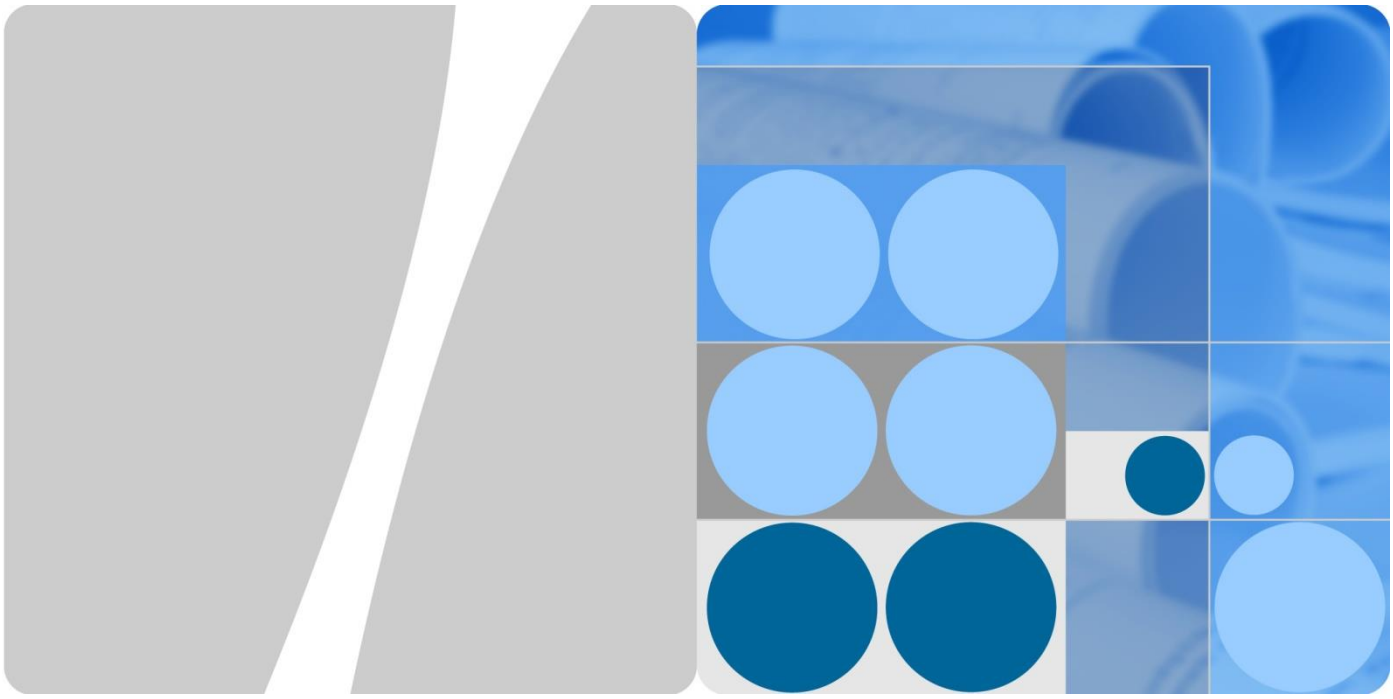


Huawei OceanStor 2600 V3 Storage Systems

Technical White Paper

Issue 01

Date 2016-04-30

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

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

Change History

Date	Version	Description	Prepared By
2016-04-30	1.0	Updated the document based on Huawei OceanStor V3 converged storage system technical white papers.	Liu Kunpeng/00254944

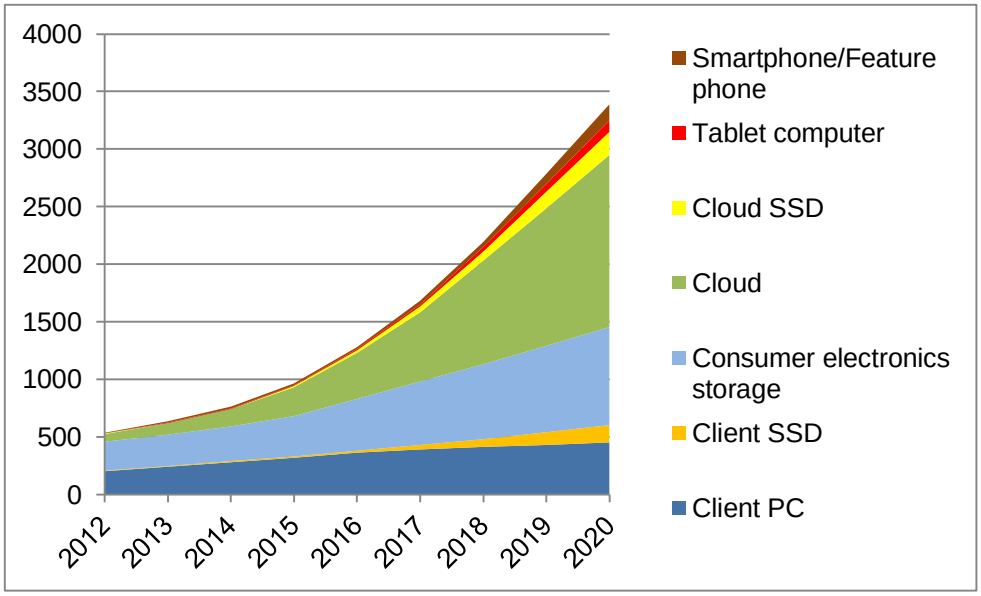
Contents

1 Overview

Evolving from mainframe servers to midrange computers, PCs, and desktop Internet, the information technology (IT) is penetrating into all walks of life. Nowadays, we are embracing the mobile Internet era. The change of application environments hastens data explosion. According to Gartner's statistics, about 2.6 EB of data was generated around the world in the era of midrange computers and 15.8 EB of data when PCs were popular. In the era of desktop Internet, the amount of data was almost quadrupled, reaching 54.5 EB. Up to 1800 EB of data may be generated in the era of mobile Internet. The skyrocketing amount of data not only requires super-large storage capacities but also imposes demanding requirements on other features of storage products.

Since data sources are increasingly diversified, clouds will gradually become the largest data sources, replacing PCs and consumer electronics (CE). The following figure shows predicted rankings of data sources.

Figure 1-1 Predicted rankings of data sources



Since data sources are changing constantly, data types change accordingly. Although the amount of critical service data, such as databases, increases continuously, it accounts for a decreasing percentage of the total data amount; whereas enterprise office data, such as emails

and large media files, once accounted for the highest percentage of the total data amount. In recent years, since the amount of personal data increases sharply, media and entertainment data replaces enterprise office data as the largest data sources. In 1993, critical service data and enterprise office data accounted for 50% of the total data amount respectively, and the amount of personal data could be ignored. In 2002, 70% of data was enterprise office data, and 20% was critical service data. Since 2010, personal data accounts for 50% of the total data volume, whereas enterprise office data accounts for 40%, and critical service data accounts for only 10%.

Different types of data from diversified sources have different requirements on the performance, reliability, and costs of storage media. Critical service data requires high-performance and robust-reliability storage devices, whereas personal entertainment data requires inexpensive storage devices. The reality is that critical service data and personal entertainment data usually need to be stored in a single set of storage device. Such contradicting requirements impose new challenges. To keep with IT development, next-generation unified storage products must have:

2. Integrated, simple, intelligent, and cost-effective system architecture
3. High flexibility, meeting diverse storage needs
4. Agile data planning and management
5. Rich and practical functions

2 On-Demand, Simple & Efficient, Secure & Reliable

Huawei OceanStor 2600 V3 unified storage systems are brand-new storage systems designed for enterprise-class applications. Leveraging a storage operating system built on a cloud-oriented architecture, a powerful new hardware platform, and suites of intelligent management software, the OceanStor V3 delivers industry-leading functions, performance, efficiency, reliability, and ease-of-use. It provides data storage for applications such as large-database Online Transaction Processing (OLTP)/Online Analytical Processing (OLAP), file sharing, and cloud computing, and can be widely applied to industries ranging from government, finance, telecommunication, to energy, and Media & Entertainment (M&E). The OceanStor 2600 V3 storage systems can provide a wide range of efficient and flexible backup and disaster recovery solutions to ensure business continuity and data security, delivering excellent storage services.

OceanStor 2000 V3 storage systems include OceanStor 2200 V3 and OceanStor 2600 V3. OceanStor 2200 V3 is entry-level SAN storage and features ease of use while OceanStor 2600 V3 is entry-level converged storage and meets flexible SAN/NAS integration requirements.

On-demand: providing customers with on-demand storage services

- Thanks to SmartVirtualization, the V3 series entry-level storage systems can efficiently manage storage systems from other mainstream vendors and unify resource pools for central and flexible resource allocation.
- SmartMigration migrates host services from a source LUN to a target LUN without interrupting these services and then enables the target LUN to take over services from the source LUN without being noticed by the hosts. After the service migration is complete, all service-related data has been replicated from the source LUN to the target LUN.
- SAN and NAS are converged on one platform that provides file and block access services, enabling flexible configurations and meeting diverse application requirements.

Simple & efficient: simplifying management and improving efficiency

- SmartTier automatically analyzes data access frequencies per unit time and migrates data to disks of different performance levels based on the analysis result. (High-performance disks store most frequently accessed data, performance disks store less frequently accessed data, and large-capacity disks store seldom accessed data.) In this way, the optimal overall performance is achieved, and the IOPS cost is reduced.
- SmartThin allocates storage space on demand rather than pre-allocating all storage space at the initial stage. It is more cost-effective because customers can start business with a few disks and add disks based on site requirements. In this way, the initial purchase cost and total cost of ownership (TCO) are reduced.

- SmartDedupe and SmartCompression deduplicate and compress data before data storage, reducing storage space for data storage, lowering the storage cost per GB, and improving the data storage efficiency.

  Secure & reliable: establishing reliable resource pools based on a virtualization architecture

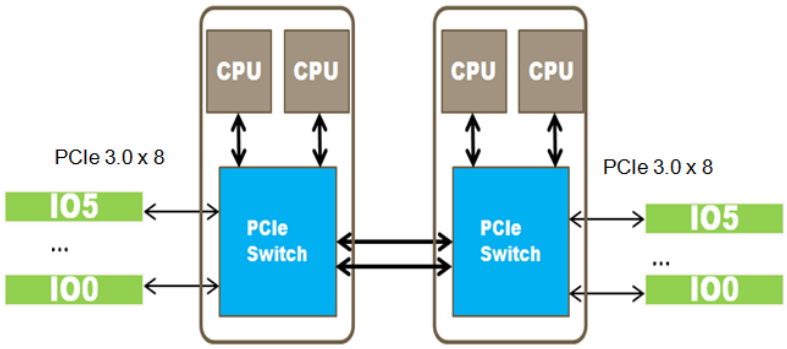
- SmartQoS categorizes service data based on data characteristics (each category represents a type of application) and sets a priority and performance objective for each category. In this way, resources are allocated to services properly, fully utilizing system resources.
- The core value of SmartPartition is to ensure the performance of critical applications by partitioning core system resources. Users can configure cache partitions of different sizes. The 2600 V3 storage systems ensure the number of cache partitions occupied by service applications. Based on the actual service condition, the 2600 V3 storage systems dynamically adjust the number of concurrent access requests from hosts to different cache partitions, ensuring the service application performance of each partition.
- HyperMetro enables storage systems to be deployed in gateway-free active-active mode, ensuring business continuity across data centers.

3 Architecture

The OceanStor 2600 V3 storage systems employ an inter-engine IP interconnection architecture to achieve scale-out. Inter-engine service switching and mirroring channels employ a 10GE network. The OceanStor 2600 V3 storage system supports a maximum of four engines, each providing two controllers. In other words, the OceanStor 2600 V3 supports a maximum of eight controllers. Each controller is connected to two switching planes using 10GE switches for data forwarding.

The two controllers in an engine are interconnected based on PCIe 3.0. That is, 8-lane PCIe 3.0 forms high-speed mirroring channels between the two controllers, as shown in the following figure.

Figure 3-1 Two controllers in an engine interconnected based on PCIe 3.0



Switching channels support direct-connection networks and switch-connection networks. On a switch-connection network as shown in Figure 3-2, multiple controller enclosures are connected to two data switches over 10GE to exchange data. Such a cluster supports a maximum of eight controllers. On a direct-connection network as shown in Figure 4, two network ports of one network adapter are connected to the controllers in another engine. Such a cluster supports a maximum of four controllers. Dual switching links are used to ensure the redundancy of cluster data exchange networks. IP interconnection reserves sufficient space for future cluster expansion, improving cluster scalability. Data exchange between controllers and mirroring channels employ the all-PCIe interconnection architecture, accelerating data exchange.

Figure 3-2 Switch-connection network employing a multi-controller architecture

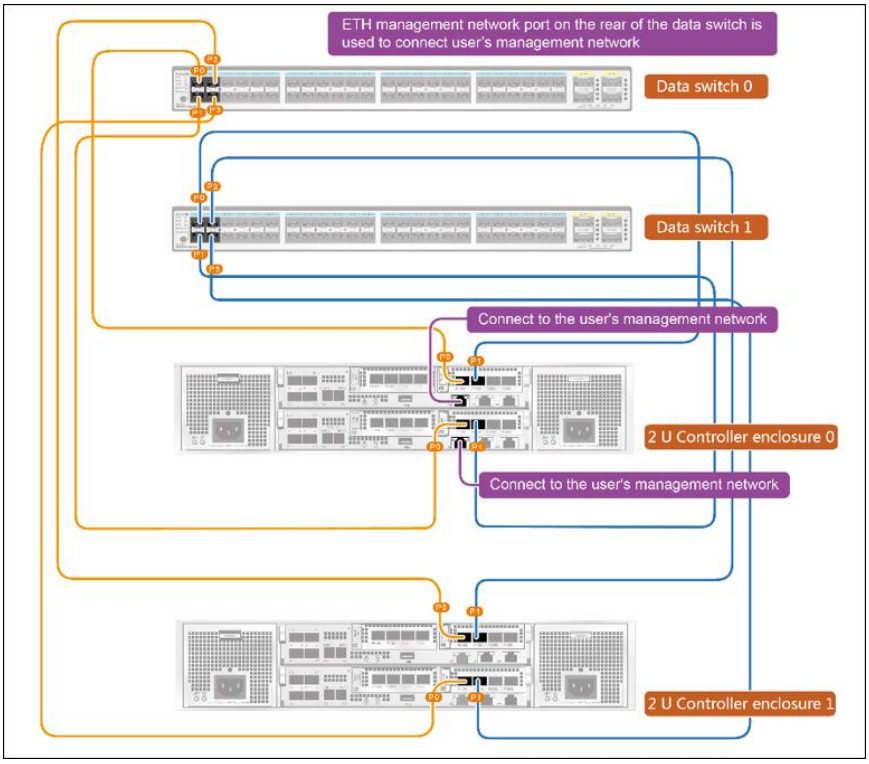
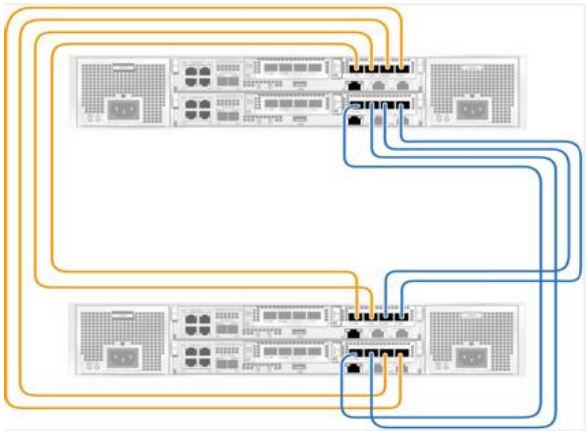


Figure 3-3 Direct-connection network employing a four-controller architecture



Four controller enclosures are connected to two data switches over 10GE to exchange data, ensuring the redundancy of cluster data exchange networks. IP interconnection reserves sufficient space for future cluster expansion, improving cluster scalability. Data exchange between controllers and mirroring channels employ the all-PCIe interconnection architecture, accelerating data exchange.

The OceanStor V3 series storage systems boast all-PCIe 3.0 interconnection, back-end SAS 3.0, and high-speed channels and powerful computing capabilities, meeting increasingly demanding performance requirements. In addition, the OceanStor 2600 V3 storage systems

employ the single point of failure prevention design and scale-out to deliver high reliability and flexible scalability under a tight budget.

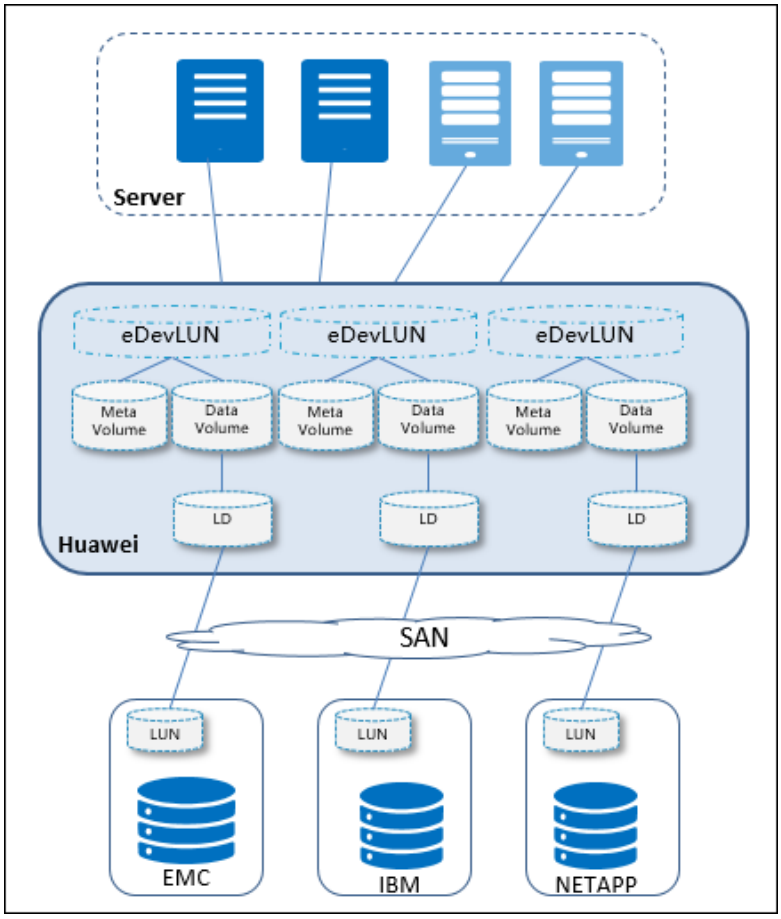
4 SmartVirtualization

The OceanStor 2600 V3 storage systems aim at providing rich virtualization functions for heterogeneous storage systems of customers. The heterogeneous takeover function reduces complexity in managing heterogeneous storage systems and improves LUN performance.

- The heterogeneous online migration function allows data to be smoothly migrated among LUNs of heterogeneous storage systems without interrupting services.
- The heterogeneous remote replication function implements DR for LUNs of heterogeneous storage systems.
- The heterogeneous snapshot function implements rapid backup for LUNs of heterogeneous storage systems. The heterogeneous virtualization feature provided by the V3 converged storage systems is called SmartVirtualization.

SmartVirtualization uses LUNs mapped from heterogeneous storage systems to the local storage system as logical disks (LDs) that can provide storage space for the local storage system and create eDevLUNs that can be mapped to the host on LDs. LDs provide data storage space for data volumes, and the local storage system provides storage space for meta volumes of eDevLUNs. SmartVirtualization ensures the data integrity of external LUNs.

Figure 4-1 SmartVirtualization



eDevLUNs and local LUNs have the same properties. For this reason, SmartMigration, HyperReplication/S, HyperReplication/A, and HyperSnap are used to provide non-disruptive migration, synchronous remote replication, asynchronous remote replication, and snapshot functions respectively for LUNs of heterogeneous storage systems. SmartQoS, SmartPartition, and write-back cache are used to improve the LUN performance of heterogeneous storage systems.

SmartVirtualization applies to:

- **Heterogeneous array takeover**

  As users' data centers develop, storage systems in the data centers may come from different vendors. How to efficiently manage and apply storage systems from different vendors is a challenge that storage administrators must tackle. Storage administrators can leverage the takeover function of SmartVirtualization to simplify heterogeneous array management. They need only to manage Huawei storage systems, and their workloads are remarkably reduced. In such a scenario, SmartVirtualization simplifies system management.

- **Heterogeneous data migration**

  A large number of heterogeneous storage systems whose warranty periods are about to expire or whose performance cannot meet service requirements may exist in a customer's data center. After purchasing the OceanStor 2600 V3 storage systems, the customer wants to migrate services from the existing storage systems to the new storage systems. The customer can leverage the online migration function of SmartMigration to migrate

data from LUNs of heterogeneous storage systems to the new storage systems. The migration process has no adverse impact on ongoing host services, but the LUNs must be taken over before the migration. In such a scenario, SmartVirtualization ensures ongoing host services when data on LUNs of heterogeneous storage systems is migrated.

- **Heterogeneous disaster recovery**

  If service data is scattered at different sites and there are demanding requirements for service continuity, the service sites need to serve as backup sites mutually, and service switchovers can be performed between sites. When a disaster occurs, a functional service site takes over services from the failed service site and recovers data. However, as storage systems at the data site come from different vendors, data on the storage systems cannot be backed up mutually. The synchronous and asynchronous replication functions of SmartVirtualization enable data on LUNs of heterogeneous storage systems to be backed up mutually, achieving data disaster recovery among sites.

- **Heterogeneous data protection**

  Data on LUNs of heterogeneous storage systems may be attacked by viruses or damaged. SmartVirtualization leverages the heterogeneous snapshot function to create snapshots for LUNs of heterogeneous storage systems instantly, and rapidly restores data at a specific point in time using the snapshots if data is damaged.

5 SAN and NAS Converged Architecture

Figure 5-1 shows the converged architecture of the V3 series storage systems. On this architecture, file systems and LUNs work above the space subsystem, whereas the storage pool subsystem based on RAID 2.0+ works under the space subsystem. 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 higher storage efficiency than the traditional unified storage architecture, as shown in Figure 5-2. In addition, LUNs and file systems are independent from each other.

Figure 5-1 OceanStor OS architecture

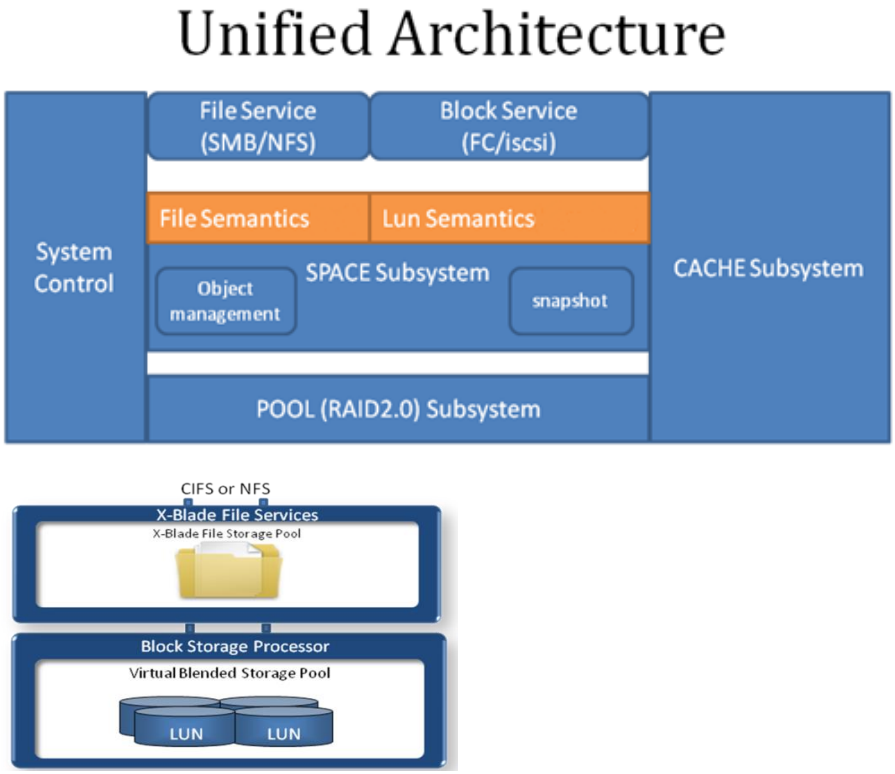
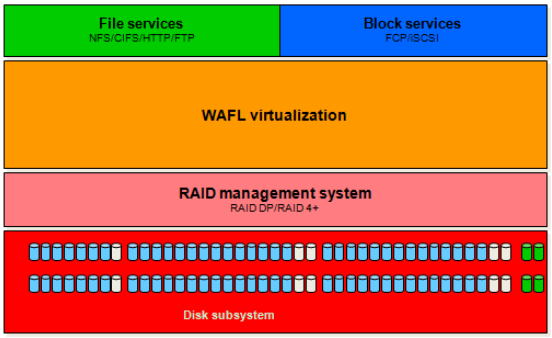


Figure 5-2 Traditional unified storage architecture



For the NAS function of EMC VNX, X-Blade (a NAS gateway) is required to provide file sharing services. File systems and block services work on different operation platforms, complicating the architecture and software stack. For NetApp FAS series, although the unified storage system works on a unified architecture, the block semantics is based on Write Anywhere File Layout (WAFL), and its software layer is more complex than the V3 series storage systems. The V3 series storage systems built on a converged architecture are more efficient than other storage systems in terms of software stack.

6 SmartMigration

The OceanStor 2600 V3 storage systems employ LUN migration to provide intelligent data migration. Services on a source LUN can be completely migrated to the target LUN without interrupting ongoing services. In addition to service migration within a storage system, LUN migration also supports service migration between a Huawei storage system and a compatible third-party storage system. The LUN migration feature provided by the OceanStor 2600 V3 storage systems is called SmartMigration.

SmartMigration replicates all data from a source LUN to a target LUN and uses the target LUN to completely replace the source LUN after the replication is complete. Specifically, all internal operations and requests from external interfaces are transferred from the source LUN to the target LUN transparently.

Implementation of SmartMigration has two stages:

1. Service data synchronization

   Ensures that data is consistent between the source LUN and target LUN after service migration.

2. LUN information exchange

   Enables the target LUN to inherit the WWN of the source LUN without affecting host services.

SmartMigration applies to:

- Storage system upgrade by working with SmartVirtualization

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

- Service performance adjustment

  SmartMigration can be used to improve or reduce service performance. It can migrate services either between two LUNs that have different performances within a storage system or between two storage systems that have different configurations.

  1. Service migration within a storage system

     When the performance of a LUN that is carrying services is unsatisfactory, you can migrate the services to another LUN that provides higher performance to boost service performance. For example, if a user requires quick read/write capabilities, the user can migrate services from a LUN created on low-speed storage media to a LUN created on high-speed storage media. Conversely, if the priority of a type of services decreases, you can migrate the services to a low-performance LUN to release the

high-performance LUN resources for other high-priority services to improve storage system serviceability.

2. Service migration between storage systems

   When the performance of an existing storage system fails to meet service requirements, you can migrate services to a storage system that provides higher performance. Conversely, if services on an existing storage system do not need high storage performance, you can migrate those services to a low-performance storage system. For example, cold data can be migrated to external storage systems without interrupting host services to reduce operating expense (OPEX).

- Service reliability adjustment

  SmartMigration can be used to adjust service reliability of a storage system.

  1. To enhance the reliability of services on a LUN with a low-reliability RAID level, you can migrate the services to a LUN with a high-reliability RAID level. If services do not need high reliability, you can migrate them to a low-reliability LUN.
  2. Storage media offer different reliabilities even when configured with the same RAID level. For example, when the same RAID level is configured, SAS disks provide higher reliability than NL-SAS disks and are more often used to carry important services.

     - LUN type adjustment to meet changing service requirements
     - Conversion between thin LUNs and thick LUNs can be implemented flexibly without interrupting host services.

7 SmartDedupe and SmartCompression

The OceanStor 2600 V3 storage systems not only deliver SAN and NAS converged architecture but also provide data deduplication and compression functions to shrink data for file systems and LUNs. As one of the data storage efficiency improvement methods, the data deduplication and compression functions have extended from the backup area to the primary storage area. They play a critical role in tiered storage with the SSD tier and all-flash arrays because they can save storage space and reduce the TCO of enterprise IT architectures.

The OceanStor 2600 V3 storage systems implements data deduplication based on file systems and thin LUNs in in-line mode. In the storage systems, 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, the storage systems can implement data deduplication based on different granularities. When the data deduplication function is enabled, user data is delivered to the deduplication module in grains. The deduplication module first calculates data fingerprints and then checks whether duplicate fingerprints exist. If yes, the data block is a duplicate one and will not be saved. If no, the data block is a new one and will be delivered to disks for storage. In addition, byte-by-byte comparison can be enabled or disabled. If byte-by-byte comparison is enabled, the deduplication module compares the data with the fingerprint byte by byte.

The OceanStor 2600 V3 storage systems implement data compression based on file systems and thin LUNs in in-line mode. When data compression is enabled, user data is delivered to the compression module in grains and is stored after being compressed. The compression module combines multiple data blocks that belong to the same compression object type and have continuous logical block addresses (LBAs) and compresses these data blocks at a time to improve the compression ratio. Tests show that compression performance is the beast when the compression granularity is 32 KB. For this reason, data blocks whose size is smaller than 32 KB are compressed together, whereas data blocks whose size is larger than 32 KB are compressed directly. To reduce the impact of decompression on host read performance in low-compression ratio scenarios, the compression module checks whether the compression effect reaches the preset threshold. If the compression effect does not reach the threshold, the data is considered low-compression ratio data and will be stored as decompressed data. In this way, the data can be read without decompression, reducing the impact on read performance.

After SmartDedupe and SmartCompression are enabled simultaneously, data is deduplicated and then compressed before being stored onto disks. The following describes the process for processing a data write request when SmartDedupe and SmartCompression are enabled simultaneously:

- Calculates the data fingerprint using the SHA1 algorithm.
- Checks whether a duplicate fingerprint exists in the fingerprint library of the file system or thin LUN.

- (Optional) Compares the data with the fingerprint byte by byte if byte-by-byte compression is enabled.
- Returns duplicate information if a duplicate data block exists and indicates the data block is unique if no duplicate data block exists.
- Compresses the unique data block.
- Writes the compressed data block to the disks and updates data block information in the fingerprint.
- Returns data block deduplication and compression information, such as deduplication and compression flags and physical address, to the file system or thin LUN.

The following describes the process for processing a data read request when SmartDedupe and SmartCompression are enabled simultaneously:

- Reads data from the disk based on the deduplication and compression information, such as physical address, delivered by the file system or thin LUN.
- Determines whether the data block is compressed based on the deduplication and compression flags delivered by the file system or thin LUN and directly returns the data block to the upper-layer application if the data block is not compressed.
- Decompresses and returns the data block to the upper-layer application if the data block is compressed.

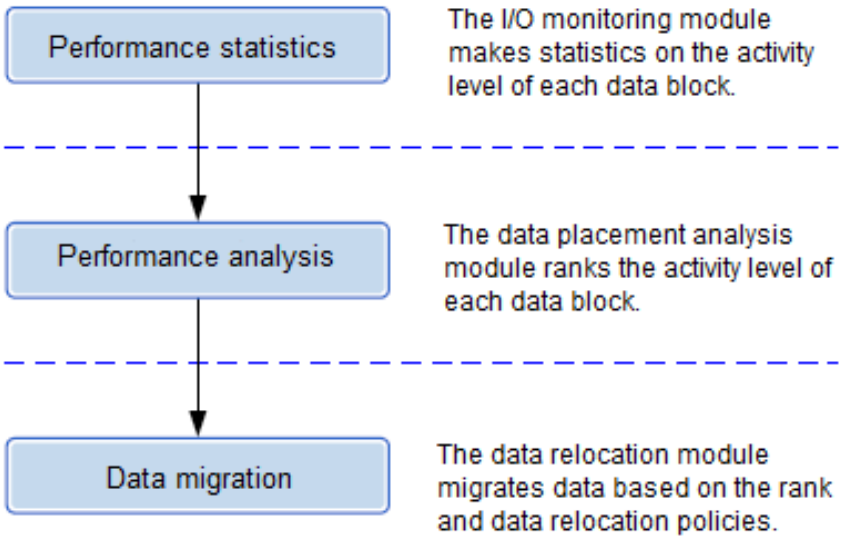
Thanks to the SAN and NAS converged architecture, SmartDedupe and SmartCompression can be enabled or disabled for file systems and thin LUNs, achieving block-based online data shrinking. As SmartDedupe and SmartCompression are independent from each other, they can be enabled or disabled independently. In addition, enabling and disabling these functions do not compromise system performance. SmartDedupe and SmartCompression provided by the V3 high-end storage systems work in in-line mode. When the functions are enabled, new data is deduplicated and compressed. When the functions are disabled, deduplicated data cannot be restored.

8 SmartTier

The OceanStor 2600 V3 storage systems support Huawei's self-developed SmartTier feature. This feature is used to implement automatic storage tiering. SmartTier stores right data onto right media at right time. SmartTier improves storage system performance and reduces storage costs to meet enterprises' requirements on both performance and capacities. By preventing historical data from occupying expensive storage media, SmartTier ensures effective investment and eliminates energy consumption caused by useless capacities, reducing TCO and optimizing cost-effectiveness.

SmartTier performs intelligent data storage based on LUNs and identifies LUNs based on a data migration granularity from 512 KB to 64 MB. The data migration granularity is called extent. SmartTier collects statistics on and analyzes the activity levels of data based on extents and matches data of various activity levels with 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). The data relocation process of SmartTier consists of performance statistics, data analysis, and data relocation, as shown in the following figure:

Figure 8-1 Three phases in data processing by SmartTier



Performance statistics collection and performance analysis are automated by the storage system based on users' configuration, and data migration is initiated manually or by a user-defined scheduled policy.

The I/O monitoring module collects performance statistics.

SmartTier allows user-defined I/O monitoring periods. During the scheduled periods, it collects statistics on data reads and writes. Activity levels of data change throughout the data life cycle. By comparing the activity level of one data block with that of another, the storage system determines which data block is more or less frequently accessed. The activity level of each extent is obtained based on the performance indicator statistics of data blocks.

The working principles are as follows:

1. During scheduled I/O monitoring periods, each I/O is recorded to serve as data sources for performance analysis and forecasting. The following information is recorded based on extents: data access frequency, I/O size, and I/O sequence.
2. The I/O monitoring module records the I/Os of each extent based on memories. Each controller can monitor a maximum of 512 TB storage space.
3. The I/O monitoring module performs weighting for I/O statistics on a daily basis to weaken the impact of historical services on current services.

The data placement analysis module implements performance analysis.

The collected performance statistics are analyzed. This analysis produces rankings of extents within the storage pool. The ranking progresses from the most frequently accessed extents to the least frequently accessed extents in the same storage pool. Note that only extents in the same storage pool are ranked. Then a data migration solution is created. Before data migration, SmartTier determines the extent migration direction according to the latest data migration solution.

The working principles are as follows:

1. The data placement analysis module determines the I/O thresholds of extents on each tier based on the performance statistics of each extent, the capacity of each tier, and the access frequency of each data block. The hottest data blocks are stored to the tier of the highest performance.
2. Extents that exceed the thresholds are ranked. The hottest extents are migrated first.
3. During data placement, a policy is made specifically for SSDs and another policy is made to proactively migrate sequence-degraded extents from SSDs to HDDs.

The data relocation module migrates data.

Frequently accessed data (hotspot data) and seldom accessed data (cold data) are redistributed after data migration. Random hotspot data is migrated to the high-performance tier and performance tier, and non-hotspot data and high-sequence data are migrated to the capacity tier, meeting service performance requirements. In addition, the TCO of the storage system is minimized and the costs of users are reduced.

SmartTier has two migration triggering modes: manual and automatic. The manual triggering mode has a higher priority over the periodic one. In manual triggering mode, data migration can be triggered immediately when necessary. In automatic triggering mode, data migration is automatically triggered based on a preset migration start time and duration. The start time and duration of data migration are user-definable.

In addition, SmartTier supports three levels of data migration speeds: high, medium, and low. The upper limits of the low-level, medium-level, and high-level data migration rates are 10 MB/s, 20 MB/s, and 100 MB/s respectively.

The working principle is as follows:

1. The data relocation module migrates data based on migration policies. In the user-defined migration period, data is automatically migrated.
2. The data relocation module migrates data among various storage tiers based on migration granularities and the data migration solution generated by the data placement analysis module. In this way, data is migrated based on activity levels and access sequences.
3. The data relocation module dynamically controls data migration based on the current load of a storage pool and the preset data migration speed.
4. The minimum unit for data migration is extent. Service data can be correctly accessed during migration. Relocating an extent is to read data from the source extent and write the data to the target extent. During data migration, read I/Os read data from the source extent while write I/Os write data to both the source and target extents. After data migration, the metadata of the source and target extents is modified. Then read and write I/Os access the target extent. The source extent is released.

9 SmartThin

The OceanStor 2600 V3 storage systems support SmartThin, a Huawei's proprietary thin provisioning feature. SmartThin allows users to allocate a certain amount of capacity to a LUN during the creation of that LUN. When the LUN is being used, storage space is further allocated on demand to improve the efficiency in using storage resources and meet service requirements. Instead of allocating all space in advance, SmartThin presents users a virtual storage space that is larger than the physical storage space, so that the storage capacity visible to uses is much larger than the capacity actually allocated by the storage system. When users begin to use storage space, SmartThin provides the required space for users based on the capacity on demand mechanism. If the storage space is insufficient, SmartThin triggers storage unit expansion to increase system storage space. The whole expansion process is transparent to users and causes no system downtime.

If the actual amount of data is larger than expected, the LUN space can be adjusted dynamically. Free space can be allocated to any LUN that needs space. In this way, storage space utilization and effectiveness are improved. In addition, the LUN size can be adjusted non-disruptively without affecting services.

SmartThin creates thin LUNs based on a RAID 2.0+ virtual storage resource pool. Thin LUNs and traditional thick LUNs coexist in the same storage resource pool. Thin LUNs are logical units created in a thin pool. They can be mapped to and then accessed by hosts. The capacity of a thin LUN is not its actual physical space but only a virtual value. A thin LUN applies for physical space from a storage resource pool based on the capacity-on-write policy only when the thin LUN starts processing I/O requests.

SmartThin allows the capacity detected by a host to be larger than the actual capacity of a thin LUN. The capacity detected by a host is the thin LUN size that a user can create, namely, the volume capacity (virtual space) displayed on the host after a thin LUN is mapped to a host. The actual capacity of a thin LUN refers to the amount of physical space actually occupied by a thin LUN. SmartThin makes the actual capacity of a thin LUN invisible to a host and provides the nominal capacity of a thin LUN for a host.

In addition, SmartThin allows users to create a thin LUN whose capacity is larger than the maximum available physical capacity of a storage resource pool. For example, if the maximum physical capacity provided by a storage resource pool is 2 TB, SmartThin allows a thin LUN larger than 10 TB to be created.

SmartThin uses the capacity-on-write and direct-on-time technologies to respond to thin LUN read and write requests initiated by hosts. Capacity-on-write is used to allocate space upon write requests, and direct-on-time is used to redirect data read and write operations.

- Capacity-on-write

Upon receiving a write request from a host, a thin LUN uses direct-on-time to determine whether the logical storage location of the request is allocated an actual storage location. If an actual storage location is not allocated, a space allocation task is triggered with a 64 KB grain as the minimum allocation granularity. Then data is written to the newly allocated actual storage location.

- Direct-on-time

  Because capacity-on-write is used, the relationship between the actual storage location and logical storage location of data is not calculated using a fixed formula but determined by random mappings based on capacity-on-write. When data is read from or written to a thin LUN, the relationship between the actual storage location and logical storage location must be redirected based on a mapping table. A mapping table is used to record the mapping relationship between an actual storage location and a logical storage location. A mapping table is dynamically updated during the write process and is queried during the read process. Therefore, direct-on-time is classified into read direct-on-time and write direct-on-time.

  - Read direct-on-time

    Upon receiving a read request from a host, a thin LUN queries the mapping table. If the logical storage location of the read request is allocated an actual storage location, the thin LUN redirects the logical storage location to the actual storage location, reads data from the actual storage location, and returns the data to the host. If the logical storage location of the read request is not allocated an actual storage location, the thin LUN sets data in the logical storage location to all zeros and returns all zeros to the host.

  - Write direct-on-time

    Upon receiving a write request from a host, a thin LUN queries the mapping table. If the logical storage location of the write request is allocated an actual storage location, the thin LUN redirects the logical storage location to the actual storage location, writes data to the actual storage location, and returns a response to the host indicating a successful data write. If the logical storage location of the write request is not allocated an actual storage location, the thin LUN performs operations based on capacity-on-write.

SmartThin supports online capacity expansion of a thin LUN and that of a storage resource pool. The two expansion approaches do not affect services running on hosts.

Capacity expansion of a thin LUN is to expand the nominal storage capacity of the thin LUN. After the nominal storage capacity of a thin LUN is changed, SmartThin automatically provides the new nominal storage capacity of the thin LUN for the host. In this way, the volume capacity (virtual space) displayed on the host is the expanded capacity. The expansion does not adjust the original storage locations. If data needs to be written to the newly added storage space, the thin LUN applies for physical storage space from the storage resource pool based on capacity-on-write.

Expansion of a storage resource pool is a capability provided by RAID 2.0+ storage virtualization to expand storage capacity without affecting services running on hosts. In addition, SmartMotion enables data to be evenly distributed on all disks (including newly added disks) in the storage resource pool.

SmartThin provides two space reclamation approaches: standard SCSI command (**unmap**) reclamation and all-zero data space reclamation. The implementation principles of the two approaches are described as follows:

Standard SCSI command reclamation: In scenarios such as virtual machine deletion, a host issues the **unmap** command using the SCSI protocol. Upon receiving this command, SmartThin uses direct-on-time to search for the actual storage location that corresponds to the

logical storage location to be released on a Thin LUN, releases the actual storage location from the Thin LUN to the storage resource pool, and deletes the corresponding mapping from the mapping table. This space reclamation approach requires applications on hosts to be able to issue the **unmap** command. VMware, SF, and Windows Server 2012 support the **unmap** command.

All-zero data space reclamation: Upon receiving a write request from a host, SmartThin determines whether the data block contained in the write request is all zeros. If the logical storage location to which the all-zero data block is delivered is not allocated an actual storage location, SmartThin returns a response indicating a successful data write to the host without allocating space. If the logical storage location to which the all-zero data block is delivered is allocated an actual storage location, SmartThin releases the actual storage location from the thin LUN to the storage resource pool, deletes the corresponding mapping from the mapping table, and returns a response indicating a successful data write to the host. This space reclamation approach does not require hosts to send special commands.

10 SmartQoS

The OceanStor 2600 V3 storage systems support SmartQoS, a Huawei's proprietary QoS guarantee feature. SmartQoS intelligently schedules and allocates computing resources, cache resources, concurrency resources, and disk resources of a storage system, meeting the QoS requirements of services that have different priorities.

SmartQoS ensures the QoS of data services based on the following techniques:

- **I/O priority scheduling**

  Service response priorities are determined based on the priorities of different services. When allocating system resources, a storage system gives priority to the resource allocation requests initiated by services that have a high priority. If resources are insufficient, more resources are allocated to services that have a high priority to maximize their QoS. Currently, three priorities are available: high, medium, and low.

- **I/O traffic control**

  Based on the user-defined performance control goal (IOPS or bandwidth), the traditional token bucket mechanism is used to control traffic. I/O traffic control prevents specific services from generating excessively large traffic, which affects other services.

- **I/O performance protection**

  Based on traffic suppression, a user is allowed to specify the lowest performance goal (minimum IOPS/bandwidth or maximum latency) for a service that has a high priority. If the minimum performance of the service cannot be ensured, the storage system gradually increases the I/O latency of low-priority services, thereby restricting the traffic of low-priority services and ensuring the lowest performance goal of the high-priority service.

I/O priority scheduling is implemented based on storage resource scheduling and allocation. The performance of a storage system is determined by the consumption of storage resources in a specific application scenario. Therefore, the system performance is optimized as long as resources, especially bottleneck resources, are properly scheduled and allocated. I/O priority scheduling monitors the usage of computing resources, cache resources, concurrency resources, and disk resources that have the greatest performance impact. If a resource bottleneck occurs, resources are scheduled to meet the needs of high-priority services to the maximum. In this way, the QoS of mission-critical services is ensured in different scenarios.

The I/O priority scheduling technique employed by SmartQoS schedules critical bottleneck resources along the I/O path. Those resources include **computing resources**, **cache resources**, **concurrency resources**, and **disk resources**. Scheduling policies are implemented based on priorities specified by users for LUNs or file systems. Different priorities correspond to different scheduling policies. The priority of a LUN or file system is specified by a used based

on the importance of the application served by the LUN or file system. Three priorities are available: **high**, **medium**, and **low**.

The I/O priority scheduling technique controls the allocation of front-end concurrency resources, computing resources, cache resources, and disk resources to control the response time of each schedule object.

- Priority scheduling of **front-end concurrency resources** is implemented at the front end of the storage system to control concurrent access requests initiated by hosts. A storage system's capability to process concurrent access requests initiated by hosts is limited. Therefore, when the maximum number of concurrent access requests supported by a storage system is reached, the SmartQoS restricts the maximum number of concurrent access requests for each priority based on the number of LUNs or file systems of each priority running under each controller. The restriction rule is as follows: High-priority services and large-traffic services are allocated a larger number of access concurrency resources.
- Priority scheduling of **computing resources** is implemented by controlling the allocation of CPU runtime resources. Based on the weight of each of the high, medium, and low priorities, SmartQoS allocates CPU runtime resources to services of each priority. When CPU resources become a performance bottleneck, priority scheduling ensures that high-priority services are allocated more CPU runtime resources.
- Priority scheduling of **cache resources** is implemented by controlling the allocation of cache page resources. Based on the weight of each priority, SmartQoS preferentially processes page allocation requests initiated by high-priority services.
- Priority scheduling of **disk resources** is implemented by controlling the I/O delivery sequence. Based on the priorities of I/Os, SmartQoS enables high-priority I/Os to access disks first. If most I/Os are queuing at the disk side, priority scheduling of disk resources reduces the queuing time of high-priority I/Os. In this way, the overall latency of high-priority I/Os is reduced.

The priority scheduling technique employed by SmartQoS is implemented based on priorities of LUNs or file systems. Each LUN or file system has a priority attribute specified by a user and saved in the database. When an I/O request initiated by a host (SCSI target) reaches a storage array, the priority of the LUN or file system to which the I/O request is destined is added to the I/O request. Then the I/O request carries the priority throughout the I/O path. In this way, priority scheduling is implemented for services.

Currently, the I/O traffic control technique mainly applies to LUNs. It restricts the overall IOPS or bandwidth of one or multiple LUNs in the storage system, so that the performance of some applications is restricted, thereby preventing these applications from causing a traffic burst that affects the performance of other applications.

The I/O traffic control technique restricts data processing resources available to data services on specific LUNs. There are two types of traffic control objects: I/O type (read, write, or both read and write) and traffic type (IOPS or bandwidth). A traffic control 2-tuple (I/O type and traffic type) aimed at a specific LUN is obtained.

The I/O traffic control technique employed by SmartQoS controls traffic based on 2-tuples and QoS policies. Each QoS policy corresponds to a traffic control group, namely, a LUN group for which an upper traffic limit is set. Implementation of QoS policy–based traffic control includes QoS policy–based queue management, token allocation, and de-queuing control.

The I/O latency control technique protects high-priority services and restricts low-priority services to ensure that the minimum performance requirements of mission-critical services are met. A user can set a lowest performance goal for a high-priority service. If the lowest performance goal cannot be reached, the storage system lowers the performance of

low-priority services and then that of medium-priority services to achieve the lowest performance goal set for the high-priority service.

SmartQoS gradually increases the latency of low-priority and medium-priority services to restrict their performance. To prevent performance instability, SmartQoS stops increasing the latency as soon as the maximum latency is reached. In addition, when the performance of a service whose lowest performance goal must be achieved reaches 1.2 times of the minimum performance, SmartQoS gradually reduces the latency of the low-priority and medium-priority services.

11 SmartPartition

The OceanStor 2600 V3 storage systems support SmartPartition, a Huawei's proprietary cache partitioning feature. The core value of SmartPartition is to ensure the performance of critical applications by partitioning core system resources. Administrators 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. Based on actual service conditions, the storage system dynamically adjusts the front- and back-end concurrent accesses to different cache partitions, ensuring the performance of the application using a specific partition. SmartPartition can be used with other QoS features (such as SmartQoS) to achieve better QoS effect.

Caches are classified into read cache and write cache. The read cache prefetches and retains data to improve the hit ratio of read I/Os. The write cache improves the disk access performance by means of combination, hitting, and sequencing. Different services need read and write caches of different sizes. SmartPartition allows users to specify read and write cache sizes for a partition, meeting cache requirements of different services.

The read cache configuration and the write cache configuration affect the I/O procedure differently. The impact on the write I/Os shows up in the phase of cache resource allocation. In this phase, the host concurrency and write cache size of a partition are determined. The reason for determining the two items in this phase is that it is the initial phase of a write process and actually occupies no cache resources. The impact on read I/Os involves two aspects. The first aspect is similar to the write I/O scenario. Specifically, the storage system determines whether the host concurrency meets the requirement. If the requirement is not met, the storage system returns the I/Os. The read cache is intended to control the size of the cache occupied by read data. The size of a read cache is controlled by the read cache knockout process. Therefore, the second aspect of the impact shows up in the read cache knockout process. If the read cache size of the partition does not reach the threshold, read cache resources are knocked out extremely slowly. Otherwise, read cache resources are knocked out quickly to ensure that the read caches size is below the threshold.

Compared with host applications, the processing resources of a storage system are limited. Therefore, a storage system must restrict the total host concurrency amount. For each partition, the concurrency is also restricted to ensure the QoS.

Regarding SmartPartition, the host concurrency of a partition is not fixed but calculated based on the priority weighted algorithm with the following factors taken into account:

- Number of active LUNs or file systems in the partition in the last statistics period
- Priorities of active LUNs or file systems in the partition in the last statistics period
- Number of I/Os completed by each LUN or file system in the last statistics period

- Number of I/Os returned to hosts because the partition concurrency of each LUN or file system reaches the threshold in the last statistics period

Weighting the preceding factors not only fully uses the host concurrency capability but also ensures the QoS of a partition.

After one statistics period elapses, the concurrency capability of a partition may need to be adjusted based on the latest statistical result. The SmartPartition logic adjusts the concurrency capability based on a specific step to ensure a smooth adjustment, minimizing host performance fluctuation.

Similar to host concurrency control, back-end concurrency control is also intended to fully use system resources while ensuring the QoS of a partition. The back-end concurrency is also calculated based on the priority weighted algorithm with the following factors taken into account:

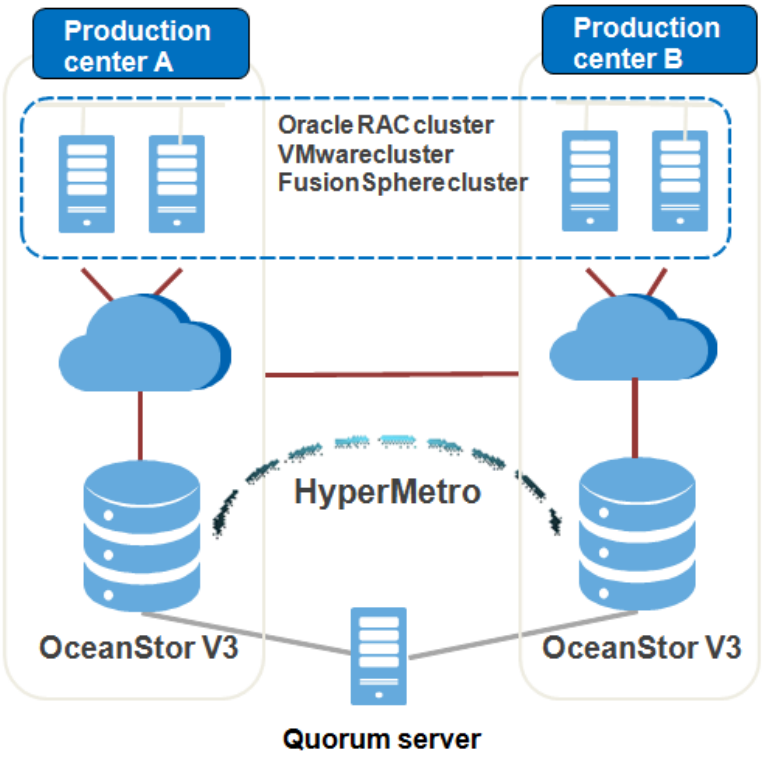
- Amount of dirty data on each priority of LUNs or file system in a partition in the last statistics period
- Disk flushing latency of LUNs or file systems in a partition in the last statistics period
- Actual disk flushing latency of LUNs or file systems in a partition in the last statistics period

The adjustment period and approach are similar to those of host concurrency control.

12 HyperMetro

The OceanStor 2600 V3 storage systems support HyperMetro, a Huawei's proprietary array-level active-active feature. It enables two storage systems to work in active-active mode in the same equipment room, in the same city, or in two places that are 100 km away or more from each other. HyperMetro allows two LUNs from two storage arrays to maintain real-time data consistency and to be concurrently accessible to hosts. If one storage array fails, hosts will automatically choose the path to the other storage array for continued service access. If only one storage array can be accessed by hosts due to failures of the links between storage arrays, the arbitration mechanism determines which storage array continues to provide services. The quorum server provides the arbitration result and it is deployed at a third-place site.

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



HyperMetro has the following features:

- Gateway-free active-active solution: The networking is simple and the deployment is easy. Without the use of a gateway, the reliability and performance are higher because there is one less possible failure point and 0.5 ms to 1 ms of latency caused by a gateway is avoided.
- Active-active mode: Storage arrays in both data centers support data read and write. The upper-layer application system can make full use of such a service capability to implement load balancing across data centers.
- Site access optimization: Huawei proprietary multipathing software UltraPath is optimized specific to active-active scenarios. UltraPath can identify region information to reduce cross-site access, thereby reducing latency. It can read data from the local or remote storage array. However, when the local storage array is working properly, UltraPath preferentially reads and writes data on the local storage array, preventing data read and write across data centers.
- FastWrite: In a normal SCSI write process, a write request involves two interactions: Write Alloc and Write Data. A write request requires two round trips between sites. FastWrite optimizes the storage transfer protocol by pre-allocating cache space on the target side to receive the write request. It eliminates the Write Alloc operations and requires only one interaction. FastWrite halves the latency of data synchronization between storage arrays, improving the performance of an active-active solution.
- Service-specific arbitration: HyperMetro can implement arbitration on a per service basis. Specifically, when links between two data centers are broken, HyperMetro determines which services run in data center A and which run in data center B. Compared with traditional arbitration that allows only one data center to survive, the arbitration mechanism with HyperMetro reduces the required amount of reserved host and storage resources, as well as balancing service load more efficiently. The arbitrated services can be on a LUN pair or consistency group.
- Link quality adaption: If there are multiple links between two data centers, HyperMetro balances service load on those links based on the quality of each link. HyperMetro monitors the link quality in real time and dynamically adjusts the load ratio between two those links, thereby reducing the re-transfer rate and enhancing network performance.
- Wide compatibility with other features: HyperMetro can work with SmartThin, SmartTier, SmartQoS, SmartCache, and also SmartVirtualization (HyperMetro allows a local LUN to be paired with a third-party LUN connected using SmartVirtualization). In addition, HyperMetro can combine with HyperSnap, HyperClone, HyperMirror, and HyperReplication to deliver a more complex, advanced data protection solution, such as the Disaster Recovery Data Center Solution (Geo-Redundant) with local active-active and remote replication.

13 HyperVault

The OceanStor 2600 V3 storage systems support HyperVault, a Huawei's proprietary integrated backup technology. It implements file system data backup and recovery within a storage system or between storage systems.

HyperVault has two working modes:

- Local backup

  It indicates data backup within a storage system. HyperVault employs a file system–based snapshot mechanism to periodically back up the file system and generate backup copies. By default, one file system has five backup copies.

- Remote backup

  It indicates data backup between storage systems. In this scenario, HyperVault employs a file system–based remote replication technology to periodically back up the file system. After the source storage side generates a backup snapshot, the incremental data between this snapshot and its previous snapshot is worked out, and then the incremental data is copied to the backup storage side. After data backup is complete, a snapshot is created on the backup storage side. By default, 35 snapshots are saved on the backup storage side.

HyperVault has the following advantages:

1. Low costs

   HyperVault can be seamlessly integrated into the primary storage system. The embedded storage management software OceanStor DeviceManager on the primary storage system allows users to configure flexible backup policies and perform data backup functions.

2. Quick data backup

   The HyperVault local backup employs the snapshot technology to complete a data backup job within seconds. The remote backup performs full backup at the first time and then only backs up incremental data blocks. Compared with the backup software that backs up files each time, HyperVault provides quicker data backup.

3. Higher recovery efficiency

   The HyperVault local recovery employs snapshot rollback technology and does not require additional data resolution, achieving data recovery within seconds. Remote recovery is a complement to local recovery. It uses full data recovery for enhanced data reliability. Each piece 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.

4. Simple management

   A HyperVault backup network comprises only one primary storage system and one backup storage system, and uses native management software OceanStor

DeviceManager, to manage the systems. Compared with other backup networks that contain primary storage, backup software, and backup media, a HyperMetro backup network is easier to manage.

14 Summary

The OceanStor 2600 V3 storage systems are brand-new converged storage systems that employ a cloud-oriented architecture. They feature block-level virtualization, SAN and NAS convergence, online deduplication/compression, and heterogeneous virtualization. Thanks to their converged architecture, OceanStor 2600 V3 storage systems have the rich features that are with high-end storage systems, including Scale-Out, online deduplication/compression, and heterogeneous virtualization. They have SmartPartition and SmartCache features to boost system performance for file system sharing and block-based access, and SmartDedupe and SmartThin to improve the storage efficiency. OceanStor 2600 V3 storage systems apply to a wide range of scenarios, including virtualization, heterogeneous storage, file servers, and databases. Huawei is dedicated to providing high-quality storage products and user-friendly services for customers. Guided by this concept, OceanStor 2600 V3 storage systems fully meet customers' requirements for high performance, functions, and efficiency, and maximize customer benefits.

15 Acronyms and Abbreviations

Table 15-1 Acronyms and abbreviations

Acronym or Abbreviation	Full Spelling
BP	Block Point
DIF	Data Integrity Field
eDevLun	External Device LUN
HDD	Hard Disk
IOPS	Input and Output Per Second
KV_DB	KeyValue-DataBase
LUN	Logical Unit Number
NL-SAS	Nearline Serial Attached SCSI
OLAP	Online Analysis Process
OLTP	Online Transaction Process
QoS	Quality of Service
RAID	Redundant Array of Independent Disks
SAS	Serial Attached SCSI
SCSI	Small Computer System Interface
SSD	Solid State Disk
TCO	Total Cost of Ownership
WAFL	Write Anywhere File Layout
WWN	World Wide Number