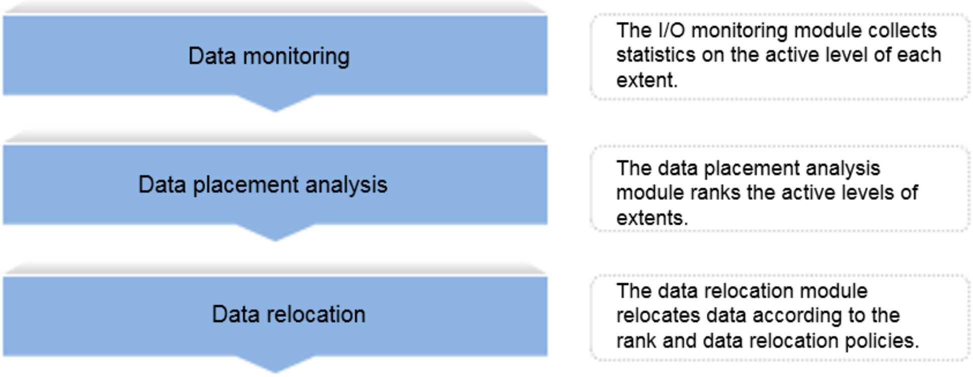
OceanStor 18000 V5 series uses SmartTier for intelligent data tiering.

SmartTier categorizes storage media into three storage tiers based on performance: high-performance tier (SSDs), performance tier (SAS disks), and capacity tier (NL-SAS disks). Storage tiers can be used independently or in combination to provide data storage space.

SmartTier performs intelligent data storage based on LUNs. It segments data into extents (whose size is 4 MB by default and can be configured to a value from 512 KB to 64 MB). SmartTier collects statistics on and analyzes the activity levels of data based on extents and matches data of various activity levels with proper storage media. Data that is more active will be promoted to higher-performance storage media (such as SSDs), whereas data that is less active will be demoted to more cost-effective storage media with larger capacities (such as NL-SAS disks).

SmartTier implements data monitoring, placement analysis, and data relocation, as shown in Figure 4-6.

Figure 4-6 SmartTier implementation



The I/O monitoring module collects statistics on the active level of each extent.

The data placement analysis module ranks the active levels of extents.

The data relocation module relocates data according to the rank and data relocation policies.

Data monitoring and data placement analysis are automated by the storage system, and data relocation is initiated manually or by a user-defined policy.

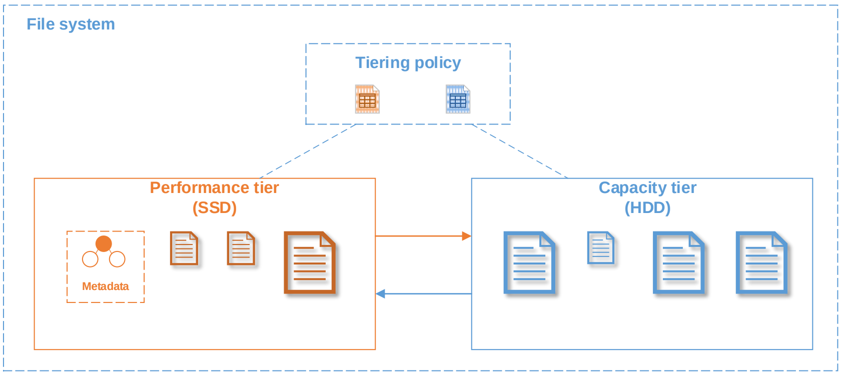
SmartTier improves storage system performance and reduces storage costs to meet enterprises' requirements on both performance and capacity. By preventing historical data from occupying expensive storage media, SmartTier ensures effective investment and eliminates energy consumption caused by useless capacities, reducing the TCO and optimizing the cost-effectiveness.

SmartTier for File

SmartTier also applies to file systems. It helps customers simplify data life cycle management, improve media usage, and reduce cost. SmartTier dynamically relocates data by file among different media based on user-defined tiering policies.

A storage pool may be composed of SSDs and HDDs. SmartTier automatically promotes files to high-performance media (SSDs) and demote files to large-capacity media (HDDs, including SAS and NL-SAS disks) based on user-configured tiering policies. Users can specify tiering policies by file name, file size, file type, file creation time, and SSD usage. Figure 4-7 shows the SmartTier working principles.

Figure 4-7 SmartTier working principles



SmartTier features:

- Custom tiering policies

Users can flexibly define tiering policies by file name, file size, file type, file creation time, SSD usage, or their combination to meet requirements in various scenarios.

- File access acceleration

  By default, file system metadata is stored in SSDs, which facilitates the locating of files and directories, thereby accelerating file access.

- Intelligent flow control

  File relocation increases CPU and disk loads. The storage system performs intelligent flow control on relocation tasks based on service pressure, minimizing the impact of data relocation on service performance.

- Saved cost

  SmartTier enables tiered storage. The storage system saves data on SSDs and HDDs, ensuring service performance at a lower cost as compared with All Flash Arrays (AFAs).

- Simplified management

  SmartTier supports tiered storage within a file system. It automatically relocates cold data to HDDs, archiving data without the need of other features or applications, thereby simplifying data life cycle management. Users are not aware of data relocation.

SmartTier applies to scenarios in which file life cycle management is required, such as financial check images, medical images, semiconductor simulation design, and reservoir analysis. The services in these scenarios have demanding requirements on performance in the early stage and have low requirements on performance later. The following describes an example.

In the reservoir analysis scenario, small files are imported to the storage system for the first time. These small files are frequently accessed and have high performance requirements. After small files are processed by professional analysis software, large files are generated. These large files have low requirements on performance. Users can configure tiering policies based on file sizes. To be specific, small files are stored in SSDs and large files are stored in HDDs (such as low-cost NL-SAS disks). In this way, SmartTier helps reduce customer's cost while meeting the performance requirements.

4.5 SmartThin

SmartThin enables the storage system to allocate storage resources on demand. SmartThin does not allocate all capacity in advance, but presents a virtual storage capacity larger than the physical storage capacity. In this way, you see a larger storage space than the actual allocated space. When you begin to use the storage, SmartThin provides only the required space. If the allocated storage space is about to use up, SmartThin triggers storage resource pool expansion to add more space. The expansion process is transparent to users and causes no system downtime.

SmartThin applies to:

- Core businesses that pose demanding requirements on continuity, such as bank transaction systems

  SmartThin allows customers to implement online capacity expansion without interrupting businesses.

- Businesses whose application system data usage fluctuates unpredictably, such as email services and online storage services

  SmartThin enables physical storage space to be allocated on demand, preventing resource waste.

- Businesses that involve various systems with diverse storage requirements, such as telecom carrier services

  SmartThin enables different applications to contend for physical storage space, improving space utilization.

4.6 SmartQoS

SmartQoS dynamically allocates storage system resources to meet the performance objectives of applications.

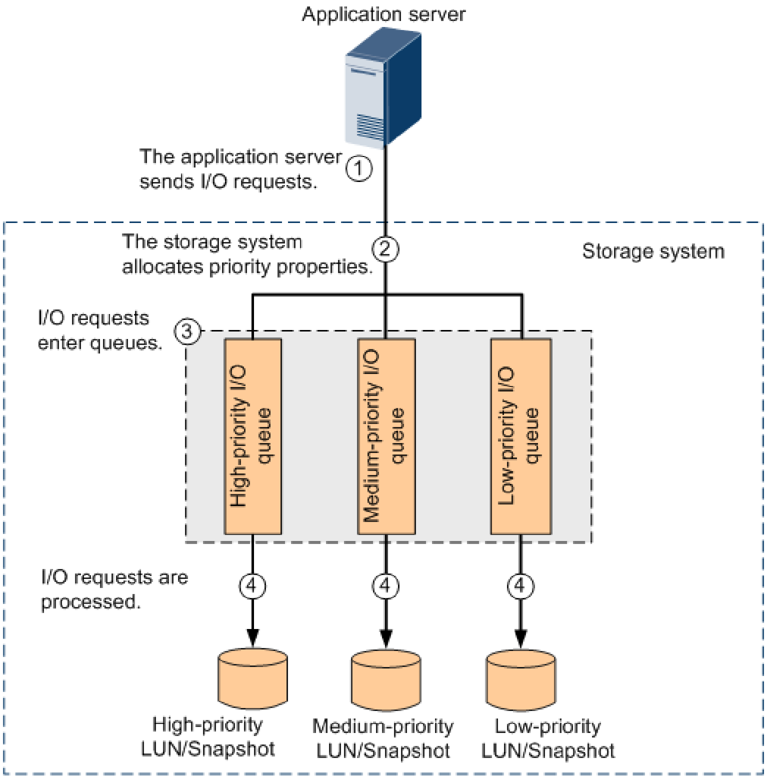
SmartQoS enables you to set upper limits on IOPS or bandwidth for specific applications. Based on the upper limits, SmartQoS can accurately limit performance of these applications, preventing them from contending for storage resources with critical applications.

SmartQoS uses LUN-, FS-, or snapshot-specific I/O priority scheduling and I/O traffic control to guarantee the service quality.

- I/O priority scheduling

  This schedules resources based on application priorities. When allocating system resources, a storage system prioritizes the resource allocation requests initiated by high-priority services. If resources are in shortage, a storage system allocates more resources to the high-priority services to meet their QoS requirements.

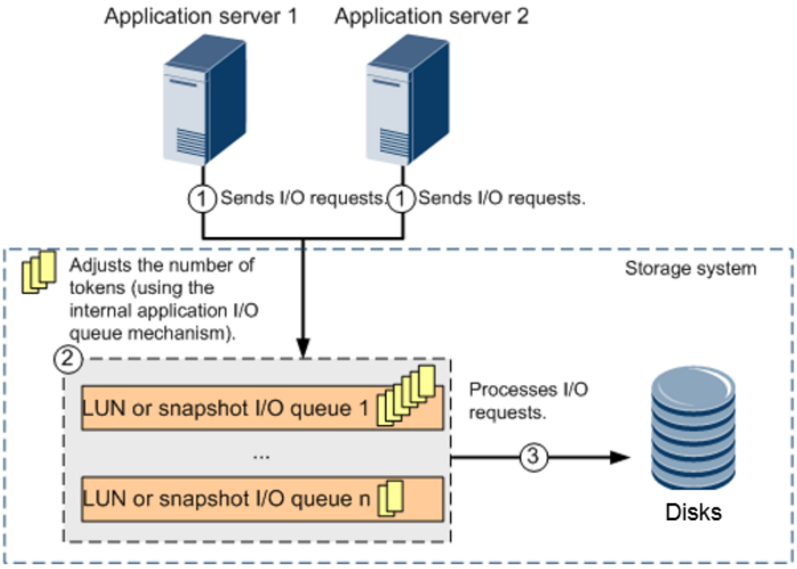
Figure 4-8 I/O priority scheduling process


Application server
1
The application server sends I/O requests.
2
The storage system allocates priority properties.
Storage system
3
I/O requests enter queues.
High-priority I/O queue
Medium-priority I/O queue
Low-priority I/O queue
4
I/O requests are processed.
4
4
High-priority LUN/Snapshot
Medium-priority LUN/Snapshot
Low-priority LUN/Snapshot

- I/O traffic control

This limits traffic of some applications by limiting their IOPS or bandwidth, thereby preventing these applications from affecting other applications. I/O traffic control involves I/O request processing, token distribution, and dequeuing control.

Figure 4-9 Managing LUN or snapshot I/O queues


Application server 1
Application server 2
1 Sends I/O requests.
1 Sends I/O requests.
Adjusts the number of tokens (using the internal application I/O queue mechanism).
Storage system
2
LUN or snapshot I/O queue 1
...
LUN or snapshot I/O queue n
Processes I/O requests.
3
Disks

4.7 SmartPartition

SmartPartition is a smart cache partitioning technique provided by OceanStor 18000 V5 series. SmartPartition ensures the performance of mission-critical applications by partitioning cache resources. An administrator can allocate a cache partition of a specific size to an application. The storage system ensures that the application uses the allocated cache resources exclusively.

NOTE

Cache is the most critical factor that affects the performance of a storage system.

- For a write service, a larger cache size means a higher write combination rate and higher write hit ratio (write hit ratio of a block in a cache).
- For a read service, a larger cache size means a higher read hit ratio.

Different types of services have different cache requirements.

- For a sequential service, the cache size must meet the I/O combination requirement.
- For a random service, a larger cache size indicates that I/Os are more likely to fall onto stripes within the cache, thereby improving performance.

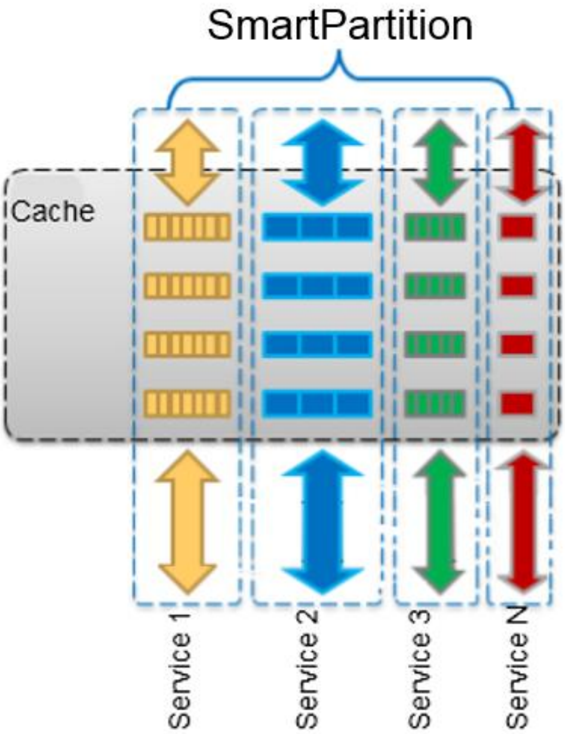
SmartPartition can be used with other QoS techniques (such as SmartQoS) for better QoS effects.

Working Principle

SmartPartition allocates cache resources to services (the actual control objects are LUNs and file systems) based on partition sizes, thereby ensuring the QoS of mission-critical services.

Figure 4-10 illustrates how SmartPartition works

Figure 4-10 SmartPartition working principle



Technical Highlights

- Intelligent partition control

  Based on user-defined cache sizes and QoS policies, SmartPartition automatically schedules system cache resources to ensure the optimal system QoS and the required partition quality.

- Ease of use

  SmartPartition is easy to configure. All configurations take effect immediately without restarting the system. Users do not need to adjust the partition, thereby improving the usability of partitioning.

Application Scenarios

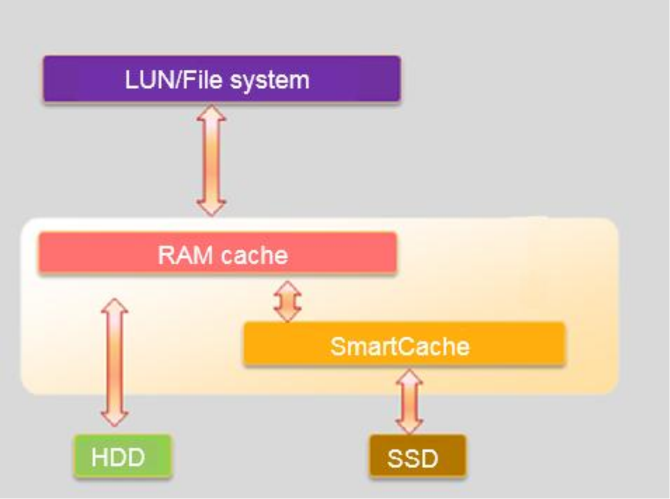
SmartPartition is applicable to scenarios where multiple applications exist, for example:

- A multi-service system. SmartPartition can be used to ensure core service's performance
- VDI scenario. SmartPartition can be used to ensure the performance for important users
- Multi-tenant scenarios in cloud computing systems

4.8 SmartCache

SmartCache, serving as a read cache module of a storage system, uses SSDs to store clean hot data that RAM cache cannot hold. Figure 4-11 shows the logical architecture of SmartCache.

Figure 4-11 Logical architecture of SmartCache



SmartCache improves the performance in accessing hot data by LUN or file system. The working principle is as follows:

1. After a LUN or file system is enabled with SmartCache, RAM cache delivers hot data to SmartCache.
2. SmartCache establishes a mapping relationship between the data and the SSD in the memory and stores the data onto the SSD.
3. When the host delivers a new read I/O to the storage system, the system preferentially attempts to look for the required data in RAM cache.
   - If the attempt fails, the system then attempts to look for required data in SmartCache.
   - If the required data is found in SmartCache, corresponding data is read from the SSD and then returned to the host.

When the amount of data buffered in SmartCache reaches the upper limit, SmartCache selects cache blocks according to the least recently used (LRU) algorithm, clears mapping items in the lookup table, and eliminates data on the buffer blocks. Data write and elimination are performed repeatedly, ensuring that data stored on SmartCache is the data frequently accessed.

Application Scenarios

SmartCache applies to services that have hotspot areas and intensive random read I/Os, such as databases, OLTP applications, web services, and file services.

4.9 SmartErase

As a head cannot read data from or write data to the same point every time, the newly written data cannot precisely overwrite the original data. For this reason, some data remains. Dedicated devices can be used to obtain the copy of the original data (called data shadow). More times of overwriting ensure less residual data.

SmartErase uses overwriting to destroy data on LUNs. SmartErase provides two data destruction methods for your choice: DoD 5220.22-M standard and customized:

- DoD 5220.22-M

  DoD 5220.22-M is a data destruction standard put forward by the US Department of Defense (DoD). The standard provides a software method of destroying data on writable storage media, namely, three times of overwriting:

  - Using an 8-bit character to overwrite all addresses
  - Using the complementary codes of the character (complements of 0 and 1) to overwrite all addresses
  - Using a random character to overwrite all addresses

- Customized

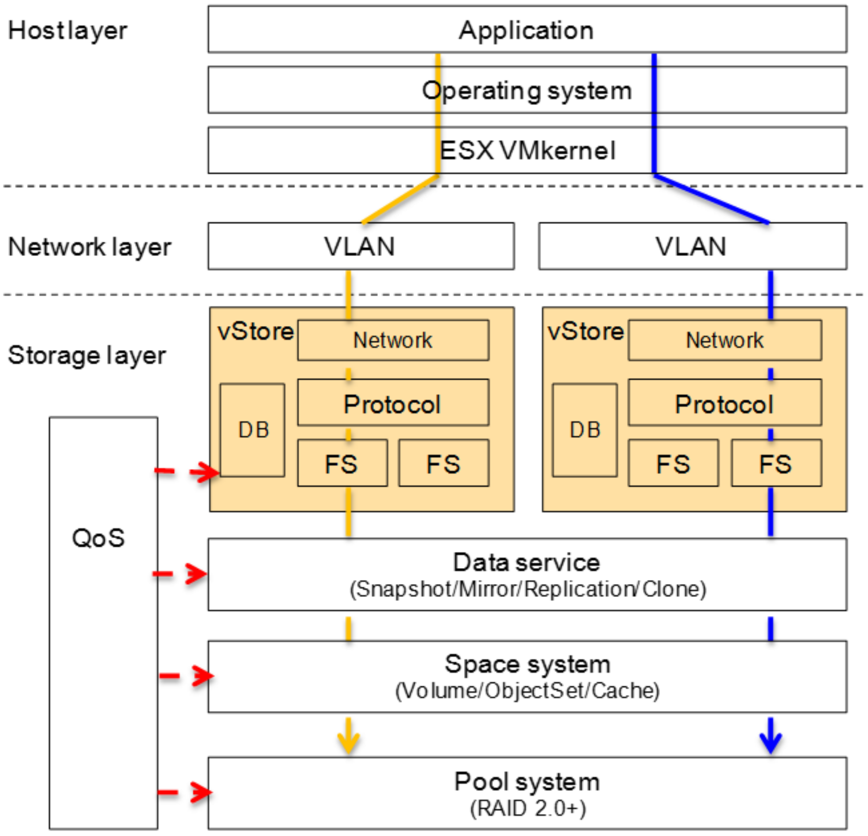
For customized overwriting, a system generates data based on internal algorithms and uses the data to overwrite all addresses of LUNs for specific times. The times of overwriting range from 3 to 99. The default value is 7.

4.10 SmartMulti-Tenant

SmartMulti-Tenant allows the creation of multiple virtual storage systems (vStore) in a physical storage system. vStores can share the same storage hardware resources in a multi-protocol unified storage architecture, without affecting data security and privacy of each other.

SmartMulti-Tenant implements management, network, and resource isolation. In this way, no data access is allowed between vStores, ensuring security.

Figure 4-12 Logical architecture of SmartMulti-Tenant



- Management isolation

  Each vStore has its own administrator. vStore administrators can only configure and manage their own storage resources through the GUI or RESTful API. vStore administrators support role-based permission control. When being created, a vStore administrator is assigned a role specific to its permissions.

- Service isolation

  Each vStore has its own file systems, users, user groups, shares, and exports. Users can only access the file systems that belong to the vStore through the logical interfaces (LIFs).

  Service isolation includes: service data isolation (covering file systems, quotas, and snapshots), service access isolation, and service configuration isolation (typically for the NAS protocol configuration).

  - Service data isolation

    System administrators assign different file systems to different vStores, thereby achieving file system isolation. File system quotas and snapshots are isolated in the same way.

  - Service access isolation

    Each vStore has its own NAS protocol instances, including the SMB service, NFS service, and NDMP service.

  - Service configuration isolation

    Each vStore can have its own users, user groups, user mapping rules, security policies, SMB shares, NFS shares, AD domain, DNS service, LDAP service, and NIS service.

- Network isolation

  VLANs and LIFs are used to isolate vStore's network, thereby preventing illegal host access to vStore's storage resources.

  vStores use LIFs to configure services. A LIF belongs only to one vStore to achieve logical port isolation. You can create LIFs from GE ports, 10GE ports, bond ports, or VLANs.

4.11 SmartQuota

In a NAS file service environment, resources are provided to departments, organizations, and individuals as shared directories. Because each department or person has unique resource requirements or limitations, storage systems must allocate and restrict resources based on shared directories in a customized manner. SmartQuota perfectly tackles the previous challenges. It can restrict and control resource consumption for directories, users, and user groups.

SmartQuota allows you to configure the following quotas:

- Space soft quota

  Specifies a soft space limit. If any new data writes that would result in exceeding this limit occur, the storage system reports an alarm indicating that the space is insufficient and asking the user to delete unnecessary files or expand the quota. The user can continue to write data to the directory.

- Space hard quota

  Specifies a hard space limit. If any new data writes that would result in exceeding this limit occur, the storage system prevents the writes and reports an error.

- File soft quota

  Specifies a soft limit on the file quantity. If the number of used files exceeds this limit, the storage system reports an alarm indicating that the file resources are insufficient and asking the user to delete unnecessary files or expand the quota. The user can continue to create files or directories.

- File hard quota

  Specifies a hard limit on the file quantity. If the number of used files of a quota exceeds this limit, the storage system prevents the creation of new files or directories and reports an error.

SmartQuota uses space and file hard quotas to restrict the maximum number of resources available to each user. The process is as follows:

1. In each write I/O operation, SmartQuota checks whether the accumulated quota (Quotas of the used space and file quantity + Quotas of the increased space and file quantity in this operation) exceeds the preset hard quota.
   - If yes, the write I/O operation fails.
   - If not, the follow-up operations can be performed.
2. After the write I/O operation is allowed, SmartQuota adds the incremental space and file quantity to the previously used space and file quantity, separately.
3. SmartQuota updates the quota (used space and file quantity + incremental space and file quantity) and enables the quota and I/O data to be written into the file system.

The I/O operation and quota update succeed or fail at the same time, ensuring that the used capacity is correct in each I/O check.

NOTE

If the directory quota, user quota, and group quota are concurrently configured in a shared directory in which you are performing operations, each write I/O operation will be restricted by the three quotas. All types of quota are checked. If the hard quota of one type of quota does not pass the check, the I/O will be rejected.

SmartQuota works as follows to clear alarms: When the used resource of a user is lower than 90% of the soft quota, SmartQuota clears the resource over-usage alarm. In this way, even though the used resource is slightly higher or lower than the soft quota, alarms are not frequently generated or cleared.

4.12 SmartMotion

In the IT industry, enterprises and administration departments are faced with challenges concerning capacity, performance, and cost in data storage. Enterprises cannot accurately assess growth of service performance when purchasing storage systems. Besides, as the service volume grows, it is hard to adjust existing services after disks are added to legacy storage systems.

To address the preceding problems, enterprises must develop a long-term performance requirement plan at the initial stage of IT system construction.

SmartMotion dynamically migrates data and evenly distributes data on all disks, resolving the problems facing customers. Customers need to assess only the recent performance requirements when purchasing storage systems, significantly reducing the initial purchase cost and total TCO. If the requirement for system performance increases with the service volume, customers only need to add disks to storage systems. Then SmartMotion migrates data and evenly distributes the original service data onto all disks, notably improving service performance.

SmartMotion is implemented based on RAID 2.0+. For RAID 2.0+, the space of all the disks in a disk domain is divided into fixed CKs. When CKGs are required, disks are selected in a pseudo-random manner and CKs from these disks compose CKGs based on a certain RAID algorithm. All CKs are evenly distributed onto all the disks.

When disks are added into a disk domain, the storage system starts SmartMotion. The process for implementing a SmartMotion task is as follows:

1. Selects the first CKG that is not load-balanced.
2. Selects disks for the CKG in a pseudo-random manner.
   - If the selected disks are consistent with the original disks for the CKG, skips this CKG and goes back to 1.
   - If they are inconsistent, goes to 3.
3. Compares the original disks for the CKG with the newly selected disks and computes the mapping between the source disks and the target disks based on disk difference. Then selects the source disks and target disks.
4. Traverses all the source disks for the CKG, allocates new CKs from the target disks, and migrates data from the source disks to the target disks to release the source disks.
5. After all CKGs in the system are traversed, the SmartMotion task is complete. Otherwise, goes back to 1 and processes the next CKG.

After the SmartMotion task is complete, disks are selected for all CKGs in a pseudo-random manner and required data is migrated. All CKs are evenly distributed onto all disks including newly added disks.

5 Hyper Series Features

5.1 HyperSnap

5.1.1 HyperSnap for Block

OceanStor 18000 V5 series uses HyperSnap to quickly generate a consistent image, that is, a duplicate, for a source LUN at a point in time without interrupting services running on the source LUN. The duplicate is available immediately after being generated. Reading or writing the duplicate has no impact on the source data. HyperSnap helps with online backup, data analysis, and application testing. It works based on the mapping table and copy-on-write (COW) technology.

Technical Highlights

- Zero backup window

  Traditional backup deteriorates application servers' performance, or even interrupts ongoing services. Therefore, a traditional backup task can be executed only after application servers are stopped or during off-peak hours. A backup window refers to the data backup duration, which is the maximum downtime tolerated by applications. HyperSnap can back up data online, requiring a close-to-zero backup window without interrupting services.

- Less occupied disk capacity

After creating a consistent copy of a source LUN, HyperSnap uses a COW volume to save the data on the source LUN at the snapshot point in time upon the first update. The COW volume size is independent of the source LUN size but dependent on the amount of data changed on the source LUN. When the amount of changed data is small, the snapshot captures a consistent copy of the source LUN and uses a small disk space. The consistent copy can be used for service tests, saving disk space.

- Quick data restoration

  Data backed up in traditional offline backup approaches cannot be read online. Long-time data restoration is required before a usable duplicate of the source data at the backup point in time is available. HyperSnap can directly read the snapshot volume to obtain data on the source volume at the snapshot point in time, thereby restoring data in the case of data corruption on the source volume quickly.

- Data consistency by consistency group

  In OLTP applications, snapshots for multiple pieces of source LUN data must be created at the same time. In this way, associated data of the applications distributed on different LUNs can keep at the same point in time. For example, the management data, service data, and log information of an Oracle database application are distributed on different source LUNs. Consistent copies of the three source LUNs must be created at the same time. Otherwise, the three source LUNs cannot be restored to the same point in time, losing data dependency. HyperSnap provides snapshot consistency groups to resolve this problem. I/Os on multiple source LUNs are frozen at the snapshot point in time, and a snapshot is generated for the frozen I/Os.

- Continuous data protection by timing snapshots

  OceanStor 18000 V5 series allows snapshots to be created for a source LUN at multiple points in time. Working together with ReplicationDirector on the host, HyperSnap can create or delete snapshots at a minute-level interval. In addition, a snapshot policy can be set to automate the activation and stopping of snapshot tasks. As time elapses, snapshots are generated at multiple points, implementing continuous data protection at a low cost.

- Snapshot copy

  A snapshot copy backs up the data of a snapshot at the snapshot activation point in time. It does not back up data written to the snapshot after the snapshot activation point in time. The snapshot copy and source snapshot share the COW volume space of the source LUN, but the private space is independent. The snapshot copy is a writable snapshot and is independent of the source snapshot. The read and write processes of a snapshot copy are the same as those of a common snapshot.

  Snapshot copy allows users to obtain multiple data copies of a snapshot for various purposes.

5.1.2 HyperSnap for File

OceanStor 18000 V5 series uses HyperSnap to quickly generate a consistent image, that is, a duplicate, for a source file system at a certain point in time without interrupting services running in the source file system. The duplicate is available immediately after being generated. Reading or writing the duplicate does not impact on data in the source file system. HyperSnap helps with online backup, data analysis, and application testing. HyperSnap can:

- Create file system snapshots and back up the snapshots to tapes.
- Provide data backups of the source file system for end users to restore files accidentally deleted.
- Work together with HyperReplication and HyperVault for remote replication and backup.

HyperSnap works based on ROW file systems. In a ROW file system, new or modified data does not overwrite the original data but is written to newly allocated storage space, which ensures enhanced data reliability and high file system scalability. ROW-based HyperSnap for file systems can create snapshots in seconds. The snapshot data does not occupy additional disk space unless the source files are deleted or modified.

Technical Highlights

- Zero backup window

  A backup window refers to the data backup duration, which is the maximum downtime tolerated by applications. The traditional backup deteriorates file system performance, or even interrupts ongoing applications. Therefore, a traditional backup task can only be executed after applications are stopped or the workload is comparatively light. HyperSnap can back up data online, requiring a close-to-zero backup window without interrupting services.

- Snapshot creation within seconds

  To create a snapshot for a file system, only the root node of the file system needs to be copied and stored in caches protected against power failure. The snapshot creation time is reduced to seconds.

- Reduced performance loss

  HyperSnap makes it easy to create snapshots for file systems. Only a small amount of data needs to be stored onto disks. After a snapshot is created, the system additionally checks whether data is protected by a snapshot before releasing the data space. If the data is protected by a snapshot, the system records the space of the data block that is protected by the snapshot but is deleted by the file system. This imposes a negligible impact on system performance. Background data space reclamation contends some CPU and memory resources against file system services only when the snapshot is deleted. However, the performance loss remains low.

- Less occupied disk capacity

  The file system space occupied by a snapshot (a consistent duplicate) of the source file system depends on the amount of changed data after a snapshot is generated, and never exceeds the file system size at the snapshot point in time. For a file system with not much changed data, only a little storage space is required to generate a consistent duplicate of the file system.

- Rapid snapshot data access

  A file system snapshot is presented in the root directory of the file system as an independent directory. Users can access the directory to quickly access the snapshot data. When a snapshot rollback is not required, users can easily access the data at the snapshot point in time and recover data by copying the file or directory if the file data in the file system is corrupted.

  If using a Windows client to access a CIFS-based file system, a user can restore a file or folder to the state at a specific snapshot point in time. To be specific, a user can right-click the desired file or folder, choose **Restore previous versions** from the short-cut menu, and select one for restoration from the displayed list of available snapshots containing the previous versions of the file or folder.

- Quick file system rollback

  Backup data generated by traditional offline backup tasks cannot be read online. A time-consuming data recovery process is inevitable before a usable duplicate of the source data at the backup point in time is available. HyperSnap can directly replace the file system root with specific snapshot root and clears cached data to quickly roll the file system back to a specific snapshot point in time.

You must exercise caution when using the rollback function because snapshots created after the rollback point in time are automatically deleted after the file system rollback succeeds.

- Continuous data protection by timing snapshots

  HyperSnap enables users to configure polices to automatically create snapshots at specific time points or at a specific interval.

  The maximum number of snapshots for a file system varies depending on the product model. If the upper limit is exceeded, earliest snapshots are automatically deleted. The file system also allows users to delete snapshots periodically.

  As time elapses, snapshots are generated at multiple points, implementing continuous data protection at a low cost. It must be noted that the snapshot technology cannot achieve real continuous data protection. The interval between two snapshots determines the granularity of continuous data protection.

5.2 HyperClone

5.2.1 HyperClone for Block

HyperClone generates a complete physical copy of a source LUN at a point in time without interrupting ongoing services. If the clone is split, writing data to and reading data from the physical copy do not affect source LUN data.

Working Principle

HyperClone is implemented by a combination of bitmap and COW as well as a combination of bitmap and dual-write (where data is written to the primary and secondary LUNs simultaneously). The working principle is as follows:

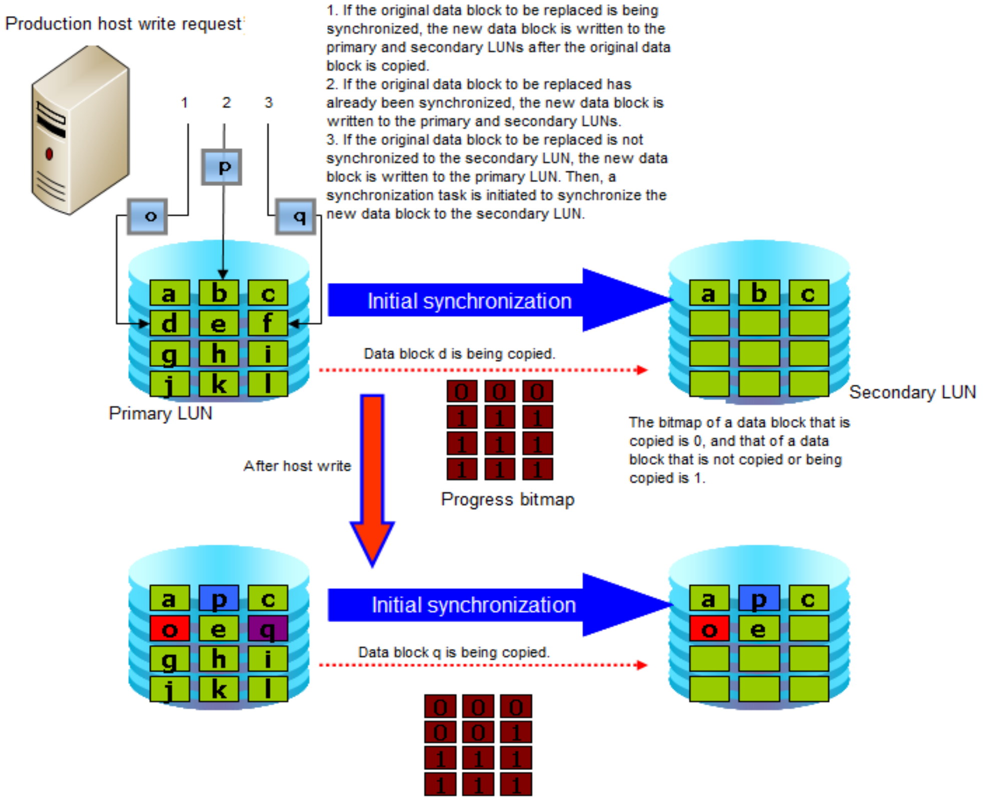
After a secondary LUN is added to a clone group, all data in the primary LUN is replicated to the secondary LUN by default, called initial synchronization. A progress bitmap reflects the synchronization process. If the primary LUN receives a write request from the production host during the initial synchronization, the storage system checks the synchronization progress, and performs subsequent operations as follows:

- If the write-targeted data block has not been synchronized to the secondary LUN, data is written to the primary LUN and the storage system returns a write success acknowledgement to the host. Then, the data is synchronized to the secondary LUN in the subsequent synchronization task.
- If the write-targeted data block has already been synchronized, data is written to both the primary and secondary LUNs.
- If the write-targeted data block is being synchronized, the storage system waits until the data block is copied. Then, the storage system writes data to both the primary and secondary LUNs.

After the initial synchronization is complete, the clone group can be split. After splitting, you can use the primary and secondary LUNs separately for testing and data analysis. Changing data in a primary or secondary LUN does not affect the other, and the progress bitmap records data changes on both LUNs.

Figure 5-1 illustrates the HyperClone working principle.

Figure 5-1 HyperClone working principle



1. If the original data block to be replaced is being synchronized, the new data block is written to the primary and secondary LUNs after the original data block is copied.
2. If the original data block to be replaced has already been synchronized, the new data block is written to the primary and secondary LUNs.
3. If the original data block to be replaced is not synchronized to the secondary LUN, the new data block is written to the primary LUN. Then, a synchronization task is initiated to synchronize the new data block to the secondary LUN.

The bitmap of a data block that is copied is 0, and that of a data block that is not copied or being copied is 1.

Technical Highlights

- 1-to-16 mode

  HyperClone allows you to assign a maximum of 16 secondary LUNs for a primary LUN. A clone in 1-to-N mode can back up multiple copies of source data for various data analyses.

- Zero backup window

  HyperClone backs up data without interrupting services, ensuring a zero backup window.

- Dynamic adjustment of the copy speed

  You can manually change the copy speed to prevent a conflict between a synchronization task and a production service. If a storage system has detected that the service load is heavy, you can manually lower the copy speed to free system resources for services. When the service load is light, you can increase the copy speed to mitigate service conflicts in peak hours.

- Reverse synchronization

  If data on the primary LUN is incomplete or corrupted, you can recover the original service data by performing an incremental reverse synchronization from the secondary LUN to the primary LUN.

- Automatic recovery

  If a problem occurs, for example, the primary or secondary LUN fails, the corresponding clone created on the system will enter the disconnected state. After the problem is resolved, the clone is recovered based on a specified recovery policy.

  - If the policy is automatic recovery, the clone automatically enters the synchronizing state, and differential data is incrementally synchronized to the secondary LUN.
  - If the policy is manual recovery, the clone waits to be recovered and you must manually initiate a synchronization.

  Incremental synchronization greatly reduces the fault/disaster recovery time.

- Clone consistency group

  In OLTP applications, you must typically simultaneously split multiple clone pairs to obtain data copies at the same point in time. In this way, associated data distributed on different LUNs can be maintained at the same point in time. HyperClone can split multiple clone pairs simultaneously, freezing data on multiple primary LUNs at the split point in time, and obtaining consistent copies of the primary LUNs.

Application Scenarios

- Data backup

  HyperClone can generate multiple physical copies of a primary volume and allow multiple services to access data concurrently.

- Data recovery and protection

  If primary LUN data is corrupted by a virus or human error, or is physically damaged, a data copy of the secondary LUN at a proper point in time can be reversely copied to the primary LUN. Then, the primary LUN is restored to the state when the data copy was created.

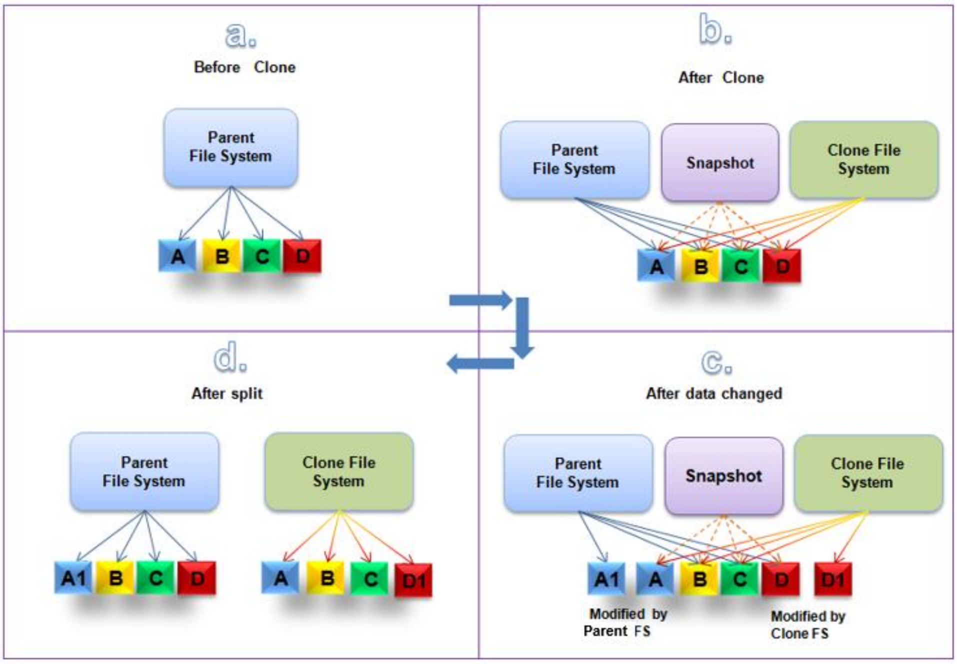
5.2.2 HyperClone for File

HyperClone creates a clone file system, that is, a copy, for a parent file system at a specified time point. Clone file systems can be shared to clients exclusively to meet requirements of rapid deployment, application tests, and DR drills.

Working Principle

A clone file system is a readable and writable copy at a time point based on the redirect-on-write (ROW) and snapshot technologies.

Figure 5-2 Working principle of HyperClone for File



- As shown in Figure a, the storage system writes new or modified data onto the newly allocated space of the ROW-based file system, instead of overwriting original data. The storage system records the time point of each data write, indicating the write sequence. Time points are serial numbers in ascending order.
- As shown in Figure b, the storage system creates a clone file system as follows:
  - Creates a read-only snapshot in the parent file system.
  - Copies the root node of the snapshot to generate the root node of the clone file system.
  - Creates an initial snapshot in the clone file system.

  This process is similar to that of creating a read-only snapshot during which no user data is copied. The snapshot creation can be complete in one or two seconds. Before data is modified, the clone file system shares data with its parent file system.
- As shown in Figure c, modifying the parent file system or the clone file system does not affect the other one.
  - When the application server modifies data block A of the parent file system, the storage pool allocates new data block A1 to store new data. Data block A is not released because it is protected by snapshots.
  - When the application server modifies data block D of the clone file system, the storage pool allocates new data block D1 to store new data. Data block D is not released because its write time is earlier than the creation time of the clone file system.
- Figure d shows the procedure for splitting a clone file system:
  - Deletes all read-only snapshots from the clone file system.
  - Traverses the data blocks of all objects in the clone file system. Allocates new data blocks in the clone file system for the shared data by overwriting data. In this way, the shared data is split.
  - Deletes the associated snapshots from the parent file system.

After the splitting is complete, the clone file system is independent of the parent file system. The time required to split the clone file system depends on the size of the share data.

Technical Highlights

- Rapid deployment

  In most scenarios, a clone file system can be created in seconds and is accessible immediately after being created.

- Saved storage space

  A clone file system shares data with its parent file system and occupies extra storage space only when it modifies shared data.

- Effective performance assurance

  HyperClone has negligible impact on the system performance because a clone file system is created based on the snapshot of the parent file system.

- Splitting a clone file system

  After a clone file system and its parent file system are split, they become completely independent of each other.

5.3 HyperReplication

OceanStor 18000 V5 series uses HyperReplication in synchronous mode (HyperReplication/S) and asynchronous mode (HyperReplication/A) to implement remote replication. Developed on the OceanStor OS unified storage software platform, OceanStor 18000 V5 series is compatible with the replication protocols of all Huawei OceanStor converged storage products. Therefore, OceanStor 18000 V5 series can interconnect with all new or legacy Huawei converged storage systems to construct a highly flexible disaster recovery solution.

OceanStor 18000 V5 series supports HyperReplication/S for Block, HyperReplication/A for Block, and HyperReplication/A for File.

5.3.1 HyperReplication/S for Block

Working Principle

HyperReplication/S maintains data consistency between primary and secondary LUNs based on a log mechanism. The working principle of HyperReplication/S is as follows:

After a synchronous remote replication relationship is set up between primary and secondary LUNs, an initial synchronization is implemented to replicate all data from the primary LUN to the secondary LUN.

After the initial synchronization is complete, I/Os are processed as follows:

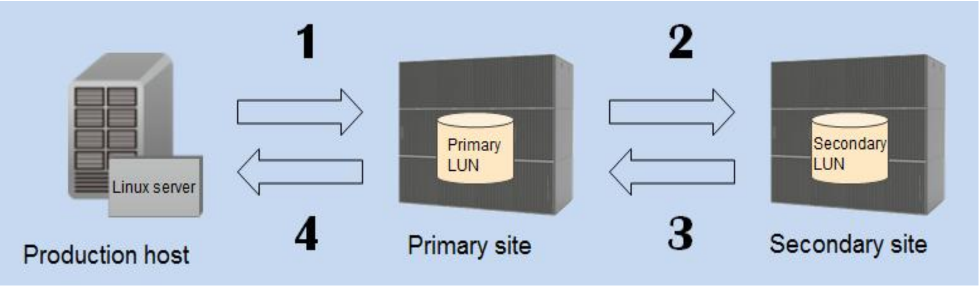
1 The primary site receives a write request from a production host. HyperReplication sets the differential log value to differential for the data block corresponding to the request.

2 The requested data is written to both the primary and secondary LUNs. When writing data to the secondary LUN, the primary site sends the data to the secondary site over a preset link.

3 If data is successfully written to both the primary and secondary LUNs, the corresponding differential log value is changed to non-differential. Otherwise, the value remains differential, and the data block is copied again in the next synchronization.

4 The primary site returns a write acknowledgement to the production host.

Figure 5-3 Working principle of HyperReplication/S



Technical Highlights

- Zero data loss

  HyperReplication/S synchronizes data from the primary LUN to the secondary LUN in real time, ensuring zero recovery point objective (RPO).

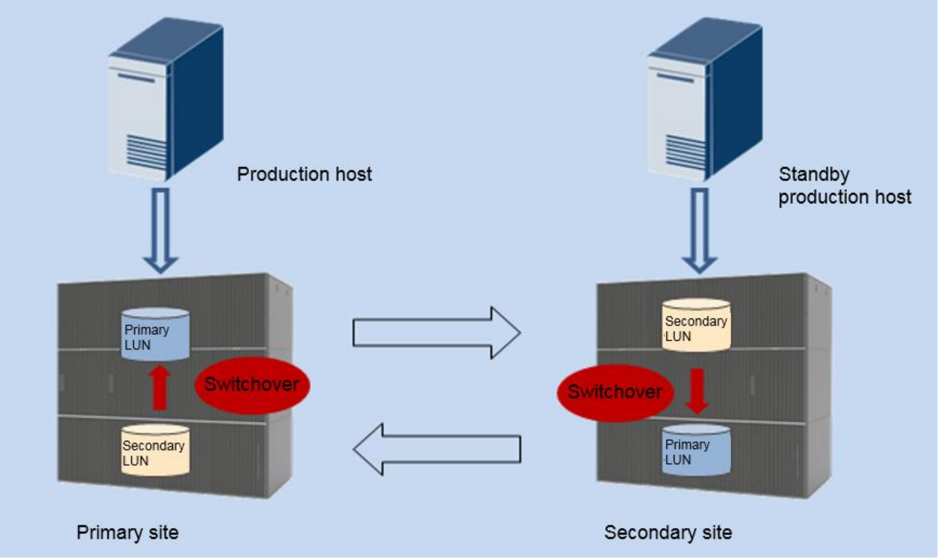
- Support for the split mode

  In split mode, write requests initiated by the production host are delivered only to the primary LUN. This mode meets certain user needs, such as temporary link maintenance, network bandwidth expansion, and saving data at a certain point in time on the secondary LUN.

- Primary/Secondary switchover

  HyperReplication/S supports primary/secondary switchover. In the following figure, the primary LUN at the primary site becomes the new secondary LUN after the switchover, and the secondary LUN at the secondary site becomes the new primary LUN. You need only perform some simple operations on the host side. The major operation, which can be performed in advance, is to map the new primary LUN to the standby production host. Then, the standby production host at the secondary site takes over services and delivers subsequent I/O requests to the new primary LUN.

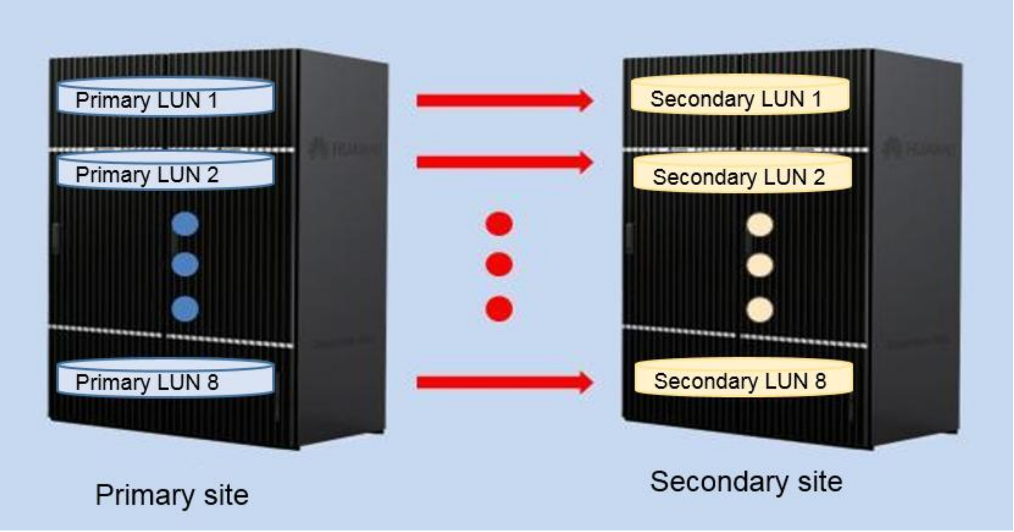
Figure 5-4 Primary/secondary switchover



- Support for consistency groups

  HyperReplication/S provides the consistency group function to ensure that data is replicated among LUNs simultaneously. HyperReplication/S allows you to add remote replication pairs to a consistency group. When you perform splitting, synchronization, or a primary/secondary switchover for a consistency group, these operations apply to all members of the consistency group. In addition, if a fault occurs, all members of the consistency group enter the disconnected state simultaneously.

Figure 5-5 Consistency group of HyperReplication/S



Application Scenarios

HyperReplication/S applies to local data disaster recovery and backup, that is, scenarios where the primary site is near the secondary site, for example, intra-city disaster recovery. For HyperReplication/S, a write success acknowledgement is returned to the production host only after the data in the write request is written to both the primary site and secondary site. If the primary site is far away from the secondary site, the write latency of foreground applications is relatively high, affecting foreground services.

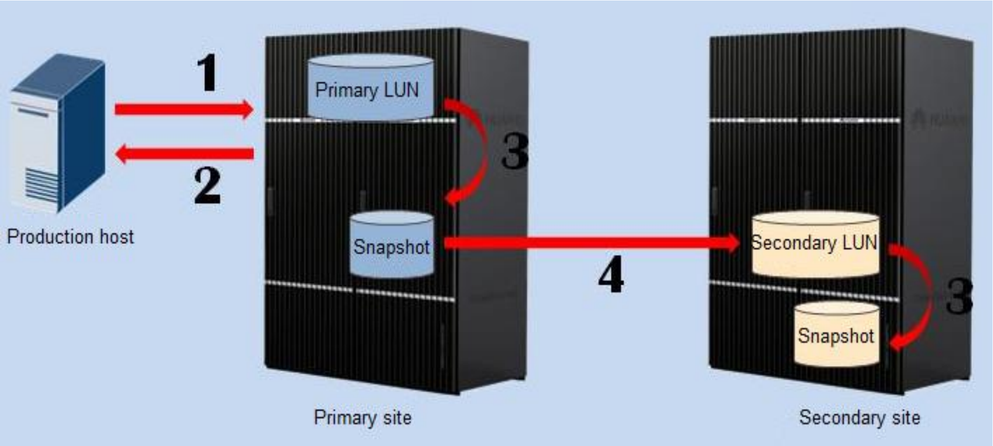
5.3.2 HyperReplication/A for Block

Working Principle

The working principle of HyperReplication/A is similar to that of HyperReplication/S: After an asynchronous remote replication relationship is set up between primary and secondary LUNs, an initial synchronization is implemented to replicate all data from the primary LUN to the secondary LUN. After the initial synchronization is complete, the data status of the secondary LUN is changed to **Synchronized** or **Consistent**. Then, I/Os are processed as follows:

1. The primary site receives a write request from a production host.
2. The primary site writes the new data to the primary LUN and immediately sends a write acknowledgement to the host.
3. Incremental data is automatically synchronized from the primary LUN to the secondary LUN based on a user-defined synchronization period, ranging from 1 to 1440 minutes. (If the synchronization type is **Manual**, you must trigger the synchronization manually.)
4. Before synchronization begins, a snapshot is generated for the primary and secondary LUNs respectively. The snapshot of the primary LUN ensures that the data read from the primary LUN during the synchronization remains unchanged. The snapshot of the secondary LUN backs up the secondary LUN's data in case an exception during synchronization causes the data to become unavailable.
5. During the synchronization, data is read from the snapshot of the primary LUN and copied to the secondary LUN. After synchronization is complete, the snapshots of the primary and secondary LUNs are canceled. The system waits the next synchronization.

Figure 5-6 Working principle of HyperReplication/A



Technical Highlights

- Data compression and data encryption

  HyperReplication/A supports data encryption specific to iSCSI links using the AES-256 algorithm. It supports data compression specific to iSCSI links. The data compression ratio varies significantly by service data type. The maximum compression ratio of database services is 4:1.

- Quick response to host requests

  After a host writes data to the primary LUN at the primary site, the primary site immediately returns a write acknowledgement to the host before the data is written to the secondary LUN. In addition, data is synchronized from the primary LUN to the secondary LUN in the background, without any impact on the access to the primary LUN. HyperReplication/A does not synchronize incremental data from the primary LUN to the secondary LUN in real time. Therefore, the amount of lost data is determined by the synchronization period, ranging from 3 seconds (default value) to 1440 minutes, that you can specify based on site requirements.

- Splitting, primary/secondary switchover, and rapid fault recovery

  HyperReplication/A supports splitting, synchronization, primary/secondary switchover, and recovery functions.

- Consistency groups

  You can create and delete consistency groups, create and delete HyperReplication pairs in a consistency group, and split pairs. When you perform splitting, synchronization, or a primary/secondary switchover for a consistency group, these operations apply to all members of the consistency group.

Application Scenarios

HyperReplication/A applies to remote data disaster recovery and backup, that is, scenarios where the primary and secondary sites are far away from each other, or the network bandwidth is limited. For HyperReplication/A, the write latency of foreground applications is independent of the distance between the primary and secondary sites.

5.3.3 HyperReplication/A for File

HyperReplication/A supports the long-distance data disaster recovery of file systems. It replicates all content of a primary file system to the secondary file system, implementing remote disaster recovery across data centers and minimizing the performance deterioration caused by remote data transmission. HyperReplication/A also applies to file systems within a storage system for local data disaster recovery, data backup, and data migration.

HyperReplication/A implements data replication based on the file system object layer. It periodically synchronizes data between primary and secondary file systems. All data changes to the primary file system since last synchronization will be synchronized to the secondary file system.

Working Principle

- Object layer-based replication

  HyperReplication/A implements data replication based on the object layer. The files, directories, and file properties of file systems consist of objects. Object layer-based replication copies objects from the primary file system to the secondary file system without considering complex file-level information, such as dependency between files and directories, and file operations, simplifying replication.

- Periodical replication based on ROW

  HyperReplication/A implements data replication based on ROW snapshots.

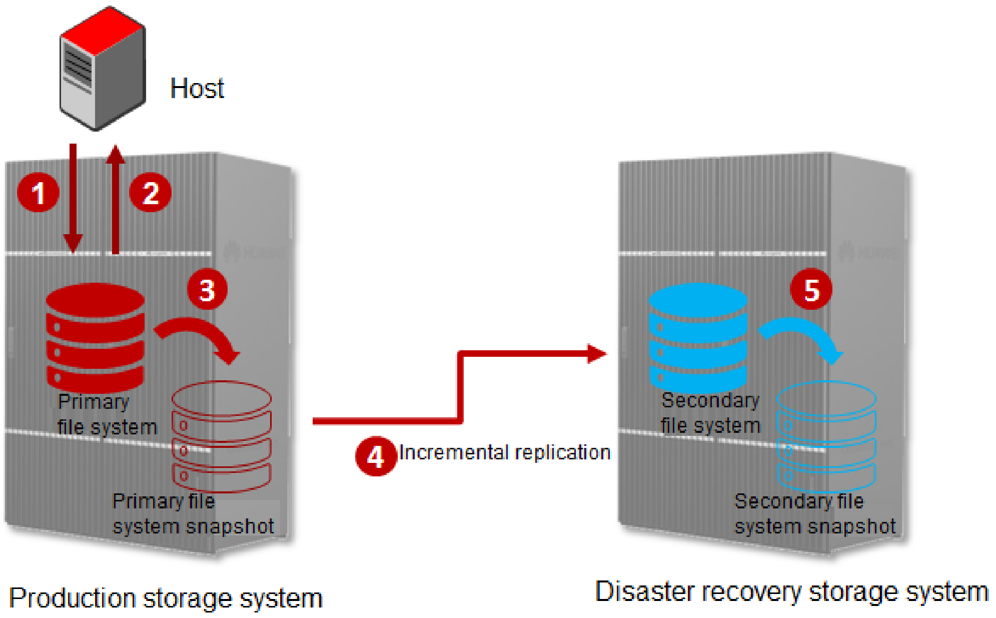
  - Periodical replication improves the replication efficiency and bandwidth utilization. In a replication period, if the host repeatedly writes data with the same address, for example, the data in the same location of a file is repeatedly modified, the data written last is replicated.

- File systems and their snapshots employ ROW to process data writes. No matter whether a file system has a snapshot, data is written to the new address space. The service performance will not decrease even though snapshots are created. Therefore, HyperReplication/A imposes a slight impact on production service performance.

The written data is periodically replicated to the secondary file system in the background. Replication periods are defined by users. The addresses rather than the content of incremental data blocks in each period are recorded. In each replication period, the secondary file system is incomplete before all incremental data is completely transferred to the secondary file system.

After the replication period ends and the secondary file system becomes a data consistency point, a snapshot is created for the secondary file system. If the next periodical replication is interrupted because the production center malfunctions or the link is down, HyperReplication/A can restore the secondary file system data to the last snapshot point to obtain consistent data.

Figure 5-7 Working principle of HyperReplication/A for File


Host
1
2
3
Primary file system
Primary file system snapshot
4
Incremental replication
5
Secondary file system
Secondary file system snapshot
Production storage system
Disaster recovery storage system

2. The production storage system receives a write request from a production host.
3. The production storage system writes the new data to the primary file system and immediately sends a write acknowledgement to the host.
4. When a replication period starts, HyperReplication/A creates a snapshot for the primary file system.
5. The production storage system reads and replicates snapshot data to the secondary file system based on the incremental information since the last synchronization.
6. After the incremental replication is complete, the content of the secondary file system is the same as the snapshot of the primary file system. The secondary file system becomes the data consistency point.

Technical Highlights

- Splitting and incremental resynchronization

  If you want to suspend data replication from the primary file system to the secondary file system, you can split the remote replication pair. For HyperReplication/A, a splitting will stop the ongoing replication process and later periodical replication.

  After the splitting, if the host writes new data, the incremental information will be recorded. You can start a resynchronization session after the splitting. During the resynchronization, only incremental data is replicated.

  Splitting applies to device maintenance scenarios, such as storage array upgrade and replication link change. In such scenarios, splitting can reduce the number of concurrent tasks so that the system becomes more reliable. The replication tasks will be resumed or restarted after the maintenance.

- Automatic recovery

  If data replication from the primary file system to the secondary file system is interrupted due to a fault, such as a link down fault, remote replication enters the interrupted state. If the host writes new data during the interruption period, the incremental information will be recorded. After the fault is rectified, remote replication is automatically recovered, and an incremental resynchronization is automatically implemented.

- Readable and writable secondary file system and incremental failback

  Normally, a secondary file system is readable but not writable. When accessing the secondary file system, the host reads data on snapshots generated in the last backup. After the next backup is completed, the host reads data on the new snapshots.

  Readable and writable secondary file system applies to the scenarios in which backup data must be accessed during replication.

  You can set a secondary file system to readable and writable when the following conditions are met:

  - The initial synchronization has been implemented. For HyperReplication/A, data on the secondary file system is in the complete state after the initial synchronization.
  - The remote replication pair is in the split or interrupted state.

  If data is being replicated from the primary file system to the secondary file system (the data is inconsistent on the primary and secondary file systems) when you set the secondary file system to readable and writable, HyperReplication/A restores data in the secondary file system to the last snapshot point.

  After the secondary file system is set to readable and writable, HyperReplication/A records the incremental information about data that the host writes to the secondary file system for subsequent incremental resynchronization. After replication recovery, you can replicate incremental data from the primary file system to the secondary file system or from the secondary file system to the primary file system (a primary/secondary switchover is required before synchronization). Before a replication session starts, HyperReplication/A restores target end data to a snapshot point in time when the data is consistent with source end data at a certain snapshot point in time and then performs an incremental resynchronization from the source end to the target end.

  Readable and writable secondary file systems are commonly used in disaster recovery scenarios.

- Primary/Secondary switchover

  Primary/secondary switchover exchanges the roles of the primary and secondary file systems. The roles determine the data replication direction. Data is replicated from the primary file system to the secondary file system.

  Primary/secondary switchover is commonly used in failback in disaster recovery.

- Quick response to host I/Os

  All I/Os generated during file system asynchronous remote replication are processed in the background. A write success acknowledgement is returned immediately after host data is written to the cache. Incremental information is recorded and snapshots are created only when data is flushed from cache to disks. Therefore, host I/Os can be quickly responded.

5.4 HyperMetro

HyperMetro, an array-level active-active technology provided by OceanStor 18000 V5 series, enables two storage systems to work in active-active mode in two locations within 100 km from each other, such as in the same equipment room or in the same city.

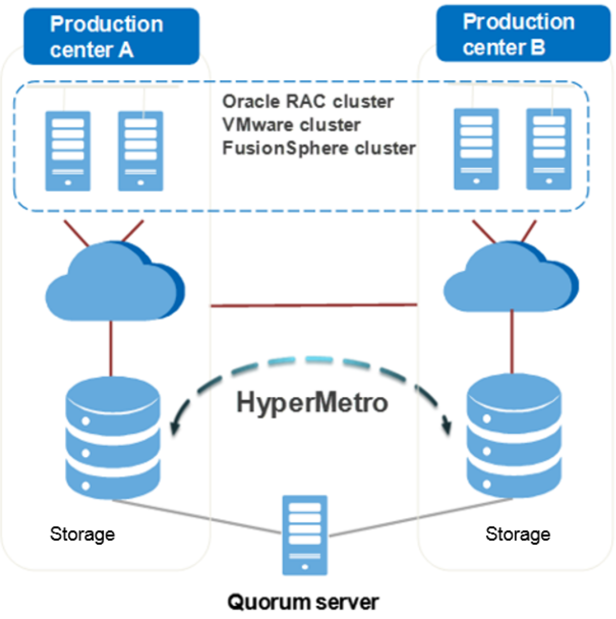
OceanStor 18000 V5 series supports both HyperMetro for Block and HyperMetro for File.

5.4.1 HyperMetro for Block

HyperMetro allows two LUNs from two storage arrays to maintain real-time data consistency and to be accessible to hosts. If one storage array fails, hosts automatically choose the path to the other storage array for service access. If the links between the storage arrays are interrupted, a quorum server deployed at a third location determines which storage array continues providing services.

HyperMetro supports both Fibre Channel and IP networking (GE/10GE).

Figure 5-8 Architecture of HyperMetro for Block



Technical Highlights

- Gateway-free active-active solution

Simple networking makes deployment easy. The gateway-free design improves reliability and performance because there is one less possible failure point and the 0.5 ms to 1 ms latency caused by a gateway is avoided.

- Active-active mode

  Storage arrays in two data centers are accessible to hosts, implementing load balancing across data centers.

- Site access optimization

  UltraPath is optimized specifically for active-active scenarios. It can identify region information to reduce cross-site access, reducing latency. UltraPath can read data from the local or remote storage array. However, when the local storage array is working properly, UltraPath preferentially reads data from and writes data to the local storage array, preventing data read and write across data centers.

- FastWrite

  In a common SCSI write process, a write request goes back and forth between two data centers twice to complete two interactions, namely Write Alloc and Write Data. FastWrite optimizes the storage transmission protocol and reserves cache space on the destination array for receiving write requests. Write Alloc is omitted and only one interaction is required. FastWrite halves the time required for data synchronization between two arrays, improving the overall performance of the HyperMetro solution.

- Service granularity-based arbitration

  If links between two sites fail, HyperMetro can enable some services to run preferentially in data center A and others in data center B based on service configurations. Compared with traditional arbitration where only one data center provides services, HyperMetro improves resource usage of hosts and storage systems and balances service loads. Service granularity-based arbitration is implemented based on LUNs or consistency groups.

- Automatic link quality adaptation

  If multiple links exist between two data centers, HyperMetro automatically balances loads among links based on the quality of each link. The system dynamically monitors link quality and adjusts the load ratio between links to minimize the retransmission rate and improve network performance.

- Compatibility with other features

  HyperMetro can work with SmartThin, SmartTier, SmartQoS, and SmartCache. HyperMetro can enable heterogeneous LUNs managed by the SmartVirtualization feature to work in A/A mode. HyperMetro can work with HyperSnap, HyperClone, HyperMirror, and HyperReplication to form a more complex, advanced data protection solution, such as the Disaster Recovery Data Center Solution (Geo-Redundant Mode), which uses local A/A and remote replication.

- Dual quorum servers

  HyperMetro supports dual quorum servers. If one quorum server fails, its services are seamlessly switched to the other, preventing the single point of failure (SPOF) and improving reliability of the HyperMetro solution.

5.4.2 HyperMetro for File

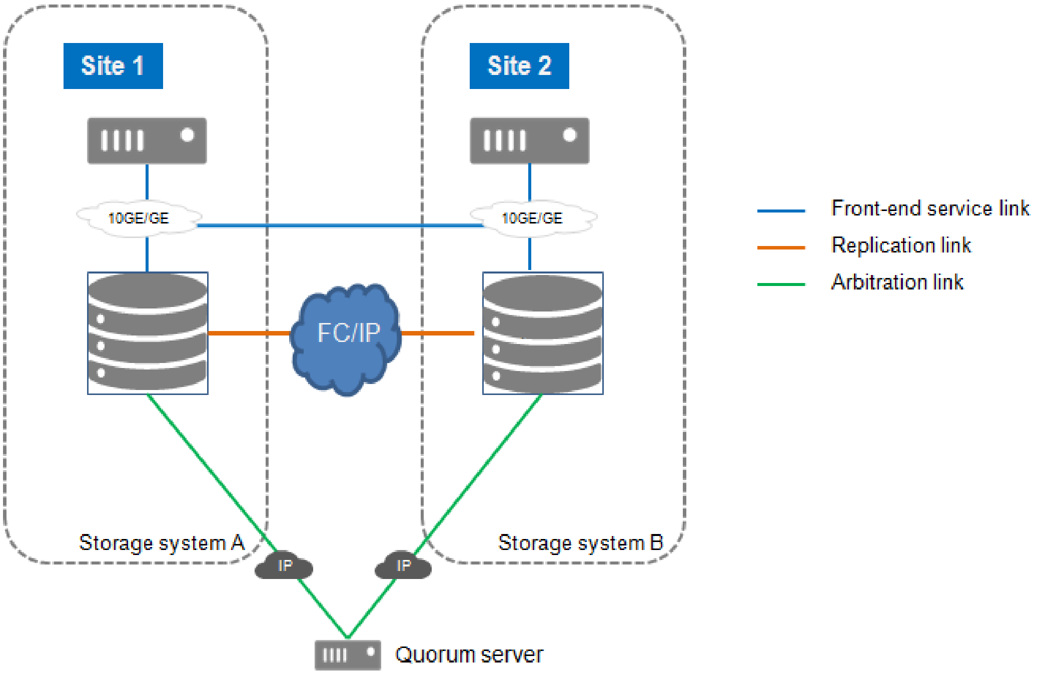
HyperMetro enables hosts to virtualize file systems on two storage system as a single file system on a single storage system, and keeps data in both file systems consistent. Data is read from or written into the primary storage system, and is synchronized to the secondary storage system in real time. If the primary storage system fails, HyperMetro switches services by vStore to the secondary storage system, without any data loss or interruption of applications.

HyperMetro provides the following benefits:

- High availability with geographic protection
- Easy management
- Minimal risk of data loss, reduced system downtime, and quick disaster recovery
- Negligible disruption to users and client applications

HyperMetro supports both Fibre Channel and IP networking (GE/10GE).

Figure 5-9 Architecture of HyperMetro for File



Technical Highlights

- Gateway-free solution

  With the gateway-free design, host I/O requests need not be forwarded by storage gateway, avoiding corresponding I/O forwarding latency and gateway failures, improving reliability. In addition, the design simplifies the cross-site high availability (HA) network, making maintenance easier.

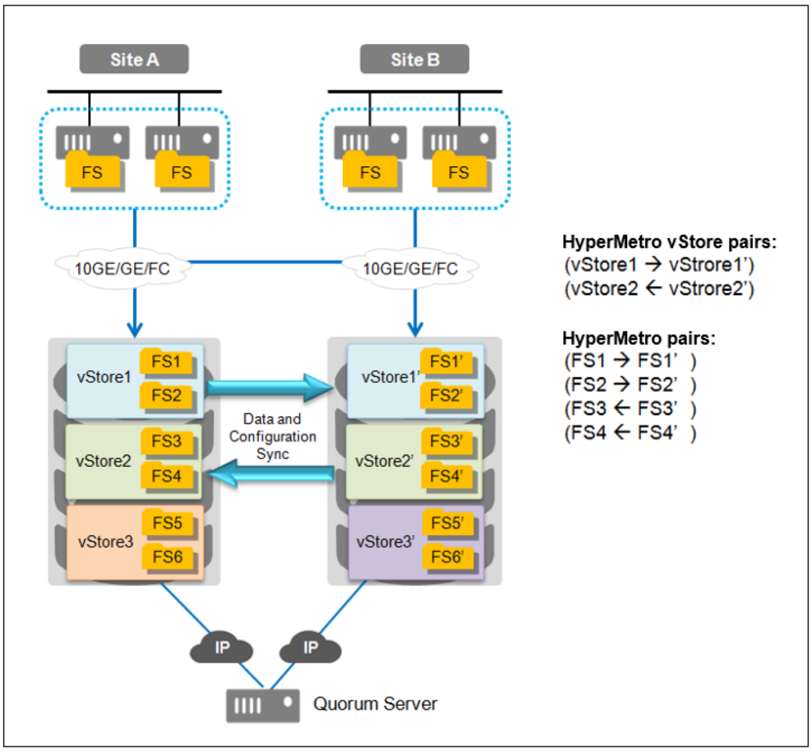
- Simple networking

  The data replication, configuration synchronization, and heartbeat detection links share the same network, simplifying the networking. Either IP or Fibre Channel links can be used between storage systems, making it possible that HyperMetro work on all-IP networks, improving cost-effectiveness.

- vStore-based HyperMetro

  Traditional cross-site HA solutions typically deploy cluster nodes at two sites to implement cross-site HA. These solutions, however, have limited flexibility in resource configuration and distribution. HyperMetro can establish a pair relationship between two vStores at different sites, implementing real-time mirroring of data and configurations. Each vStore pair has an independent arbitration result, providing true cross-site HA capabilities at the vStore level. HyperMetro enables applications to run more efficiently at two sites with better load balancing. A vStore pair includes a primary vStore and a secondary vStore. If either of the two storage systems in the HyperMetro solution fails or the links connecting them go down, HyperMetro implements arbitration on a per vStore pair basis. Paired vStores are mutually redundant, maintaining service continuity in the event of a storage system failure.

Figure 5-10 vStore-based HyperMetro architecture



- Automatic recovery

  If site A breaks down, site B becomes the primary site. After site A recovers, HyperMetro automatically initiates resynchronization. When resynchronization is complete, the HyperMetro pair returns to its normal state. If site B then breaks down, site A becomes the primary site again to maintain host services.

- Easy upgrade

  To use the HyperMetro feature, upgrade your storage system software to latest version and purchase the required feature license. You can establish a HyperMetro solution between the upgraded storage system and another storage system, without requiring extra data migration. Users are free to include HyperMetro in initial configurations or add it later as needed.

- FastWrite

  In a common SCSI write process, a write request goes back and forth between two data centers twice to complete two interactions, namely Write Alloc and Write Data. FastWrite optimizes the storage transmission protocol and reserves cache space on the destination array for receiving write requests. Write Alloc is omitted and only one interaction is required. FastWrite halves the time required for data synchronization between two arrays, improving the overall performance of the HyperMetro solution.

- Self-adaptation to link quality

If there are multiple links between two data centers, HyperMetro automatically implements load balancing among links based on quality. The system dynamically monitors link quality and adjusts the load ratio between links to minimize the retransmission rate and improve network performance.

- Compatibility with other features

  HyperMetro can be used with SmartThin, SmartQoS, and SmartCache. HyperMetro can work with HyperVault, HyperSnap, and HyperReplication to form a more complex, advanced data protection solution, such as the Disaster Recovery Data Center Solution (Geo-Redundant Mode), which uses HyperMetro and HyperReplication.

- Dual quorum servers

  HyperMetro supports dual quorum servers. When one quorum server fails, its services can be seamlessly switched to the other quorum server, preventing the single point of failure (SPOF) and improving reliability of the HyperMetro solution.

5.5 HyperVault

OceanStor 18000 V5 series provides an all-in-one backup feature called HyperVault to implement file system data backup and recovery within or between storage systems. HyperVault can work in either of the following mode:

- Local backup

  Data backup within a storage system. HyperVault works with HyperSnap to periodically back up a file system, generate backup copies, and retain the copies based on user-configured policies. By default, five backup copies are retained for a file system.

- Remote backup

  Data backup between storage systems. HyperVault works with HyperReplication to periodically back up a file system. The process is as follows:

1. A backup snapshot is created for the primary storage system.
2. The incremental data between the backup snapshot and its previous snapshot is synchronized to the secondary storage system.
3. After data is synchronized, a snapshot is created on the secondary storage system.

   By default, 35 snapshots can be retained on the backup storage system.

Technical Highlights

- High cost efficiency

  HyperVault can be seamlessly integrated into the primary storage system and provides data backup without requiring additional backup software. Huawei-developed storage management software, OceanStor DeviceManager, allows you to configure flexible backup policies and perform data backup efficiently.

- Fast data backup

  HyperVault works with HyperSnap to achieve second-level local data backup. For remote backup, the system performs full backup the first time, and then only backs up incremental data blocks. Compared with the backup software that backs up files each time, HyperVault provides faster data backup.

- Fast data recovery

  HyperVault uses the snapshot rollback technology to implement local data recovery, without requiring additional data resolution, achieving second-level data recovery.

Remote recovery can be used when local recovery cannot meet requirements. Remote recovery is incremental data recovery. Each copy of backup data is a logically full backup of service data. The backup data is saved in its original format and can be accessed immediately.

- Simple management

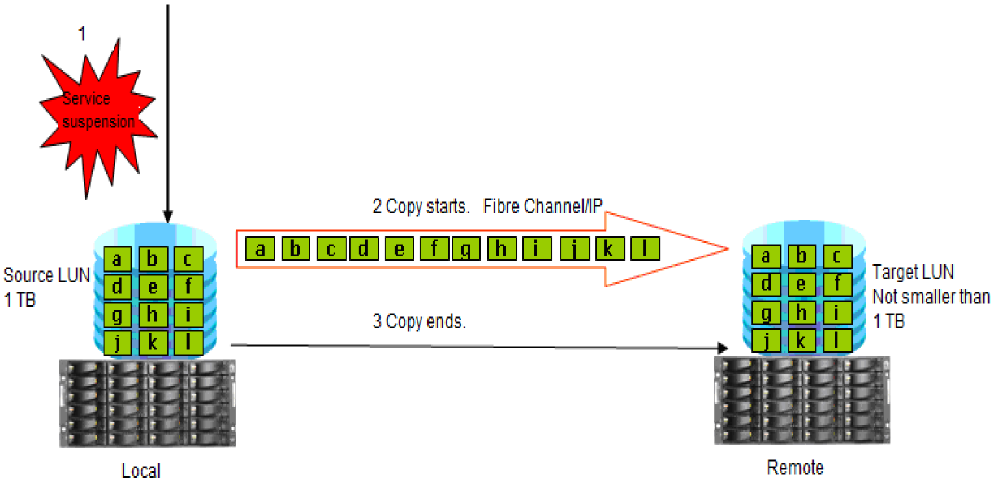
  Only one primary storage system and one backup storage system, and native management software, OceanStor DeviceManager are required. This mode is simpler and easier to manage than old network designs, which contain primary storage, backup software, and backup media.

5.6 HyperCopy

OceanStor 18000 V5 series uses HyperCopy to copy data from a source LUN to a target LUN within a storage system or between storage systems.

HyperCopy implements full copy, that is, copying all data from a source LUN to a target LUN. Figure 5-11 illustrates the working principle of HyperCopy.

Figure 5-11 HyperCopy working principle



2. A user suspends services to which HyerCopy is applied.

   This prevents services from being interrupted during full LUN copy.

3. A user triggers full LUN copy.

   Data can be copied to a target LUN over a Fibre Channel or IP link. The target LUN must have a capacity not less than that of the source LUN. Otherwise, data cannot be copied successfully.

   During the copy, the copy progress is displayed.

OceanStor 18000 V5 series can implement full LUN copy by reading snapshot volumes without interrupting services, ensuring a zero backup window.

Technical Highlights

- Multiple copy approaches

OceanStor 18000 V5 series supports LUN copy within one storage system and between storage systems. Data can be copied from the local/target storage system to the target/local storage system. One-to-many LUN copy is provided to generate multiple copies for a source LUN.

- Dynamic adjustment of the copy speed

  HyperCopy allows users to dynamically adjust the copy speed, thereby preventing LUN copy from affecting production services. When a storage system detects that the service load is heavy, it dynamically lowers the LUN copy speed to make system resources available to services. When the service load is light, the storage system dynamically increases the copy speed, mitigating service conflicts in peak hours.

- Support for third-party storage systems

  LUN copy can be implemented within OceanStor storage systems or between OceanStor storage systems and Huawei-certified third-party storage systems. Table 5-1 describes the storage systems supported by LUN copy.

Table 5-1 Storage systems supported by LUN copy

Storage System Where a Source LUN Resides	OceanStor Storage System Where a Target LUN Resides	Huawei-Certified Third-Party Storage System Where a Target LUN Resides
OceanStor storage system	Supported	Supported
Huawei-certified third-party storage system	Supported	N/A

- IP network-based LUN copy

  Regarding LUN copy between storage systems, most vendors in the industry support only Fibre Channel-based LUN copy. OceanStor 18000 V5 series supports both Fibre Channel-based and IP network–based LUN copy. Customers can flexibly choose one based on site requirements. Furthermore, with the popularization of IP networks, IP network–based LUN copy features low costs, easy deployment, and simple maintenance.

5.7 HyperMirror

OceanStor 18000 V5 series uses HyperMirror for volume mirroring.

HyperMirror creates two physical copies for a LUN. Space for each copy can be either from a local storage pool or an external LUN. Each copy has the same virtual storage capacity as its mirror LUN. When a server writes data to a mirror LUN, the storage system simultaneously writes the data to the LUN's copies. When a server reads data from a mirror LUN, the storage system reads data from one copy of the mirror LUN. Even if one mirror copy of a mirror LUN is temporarily unavailable (for example, when the storage system where the storage pool resides is unavailable), servers can still access the LUN. Then, the storage system records the LUN areas to which data has been written and synchronizes these areas after the mirror copy recovers.

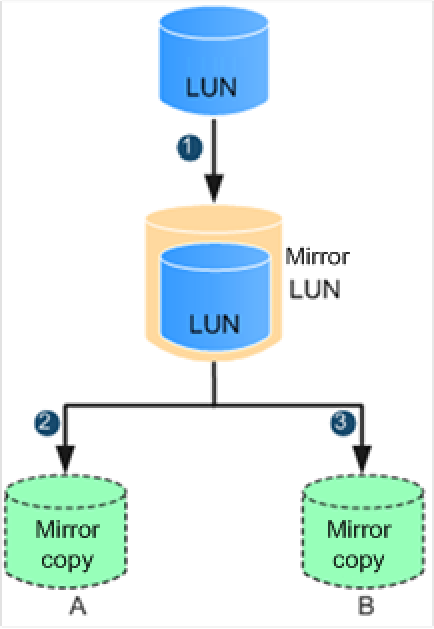
Working Principle

HyperMirror implementation involves mirror LUN creation, synchronization, and splitting.

Mirror LUN creation

Figure 5-12 shows the process for creating a mirror LUN.

Figure 5-12 Process for creating a mirror LUN



2. A user creates a mirror LUN for a local or external LUN. The mirror LUN has the same storage space, properties, and services as the source LUN. Host services are not interrupted during the creation.
3. Local mirror copy A is automatically generated during the mirror LUN creation. The storage space is swapped from the mirror LUN to mirror copy A. Mirror copy A synchronizes data from the mirror LUN.
4. A user creates mirror copy B for the mirror LUN. Mirror copy B copies data from mirror copy A. In doing so, the LUN with mirror copies A and B has the space mirroring function.

The following describes the process when a host sends an I/O request to the mirror LUN.

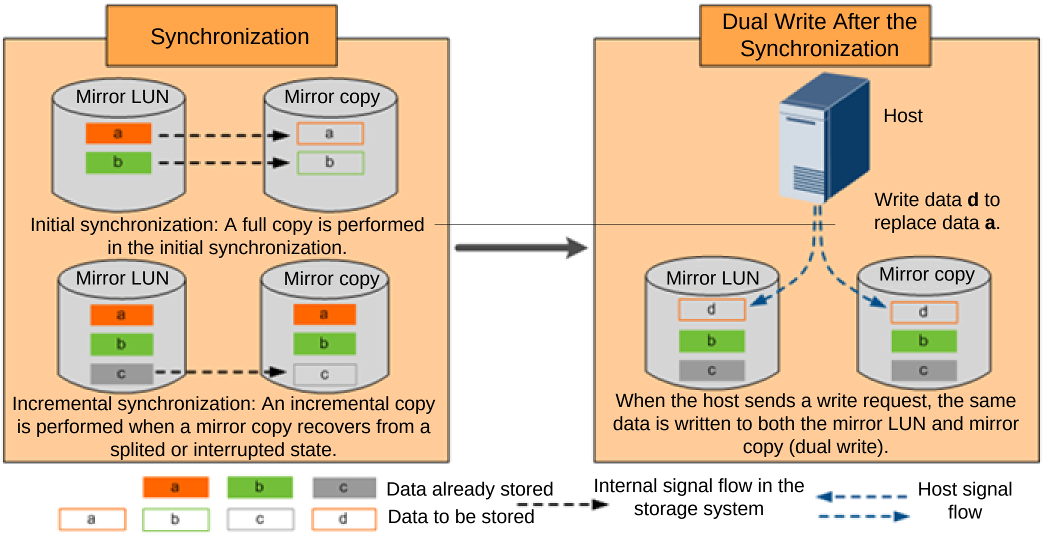
1 When a host sends a read request to the mirror LUN, the storage system reads data from the mirror LUN and its mirror copies in round-robin mode. If the mirror LUN or one mirror copy malfunctions, host services are not interrupted.

2 When a host sends a write request to the mirror LUN, the storage system writes data to the mirror LUN and its mirror copies in dual-write mode.

Synchronization

Figure 5-13 illustrates the synchronization process. When a mirror copy recovers from a fault or data on a mirror copy becomes complete, it copies incremental data from the other mirror copy, ensuring data consistency between mirror copies.

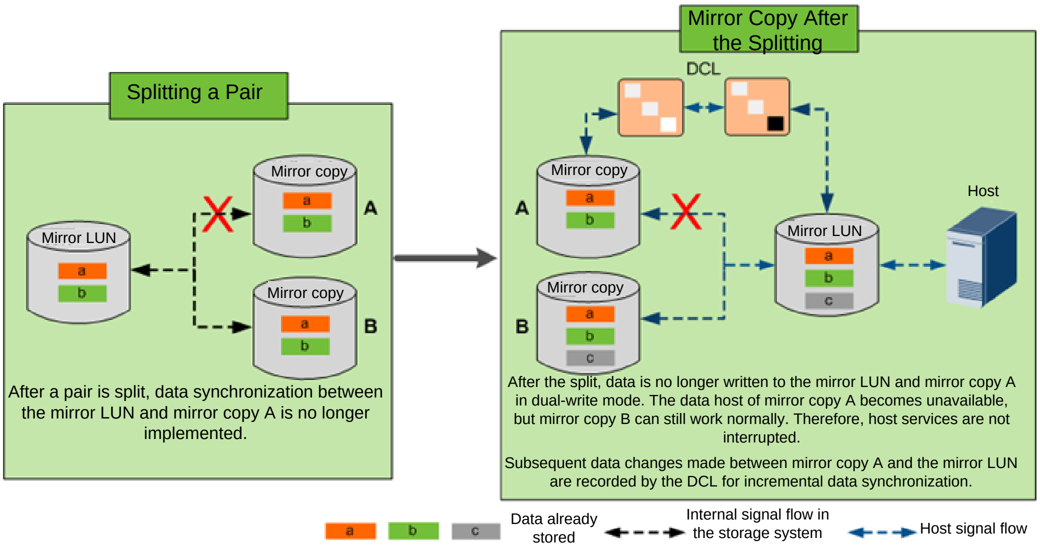
Figure 5-13 Synchronization process


Synchronization
Mirror LUN
Mirror copy
a
b
Initial synchronization: A full copy is performed in the initial synchronization.
Mirror LUN
Mirror copy
c
Incremental synchronization: An incremental copy is performed when a mirror copy recovers from a splited or interrupted state.
Dual Write After the Synchronization
Host
Write data **d** to replace data **a**.
Mirror LUN
Mirror copy
d
When the host sends a write request, the same data is written to both the mirror LUN and mirror copy (dual write).
Data already stored
Data to be stored
Internal signal flow in the storage system
Host signal flow

Splitting

A mirror copy can be split from its mirror LUN. After the mirror copy is split, the mirror LUN cannot perform mirroring on the mirror copy. Subsequent data changes made between the mirror copy and the mirror LUN are recorded by the DCL for incremental data synchronization when the mirroring relationship is restored.

Figure 5-14 Splitting


Splitting a Pair
Mirror LUN
Mirror copy
A
B
a
b
After a pair is split, data synchronization between the mirror LUN and mirror copy A is no longer implemented.
Mirror Copy After the Splitting
DCL
Host
Mirror copy
Mirror LUN
c
After the split, data is no longer written to the mirror LUN and mirror copy A in dual-write mode. The data host of mirror copy A becomes unavailable, but mirror copy B can still work normally. Therefore, host services are not interrupted.
Subsequent data changes made between mirror copy A and the mirror LUN are recorded by the DCL for incremental data synchronization.
Data already stored
Internal signal flow in the storage system
Host signal flow

Technical Highlights

- High data reliability within a storage system

  HyperMirror creates two independent mirror copies for a LUN. If one mirror copy malfunctions, host services are not interrupted, significantly enhancing data reliability.

- Robust data reliability of heterogeneous storage systems

  When SmartVirtualization is used to take over LUNs of heterogeneous storage systems, HyperMirror is employed to create a mirror LUN and local mirror copies for each heterogeneous LUN. Services will not be interrupted when heterogeneous storage systems are unstable or their links are down.

- Little impact on host performance

  Mirror copies generated by HyperMirror reside in the cache of their LUN and the concurrent write and round-robin read technologies are implemented between mirror spaces. In this way, host service performance is not affected.

- Ensured host service continuity

  HyperMirror allows mirror copies to be created online for ongoing LUNs. In this way, host services are unaware of changes made to LUN data space.

5.8 HyperLock

As information explosively grows, increasing importance has been attached to secure access and application. For laws and regulations compliance, important data such as case documents of courts, medical records, and financial documents can only be read but cannot be written within a specific period. Therefore, measures must be taken to prevent such data from being tampered with. In the storage industry, Write Once Read Many (WORM) is the most common method used to archive and back up data, ensure secure data access, and prevent data tampering.

The WORM feature developed by Huawei is called HyperLock. A file protected by WORM can enter the read-only state immediately after data is written to it. In the read-only state, the file can be read, but cannot be deleted, modified, or renamed. WORM can prevent data from being tampered with, meeting data security requirements of enterprises and organizations.

File systems for which WORM has been configured are called WORM file systems and can only be configured by administrators. There are two WORM modes:

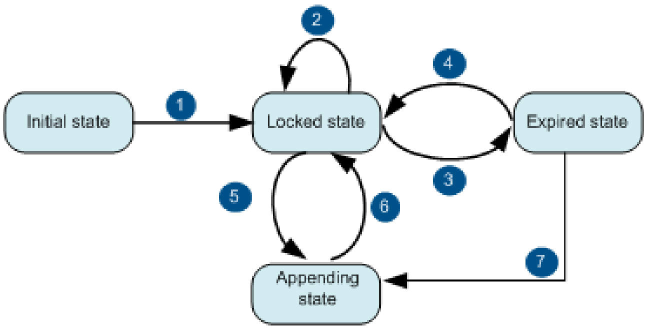
- Regulatory Compliance WORM (WORM-C for short): applies to archive scenarios where data protection mechanisms are implemented for laws and regulations compliance.
- Enterprise WORM (WORM-E): is mainly used by enterprises to implement internal control.

Working Principle

With WORM, data can be written to files once only, and cannot be rewritten, modified, deleted, or renamed. If a common file system is protected by WORM, files in the WORM file system can be read only within the protection period. After WORM file systems are created, you need to map them to application servers using the NFS or CIFS protocol.

WORM enables files in a WORM file system to be shifted between initial state, locked state, appending state, and expired state, preventing important data from being falsely or maliciously tampered with in a specified period. Figure 5-15 shows how a file shifts from one state to another.

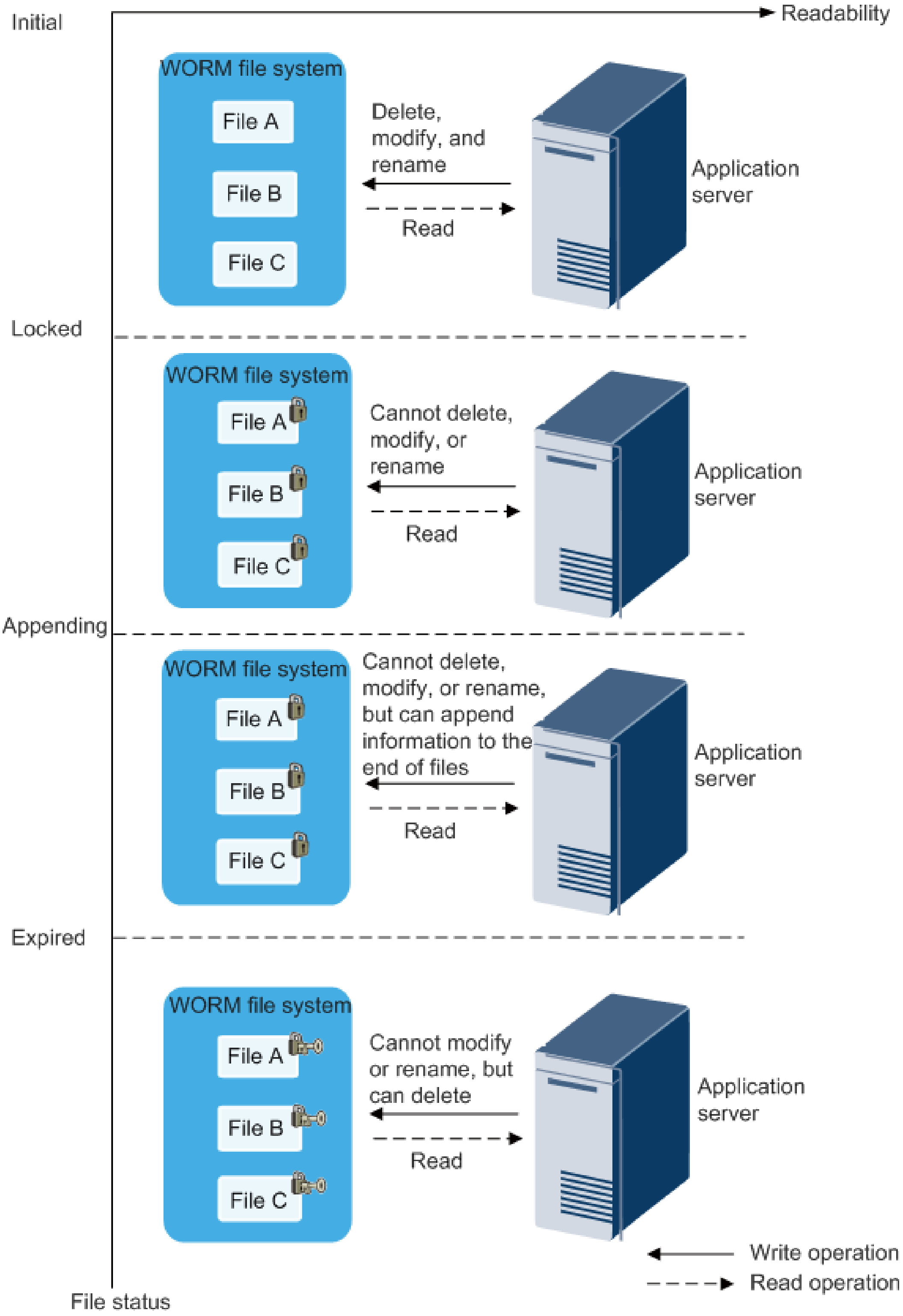
Figure 5-15 File state shifting



2. Initial to locked: A file can be shifted from the initial state to the locked state using the following methods:
   - If the automatic lock mode is enabled, the file automatically enters the locked state upon the expiration of a specific period after a change.
   - You can manually set the file to the locked state. Before locking the file, you can specify a protection period for the file or use the default protection period.
3. Locked to locked: In the locked state, you can manually extend the protection periods of files. Protection periods cannot be shortened.
4. Locked to expired: After the WORM file system compliance clock reaches the file overdue time, the file shifts from the locked state to the expired state.
5. Expired to locked: You can extend the protection periods of a file to shift it from the expired state to the locked state.
6. Locked to appending: You can delete the read-only permission of a file to shift the file from the locked state to the appending state.
7. Appending to locked: You can manually set the file in the appending state to the locked state to ensure that the file cannot be modified.
8. Expired to appending: You can manually set the file in the expired state to the appending state.

You can save files to WORM file systems and set the WORM properties of the files to the locked state based on service requirements. Figure 5-16 shows the reads and writes of files in all states in a WORM file system.

Figure 5-16 Read and write of files in a WORM file system



5.9 3DC

OceanStor 18000 V5 series can be used in various SAN and NAS disaster recovery solutions.

SAN 3DC solutions deployed on:

- Cascading/Parallel network equipped with HyperMetro + HyperReplication/S
- Cascading/Parallel network equipped with HyperMetro + HyperReplication/A
- Ring network equipped with HyperMetro + HyperReplication/A
- Cascading/Parallel network equipped with HyperReplication/S + HyperReplication/A
- Cascading/Parallel network equipped with HyperReplication/A + HyperReplication/A
- Ring network equipped with HyperReplication/S + HyperReplication/A

NAS 3DC solutions deployed on:

- Cascading/Parallel network equipped with HyperMetro + HyperReplication/A
- Cascading/Parallel network equipped with HyperMetro + HyperVault
- Cascading/Parallel network equipped with HyperReplication/A + HyperReplication/A

Two data centers equipped with HyperMetro or HyperReplication/S + HyperReplication/A can be flexibly expanded to three data centers without requiring external gateways.

For details about 3DC solutions that OceanStor 18000 V5 series supports, visit http://storage.huawei.com/en/index.html.

6 Best Practices

For best practices of OceanStor 18000 V5 series, visit http://storage.huawei.com/en/html/OceanStor_V5_en.html.

A Appendix

A.1 More Information

You can obtain more information about OceanStor 18000 V5 series at the following site:

http://e.huawei.com/en/products/cloud-computing-dc/storage/massive-storage/18500-18800-v5

You can also visit our official website to get more information about Huawei storage:

http://e.huawei.com/en/products/cloud-computing-dc/storage

For after-sales support, visit our technical support website:

http://support.huawei.com/enterprise/en

For pre-sales support, visit the following website:

http://e.huawei.com/en/how-to-buy/contact-us

You can also contact your local Huawei office:

http://e.huawei.com/en/branch-office

A.2 Feedback

Huawei welcomes your suggestions for improving our documentation. If you have comments, send your feedback to storagedoc@huawei.com.

Your suggestions will be seriously considered and we will make necessary changes to the document in the next release.

A.3 Acronyms and Abbreviations

Table A-1 Acronyms and abbreviations

Acronym and Abbreviation	Full Spelling
AK	Authentication Key
BBU	Backup Battery Unit

CK	Chunk
CKG	Chunk group
CIFS	Common Internet File System
COW	Copy-on-write
DCL	Data Change Log
DEK	Data Encryption Key
DIX	Data Integrity Extensions
DoD	Department of Defense
eDevLUN	External device LUN
HA	High availability
KMS	Key Manager Server
LRU	Least Recently Used
NDMP	Network Data Management Protocol
NFS	Network File System
ODX	Offload Data Transfer
OLAP	Online Analytical Processing
OLTP	Online Transaction Processing
RAID	Redundant Array of Independent Disks
RPO	Recovery Point Objective
ROW	Redirect-on-write
SED	Self-encrypting drive
SPOF	Single point of failure
SRM	Site Recovery Manager
SCOM	System Center Operations Manager
SCVMM	System Center Virtual Machine Manager
TCO	Total cost of ownership
VAAI	VMware vStorage APIs for Array Integration
VASA	vStorage APIs for Software Awareness
WWN	World Wide Name
WORM	Write Once Read Many