Huawei FusionCube DB 3.1

Technical White Paper

Issue 04

Date 2018-12-26

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

About This Document

Overview

This document introduces the FusionCube 3.1 Database Infrastructure from the following perspectives: benefits, architecture, performance, scalability, security, and reliability.

You can obtain comprehensive information about FusionCube by reading this document.

Intended Audience

This document is intended for:

- Marketing engineers
- Technical support engineers
- Maintenance engineers

Symbol Conventions

The symbols that may be found in this document are defined as follows.

Symbol	Description
DANGER	Indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.
WARNING	Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.
CAUTION	Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury.
NOTICE	Indicates a potentially hazardous situation which, if not avoided, could result in equipment damage, data loss, performance deterioration, or unanticipated results. NOTICE is used to address practices not related to personal injury.

Symbol	Description
NOTE	Calls attention to important information, best practices and tips. NOTE is used to address information not related to personal injury, equipment damage, and environment deterioration.

Change History

Issue	Date	Description
04	2018-12-26	This issue is the fourth official release.
03	2018-10-26	This issue is the third official release.
02	2018-08-15	This issue is the second official release.
01	2018-06-01	This issue is the first official release.

Contents

1 Product Description

With the rise of data and Internet services, new services are growing rapidly, resulting in an exponential increase of service data. The traditional server+storage architecture can no longer meet service development requirements. In this background, distributed and cloud-based technologies emerge. An increasing number of enterprises now tend to use virtualization and cloud computing technologies to build their IT systems, in an attempt to improve the IT system resource usage and shorten the time to market (TTM). However, they are facing the following challenges:

- Complex management and soaring operation and maintenance (O&M) costs
- High planning, deployment, and optimization skill requirements on operation personnel because hardware devices may be provided by different vendors
- Slow response to handling after-sales problems on operation portals of multiple vendors
- Huge maintenance system that involves hardware maintenance and virtualization platform management of multi-vendor products

In addition, customers now attach more importance to cost control, rapid service provisioning, and risk management. They want resource-scalable, reliable, and high-performance IT systems with low total cost of ownership (TCO) and short TTM.

To help customers address these concerns, Huawei rolls out the FusionCube Hyper-converged Infrastructure, which is an open, scalable system that boasts the following outstanding features: compute, storage, and network convergence, preintegration, high performance, high reliability, high security, automatic and quick service deployment, unified management, intelligent resource scaling, and easy O&M. This solution allows customers to quickly deploy services and cloud applications while significantly reducing the difficulty in maintenance and management.

2 Product Features

FusionCube is a flagship Huawei IT product. Designed based on an open architecture, FusionCube is able to integrate servers, distributed storage, and network switches, eliminating the need for external storage devices. In addition, FusionCube is preintegrated with the distributed storage engine and management software and therefore supports on-demand resource allocation and linear expansion. Key messages of FusionCube are: converged, simple, optimized, and open.

Converged

FusionCube integrates computing, storage, and network resources.

- Hardware convergence: Computing, storage, and network resources are integrated into a linearly scalable system.
- Management convergence: Centralized O&M significantly improves resource utilization and reduces operating expenses (OPEX).
- Application convergence: Hardware and software are optimized based on application service models to improve system performance.

Simple

FusionCube supports system preintegration and preverification, automatic device discovery upon power-on, and unified O&M, greatly simplifying service delivery.

- Simplified installation: FusionCube provides a quick installation tool that helps you complete software installation by one click.
- Rapid deployment: Upon system power-on, FusionCube automatically discovers devices and configures parameters, making service rollout more efficient.
- Easy O&M: FusionCube provides a unified management GUI and supports automatic fault locating, making routine O&M simpler.

Optimized

FusionCube uses industry-leading hardware and distributed storage software to ensure optimal user experience.

- Storage optimized: FusionCube uses built-in distributed storage to provide storage services with high concurrency and throughput.

- Network optimized: FusionCube supports 56 Gbit/s and 100 Bit/s InfiniBand (IB) and remote direct memory access (RDMA) to provide the fastest switching network in the industry.

Open

FusionCube is an open infrastructure platform independent of specific applications. It provides computing, storage, and network resources for mainstream databases.

- The open and highly efficient platform supports mainstream commercial databases, such as Oracle RAC, IBM DB2, and Sybase IQ.
- FusionCube has passed the SAP HANA certification and has been used to deploy the global largest SAP HANA cluster.

3 System Architecture

3.1 System Architecture

Figure 3-1 shows the architecture of the FusionCube database infrastructure.

Figure 3-1 Huawei FusionCube architecture

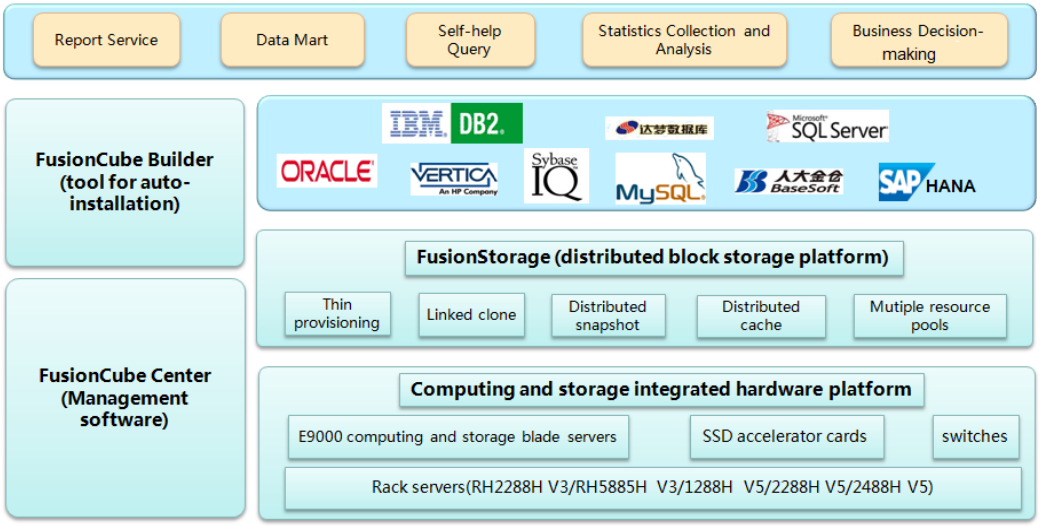


Table 3-1 FusionCube components

Component	Function
FusionCube Center	Manages FusionCube storage and hardware resources, and implements system monitoring and O&M.
FusionCube Builder	Provides quick installation and deployment of FusionCube system software. It can be used to replace or update the database platform software.
FusionStorage	Provides high-performance and highly reliable block storage services from the server storage (SAS/SATA/NL-SAS HDDs, SSDs, or NVMe SSDs) that is organized and scheduled using the distributed storage technology.
Hardware platform	Consists of Huawei E9000 and rack servers. The servers provide the following features: ● Modular design of compute, storage, switch, and power modules. ● On-demand configuration of compute nodes and storage nodes in a chassis. ● Support for GPU and SSD PCIe expansion and acceleration. ● Rich options of 10GE and IB switch modules.

As a flagship product of Huawei, FusionCube complies with open architecture and standards. It integrates servers, distributed storage, and network switches in an out-of-the-box packaging. External storage devices or switches are not required. FusionCube is pre-integrated with distributed storage engines and management software to implement on-demand resource allocation and linear expansion.

3.1.1 Open Database Infrastructure Platform

FusionCube is an open hyper-converged infrastructure platform independent of specific database applications. It provides compute, storage, and network resources for mainstream database applications. FusionCube has the following features:

- Supports mainstream databases, such as Oracle RAC, Sybase IQ, IBM DB2, and SAP HANA.
- Provides built-in network and storage devices, eliminating the need to purchase extra switches or storage devices.
- Supports unified management, improving resource utilization and reducing O&M costs.
- Uses built-in distributed storage to deliver higher IOPS, faster data exchange and application query, and lower latency than conventional databases.

3.1.2 Convergence of Computing, Storage, and Networking Resources

FusionCube is prefabricated with compute, network, and storage devices in an out-of-the-box packaging, eliminating the need for users to purchase extra storage or network devices.

- Distributed storage engines are deployed on compute nodes (server blades) to implement convergence of computing and storage resources, which reduces data access delay and improves overall access efficiency.
- FusionCube implements automatic network deployment and network resource configuration. In addition, the automatic network resource configuration is dynamically associated with computing and storage resource allocation.

3.1.3 Distributed Block Storage

FusionStorage Block provides distributed storage services for FusionCube. FusionStorage Block uses an innovative cache algorithm and adaptive data distribution algorithm based on a unique parallel architecture, which eliminates high data concentration and improves system performance. FusionStorage Block also allows rapid automatic self-recovery and ensures high system availability and reliability.

- Linear scalability and elasticity

  FusionStorage Block uses the distributed hash table (DHT) to distribute all metadata among multiple nodes. This mode prevents performance bottlenecks and allows linear expansion. FusionStorage Block leverages innovative data slicing technology and DHT-based data routing algorithm to evenly distribute volume data to fault domains of large resource pools. This allows load balancing on hardware devices and higher input/output operations per second (IOPS) and megabit per second (MBPS) performance of each volume.
- Supreme performance

  FusionStorage Block uses a lock-free scheduled I/O software subsystem to prevent conflicts of distributed locks. I/O paths are shortened and the delay is reduced as there is no lock operation or metadata query on I/O paths. Distributed stateless engines make hardware nodes to be fully utilized, greatly increasing the concurrent IOPS and MBPS of the system. FusionStorage Block provides ultimate performance by using the distributed SSD cache technology as well as large-capacity SAS/SATA/NL-SAS SSDs as the primary storage.
- High reliability

  FusionStorage Block supports multiple data redundancy protection mechanisms, such as two copies, three copies, and EC. Based on these mechanisms, FusionStorage Block supports flexible data reliability policies, allowing different copies to be stored on different servers (server-level reliability) or in different cabinets (cabinet-level reliability). This ensures that data is not lost and can be accessed even if the server is faulty or the entire cabinet is powered off. FusionStorage Block also protects valid data slices against loss. If a hard disk or server is faulty or the entire cabinet is powered off, valid data can be rebuilt concurrently. It takes less than 30 minutes to rebuild data of 1 TB. All these measures improve system reliability.
- Rich advanced storage functions
  - The thin provisioning function provides users with more virtual storage resources than physical storage resources. Physical storage space is allocated to a volume only when data is written into the volume.
  - The volume snapshot function saves the state of the data on a logical volume at a certain time point. The number of snapshots is not limited, and performance does not deteriorate.
  - The linked clone function is implemented based on incremental snapshots. A snapshot can be used to create multiple cloned volumes. When a cloned volume is created, the data on the volume is the same as the snapshot. Subsequent

modifications on the cloned volume do not affect the original snapshot and other cloned volumes.

3.1.4 Automatic Deployment

FusionCube supports co-deployment of virtualization and physical deployment. The virtualization deployment improves resource utilization and implements cloud-based management. Moreover, deployment of database services on physical servers improves database performance.

- FusionCube supports preintegration, and preverification before the delivery, which simplifies on-site installation and deployment and reduces the deployment time.
- One-click installation helps simplify the installation and deployment, greatly reducing the installation and deployment time.
- Devices are automatically discovered upon the system is powered on. Wizard system initialization configuration is provided to implement initialization of compute, storage, and network resources.

3.1.5 Unified O&M

FusionCube supports unified management of hardware devices (such as servers and switches) and resources (including compute, storage, and network resources). It can greatly improve O&M efficiency and quality of service (QoS).

- A unified management interface is provided for managing hardware devices and monitoring compute, storage, and network resources on a real-time basis.
- The IT resource usage and system operating status are automatically monitored. Alarms are reported for system faults and potential risks in real time, and alarm notifications can be sent to O&M personnel via email.
- FusionCube allows rapid, automatic capacity expansion. It supports automatic discovery of devices to be added and provides wizard-based configuration for system expansion.

3.2 Distributed Block Storage

3.2.1 System Architecture

FusionStorage Block employs the distributed cluster control technology and DHT routing technology to implement distributed storage. **Figure 3-2** shows the functional architecture of FusionStorage Block.

Figure 3-2 Functional architecture of FusionStorage Block

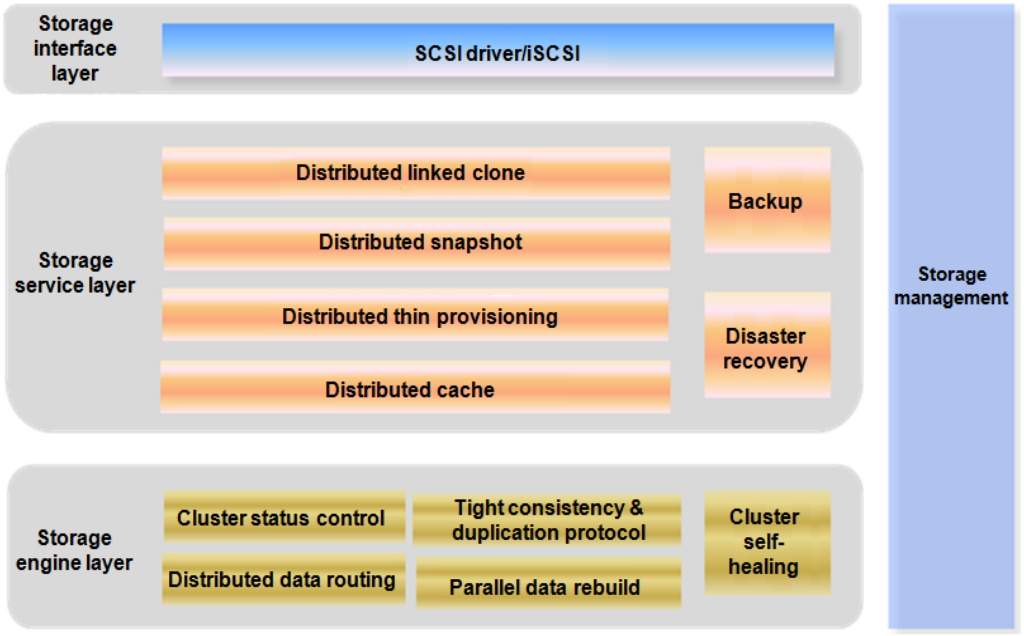


Table 3-2 Layers of FusionStorage Block

Name	Description
Storage interface layer	Provides volumes for OSs and databases over Small Computer System Interface (SCSI) or Internet Small Computer Systems Interface (iSCSI).
Storage service layer	Provides various advanced storage features, such as snapshot, linked clone, thin provisioning, distributed cache, and disaster recovery (DR) and backup.
Storage engine layer	Provides basic storage functions, including management status control, distributed data routing, strong-consistency replication, cluster self-recovery, and parallel data rebuilding.
Storage management layer	Provides O&M functions, such as software installation, automatic configuration, online upgrade, alarming, monitoring, and logging, and also provides a portal for user operations.

Figure 3-3 shows the logical architecture of FusionStorage Block.

Figure 3-3 Logical architecture of FusionStorage Block

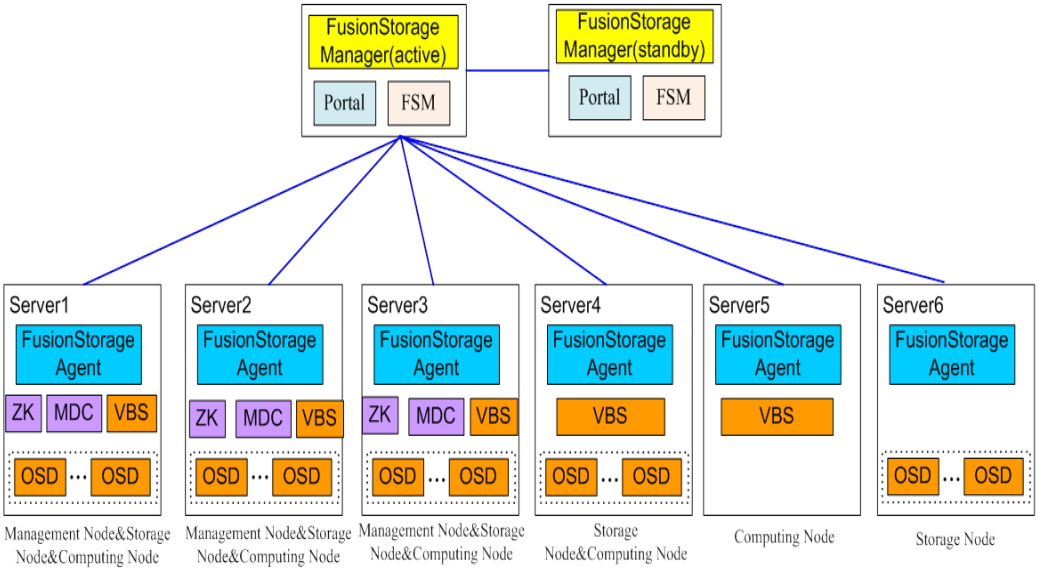


Table 3-3 Logical components of FusionStorage Block

Name	Description
FusionStorage Manager	A FusionStorage management module, providing O&M functions such as alarming, monitoring, logging, and configuration. Two FusionStorage Managers are generally deployed in active/standby mode.
FusionStorage Agent	An agent process deployed on each node, for communicating with FusionStorage Manager, collecting monitoring and alarm information of the node, and receiving upgrade packages and performing upgrades when software components on the node need to be upgraded.
ZK	A ZooKeeper process. A FusionStorage system needs three, five, or seven ZK processes to form a ZK cluster, which elects a primary metadata controller (MDC) for the MDC cluster. At least three ZKs are deployed, and more than 50% of the ZKs are available.

Name	Description
MDC	Metadata control software for controlling the status of the distributed cluster and managing the data distribution and rebuilding rules. At least three MDCs are deployed in the system to form an MDC cluster. When the system starts, the ZK cluster elects a primary MDC out of multiple MDCs. The primary MDC monitors other MDCs. When the primary MDC becomes faulty, another MDC is elected as a new primary MDC. Each resource pool is configured with a home MDC. When the home MDC of a resource pool becomes faulty, the primary MDC instructs another MDC to host the resource pool. One MDC manages a maximum of two resource pools. The MDC can start at any storage node as a process. An MDC is automatically enabled when a resource pool is added. One system runs a maximum of 96 MDCs.
VBS	A virtual block storage management component for managing volume metadata. The VBS provides the distributed storage access point service through the SCSI or iSCSI, enabling computing resources to access distributed storage resources. The VBS communicates with the object storage devices (OSDs) of all accessible resource pools in a point-to-point (PTP) manner, to concurrently access all hard disks of these resource pools. One VBS is deployed on each node by default. The VBSs on multiple nodes form a VBS cluster. When the VBSs start, they connect to the primary MDC and elect a primary VBS by means of coordination. Alternatively, multiple VBSs can be deployed on one node to improve the I/O performance.
OSD	A key-value (KV) device service for performing specific I/O operations. Multiple OSD processes are deployed on each node. One OSD process is deployed for one disk by default. When SSD cards are used as the primary storage, multiple OSD processes can be deployed on one SSD card to maximize the SSD card performance. For example, two OSD processes can be deployed on one 3.2 TB SSD card, and each OSD process manages the I/O operations for 1.6 TB space.

3.2.2 Key Service Processes

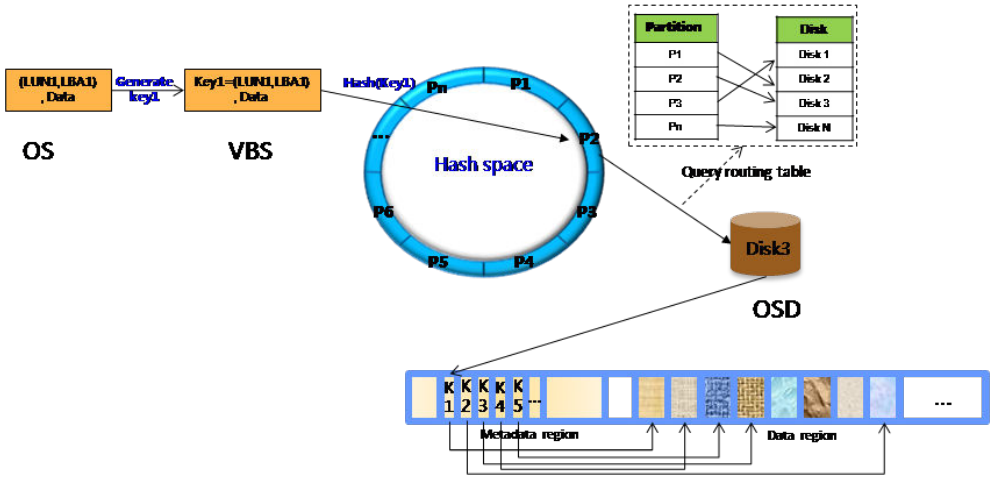
3.2.2.1 Data Routing

Data routing in FusionStorage Block is implemented in a hierarchical manner:

- The VBS identifies, by calculation, the server and the hard disk where data is stored.
- The OSD identifies, by calculation, the specific location on the hard disk.

The following figure shows the detailed process.

Figure 3-4 Data routing of FusionStorage Block



1. When the system is initialized, FusionStorage Block divides the hash space (0 to 2^32, available space of the resource pool in the unit of MB) into N equal portions. Each portion is a partition, and all these partitions are evenly allocated to hard disks in the system. For example, in a two-copy scenario, the system has 3600 partitions by default. If the system is equipped with 36 hard disks, each hard disk is allocated with 100 partitions. The partition-hard disk mapping is configured during system initialization and may be adjusted subsequently based on the change of the hard disk quantity. The mapping table requires only small space, and FusionStorage nodes store the mapping table in the memory for rapid routing. In addition, the relationship among the partition, primary disk, secondary disk 1, and secondary disk 2 (no secondary disk 2 when two copies are available) is determined based on the number of copies in the resource pool and other reliability configurations. The primary disk and secondary disks are deployed on different servers and even on different cabinets when a cabinet security plan is developed. The partitioning mechanism used in an EC scenario is the same as that used in a copy scenario. The partition-hard disk mapping is still used to manage hard disks. The difference is that the EC reliability is implemented through data disks and redundant disks.
2. FusionStorage Block logically divides a LUN by every 1 MB of space. For example, a LUN of 1 GB space is divided into 1024 slices of 1 MB space. When an upper-layer application accesses FusionStorage, the SCSI command carries the LUN ID, logical block addressing (LBA) ID, and read/write data content. The OS forwards the message to the VBS of the local node. The VBS generates a key based on the LUN ID and LBA ID. The key contains rounding information of the LBA ID based on the unit of 1 MB. An integer is obtained by hash calculation based on the DHT. The integer ranges from 0 to 2^32 and falls within the specified partition. The specific hard disk is identified based on the partition-hard disk mapping recorded in the memory. The VBS forwards the I/O operation to the OSD to which the hard disk belongs.
3. Each OSD manages a hard disk. During system initialization, the OSD divides the hard disk into slices of 1 MB and records the slice allocation information in the metadata management area of the hard disk. After receiving the I/O operation sent by the VBS, the OSD searches the hard disk by key for data slice information, obtains the data, and returns the data to the VBS. The entire data routing process is complete. For a write request on a copy, the OSD instructs each secondary OSD to perform the write operation based on the partition-primary disk-secondary disk 1-secondary disk 2 mapping table. Data is returned to the VBS after the primary and secondary OSDs complete the write operation.

3.2.2.2 Cache Mechanisms

FusionStorage Block uses hierarchical cache mechanisms to improve the I/O performance. Write and read cache mechanisms follow different processes.

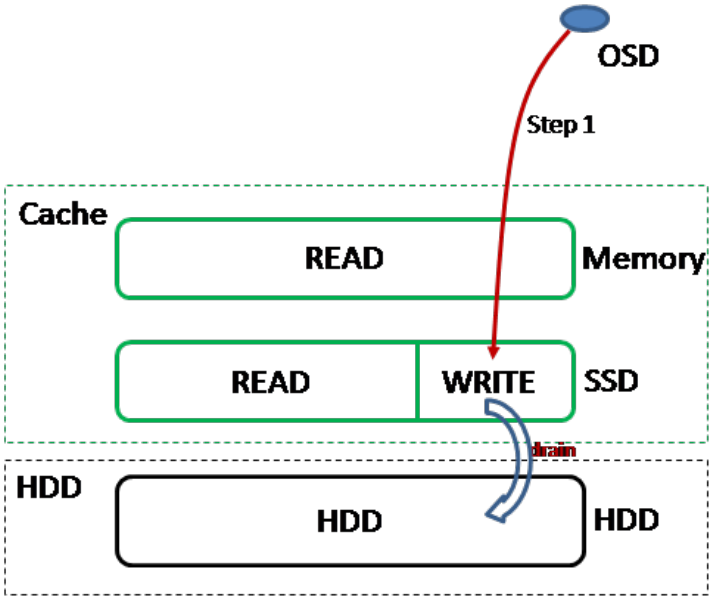
Write Cache

When receiving a write I/O request from the VBS, the OSD temporarily stores the write I/O in the SSD cache to complete the write operation on the local node. If multi-copy backup is used to protect data, the content in the SSD cache is complete I/O data. If EC is used to protect data, the content in the SSD cache is data or redundant strips. In addition, the OSD periodically writes the write I/O data from the SSD cache to hard drives in batches. A threshold is set for the write cache. If the threshold is reached, data will be written to hard drives even if the data refreshing period is not due. Eventually, all data will be written to hard drives.

NOTE

FusionStorage supports large-block write through. By default, data blocks greater than 256 KB will be written directly to hard drives rather than being cached. This configuration can be modified based on service requirements.

Figure 3-5 Write cache mechanism



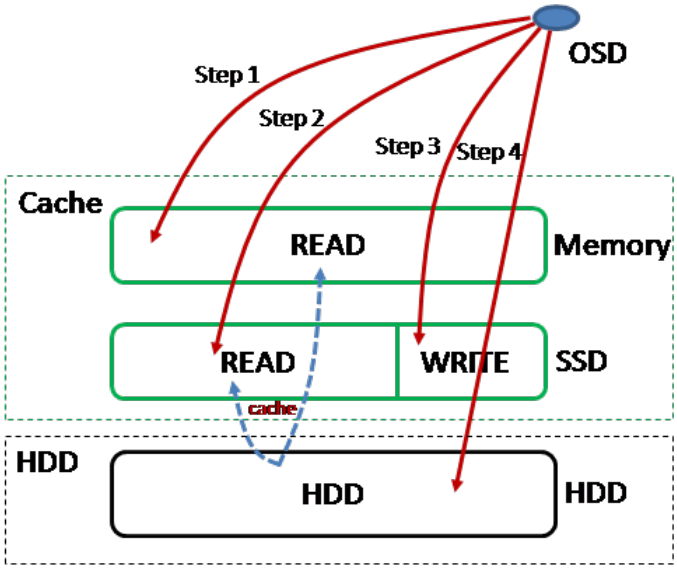
Read Cache

FusionStorage uses SSDs as read cache media to speed up storage access. FusionStorage adopts a hierarchical mechanism for read cache. The first layer is the memory cache, which caches data using the least recently used (LRU) mechanism. The second layer is the SSD cache, which functions based on the hotspot read mechanism. The system collects statistics on each piece of read data and the hotspot access factor. When the threshold is reached, the system automatically caches data to the SSD and removes the data that has not been accessed for a long time. In addition, FusionStorage supports the prefetch mechanism. It collects statistics on the correlation of read data and automatically reads the highly correlated blocks and caches them to the SSD when reading specific data.

When receiving a read I/O request from the VBS, the OSD performs the following operations:

1. Search the read cache of the memory for the required I/O data.
   - If the I/O data is found, return it to the VBS and move the I/O data to the LRU queue head of the read cache.
   - If the I/O data is not found, perform 2.
2. Search the read cache of the SSD for the required I/O data.
   - If the I/O data is found, return it directly in a copy scenario. In an EC scenario, obtain data blocks on other nodes, combine data by using the EC algorithm, return data, and add the hotspot access factor of the I/O data.
   - If the I/O data is not found, perform 3.
3. Search the write cache of the SSD for the required I/O data.
   - If the I/O data is found, the handling procedure varies depending on the data protection mechanism. If multi-copy backup is used, the found I/O data will be returned directly. If EC is used, the system obtains data blocks from other nodes, combines data using the EC algorithm, returns data, and adds the hotspot access factor of the I/O data. If the hotspot access factor reaches the threshold, the I/O data will be added to the read cache of the SSD.
   - If the I/O data is not found, perform 4.
4. Search the hard drives for the required I/O data. If multi-copy backup is used, the system returns the data directly. If EC is used, the system obtains data blocks from other nodes, combines data using the EC algorithm, returns data, and adds the hotspot access factor of the I/O data. If the hotspot access factor reaches the threshold, the I/O data will be added to the read cache of the SSD.

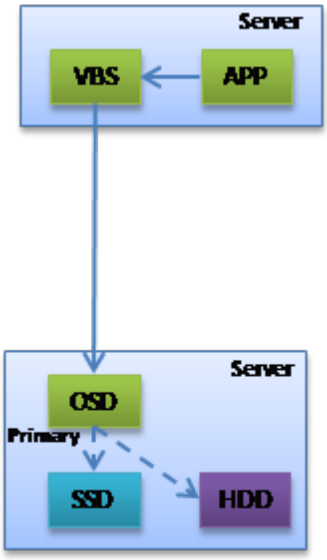
Figure 3-6 Read cache mechanism



Read I/O Process

Figure 3-7 shows the read I/O process of FusionStorage Block.

Figure 3-7 Read I/O process



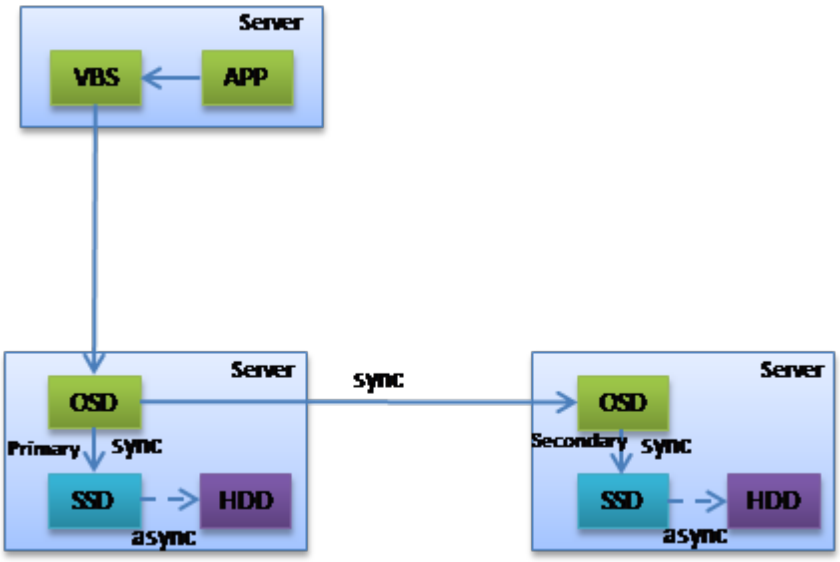
An application delivers a read I/O request to the OS. The OS forwards the read I/O request to the VBS of the local server. The VBS uses the data routing mechanism (for details, see **3.2.2.1 Data Routing**) to identify the primary OSD where the data is located based on the LUN and LBA information in the read I/O request. If the primary OSD is faulty, the VBS reads the data from the secondary OSD.

After receiving the read I/O request, the primary OSD obtains the required data based on the read cache mechanism (for details, see **3.2.2.2 Cache Mechanisms**) and returns a read I/O success message to the VBS.

Write I/O Process

Figure 3-8 shows the write I/O process of FusionStorage Block that uses two-copy backup mechanism.

Figure 3-8 Write I/O process (two-copy backup)



An application delivers a write I/O request to the OS. The OS forwards the write I/O request to the VBS of the local server. The VBS uses the data routing mechanism (for details, see **3.2.2.1 Data Routing**) to identify the primary OSD where the data is located based on the LUN and LBA information in the write I/O request.

After receiving the write I/O request, the primary OSD writes the data in both the SSD cache of the local server and the secondary OSD of the server where the data copy is located. The secondary OSD also writes the data in the SSD cache of the server where the secondary OSD is located. After the two write operations are successful, the primary OSD returns a write I/O success message to the VBS. In addition, the data in the SSD cache is asynchronously moved to the hard drives.

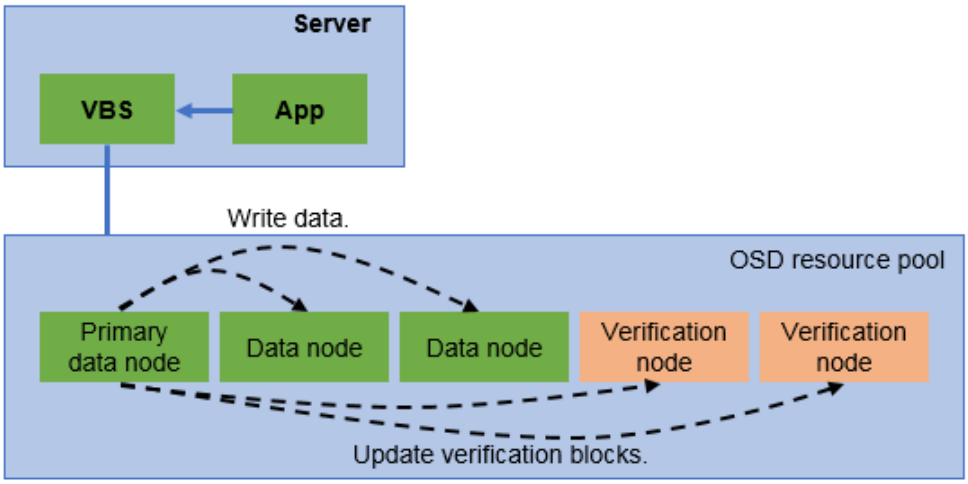
The VBS returns a write I/O success message.

NOTE

If three-copy backup is used, the primary OSD synchronizes the write I/O operation to both the secondary OSD and the third OSD.

The following figure shows the write I/O process of FusionStorage Block that uses the EC mechanism.

Figure 3-9 Write I/O process (EC mechanism)



An application delivers a write I/O request to the OS. The OS forwards the I/O request to the VBS of the local server. The VBS uses the data routing mechanism to determine the primary OSD where the data is located based on the LUN and LBA information in the write I/O request.

After receiving the write I/O request, the primary OSD converts the data to EC data blocks based on the data range of the volume, determines the nodes where the data blocks are located, writes the data blocks to the SSD cache of the corresponding nodes, calculates the verification node data, and writes the verification node data to the SSD cache of the corresponding verification node. In addition, the data in the SSD cache is asynchronously moved to the hard drives.

The VBS returns a write I/O success message.

3.2.2.3 Storage Cluster Management

FusionStorage Block uses the cluster management software to manage clusters. The cluster management software performs basic cluster information monitoring, performance monitoring, alarm management, user management, license management, and hardware management.

Table 3-4 Functions of FusionStorage Block

Name	Description
Basic cluster information monitoring	Allows users to view basic cluster information, including the cluster name, health status, running status, node information, and node process. Supports system sub-health monitoring management, which includes the following: ● Drive subhealth management: The management software periodically checks the drive SMART information, determines the drive subhealth status (check whether the number of drive sector remaps exceeds the threshold, whether the read error rate exceeds the threshold, and whether a slow drive exists), and isolates the drive and generates an alarm before it is damaged. ● Network subhealth management: The system generates alarms and starts automatic recovery when faults such as packet loss, error packet, long delay, and unmatched rate are detected on the network of a storage node. ● Storage node subhealth management: The distributed storage software automatically checks the storage node that has slower processing speed than other nodes, generates an alarm, and provides solutions.
Performance monitoring	Allows users to view the CPU usage, memory usage, bandwidth, IOPS, delay, disk usage, and storage pool usage statistics.
Alarm management	Allows users to view alarm information, clear alarms, and shield alarms.
User management	Enables the system administrator to create new administrators and grant them management permission for managing the system or resources. The administrator can query, delete, create, unlock, and freeze user accounts. Password policies can also be configured to enhance system security.
License management	Allows users to view activated licenses and import new licenses.

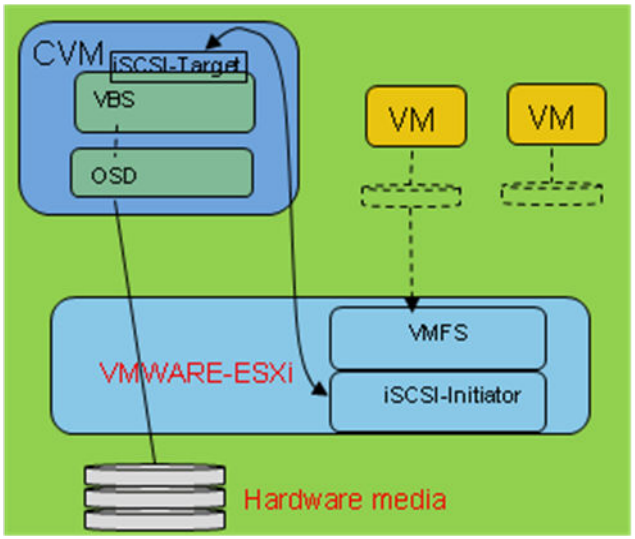
Name	Description
Hardware management	Allows users to perform hardware management, which includes the following: ● Server management: allows users to view server software installation status, software version, cluster information, and status and topology of storage pools created on servers, set the maintenance mode to facilitate fault handling, and monitor CPU and memory performance of servers. ● Unified hard drive management: allows users to view the hard drive status, slot number, SN, drive usage, and type, and collect statistics on the IOPS, delay, bandwidth, and utilization of hard drives.

3.2.3 Features

3.2.3.1 SCSI/iSCSI Block Interface

FusionStorage Block provides a SCSI or iSCSI using the VBS. The SCSI mode provides storage access for the host that runs the VBS. SCSI is used in physical deployment, FusionSphere, and KVM scenarios. The iSCSI mode provides storage access for virtual machines (VMs) or hosts other than the host that runs the VBS. This mode applies to VMware and Microsoft SQL Server clusters.

Figure 3-10 iSCSI block interface of FusionStorage Block



Regarding the SCSI protocol, SCSI-3 persistent reservations and non-persistent reservations are supported.

- Persistent reservations apply to HANA clusters.
- Non-persistent reservations apply to the Microsoft Clustering Service (MSCS).

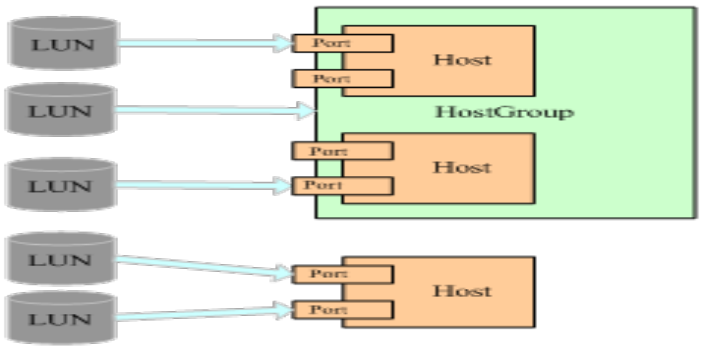
Regarding the iSCSI protocol, an iSCSI target is provided by the VBS. FusionStorage Block connects the local initiator to the iSCSI target to access the storage.

Secure access must be ensured in iSCSI mode. FusionStorage Block supports the following secure access standards:

- FusionStorage uses the Challenge Handshake Authentication Protocol (CHAP) identity verification to ensure that the client access is reliable and secure. This protocol periodically verifies the identity of the peer end by using a three-way handshake. The verification can be repeatedly performed when and after the initial link is established. CHAP provides protection against replay attacks from the peer end using an incrementally changing identifier and a variable challenge-value. It limits the time of exposure to an attack.
- LUN Masking is used to authorize a host to access LUNs. For a SAN storage device, hosts use LUNs as local storage devices, and therefore data maintenance is performed on the hosts. In this case, isolate hosts' access to LUNs, preventing one host from damaging the data of another host. LUN Masking binds LUNs to host bus adapter (HBA) world wide names (WWNs) and ensures that the LUNs can be accessed only by authorized hosts or host groups. The relationship between hosts and LUNs can be multiple-to-one or one-to-multiple. The one-to-multiple mapping meets storage requirements of small LUNs in virtualization scenarios, and the multiple-to-one mapping meets requirements of cluster systems such as Oracle RAC for shared volumes.

The core functions of LUN Masking are implemented by the mapping among ports, hosts, host groups, and LUNs.

Figure 3-11 Mapping between LUN Masking components



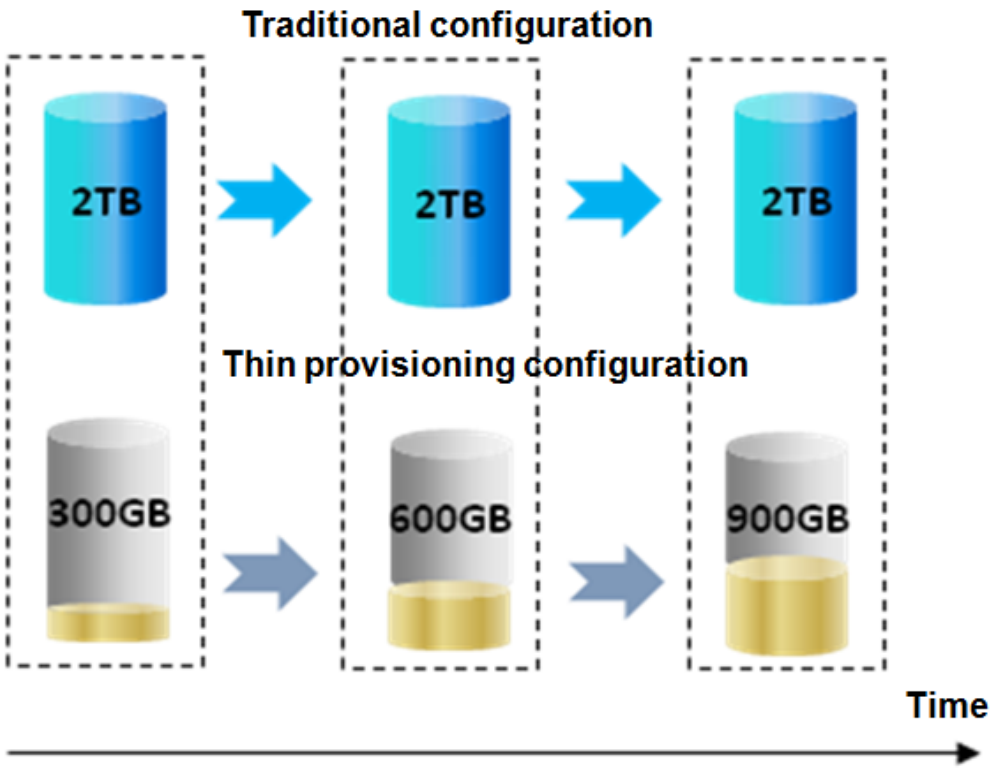
LUN Mapping binds LUNs to ports on storage devices so that hosts can access different LUNs using these ports. LUN Mapping can be used if a storage system concurrently provides data storage services for multiple applications and the hosts of these applications are located in different geographical areas.

3.2.3.2 Thin Provisioning

FusionStorage Block provides the thin provisioning function, which allows users to use much more storage space than that actually available on the physical storage device. This function significantly improves storage utilization compared with thick provisioning.

FusionStorage Block uses the DHT mechanism. No centralized metadata is required for recording thin provisioning information. Compared with traditional SAN storage devices, FusionStorage Block does not cause system performance deterioration.

Figure 3-12 Automatic thin provisioning mechanism

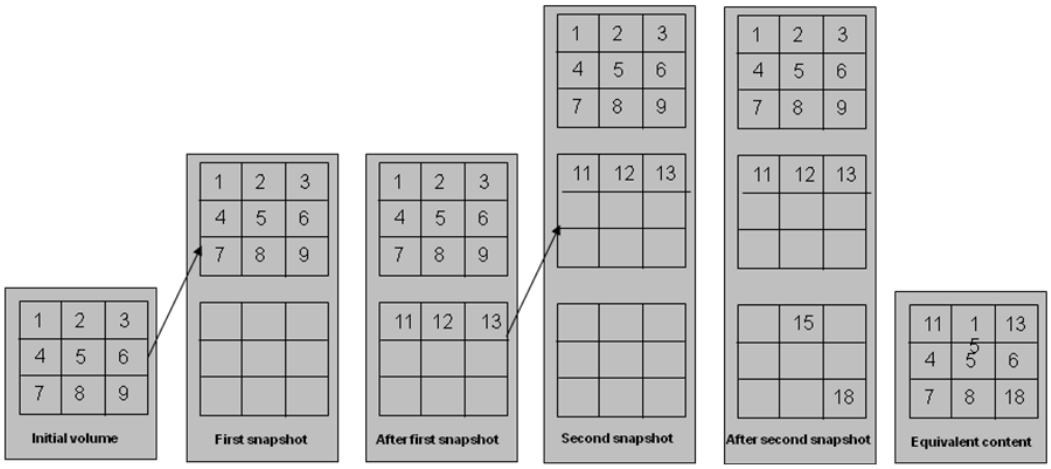


3.2.3.3 Snapshot

FusionStorage Block provides the snapshot mechanism, which allows the system to capture the status of the data written into a logical volume at a particular time point. The data snapshot can then be exported and used for restoring the volume data when required.

FusionStorage Block uses the redirect-on-write (ROW) technology when storing snapshot data. Snapshot creation does not deteriorate performance of the original volume.

Figure 3-13 Snapshot mechanism



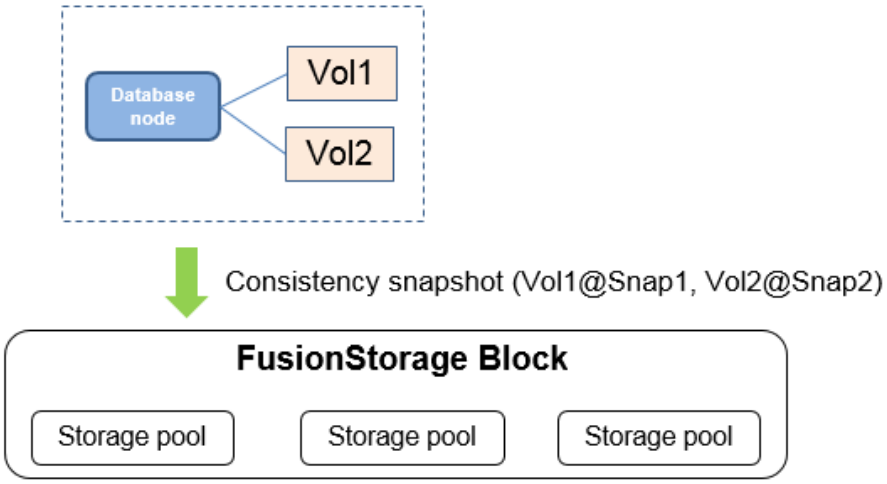
3.2.3.4 Consistency Snapshot

The consistency snapshot function is used for VM backup. A VM is usually mounted with multiple volumes. When a VM is backed up, all volume snapshots must be in the same time point to ensure data restoration reliability.

FusionStorage Block supports the consistency snapshot capability. Specifically, FusionStorage Block ensures that multiple volumes are snapshot at the same time point if an upper-layer application delivers a consistency snapshot request.

To ensure time consistency in snapshots of multiple volumes, FusionStorage Block implements I/O suspension for volumes and then updates the snapshot information operation.

Figure 3-14 Consistency snapshot mechanism



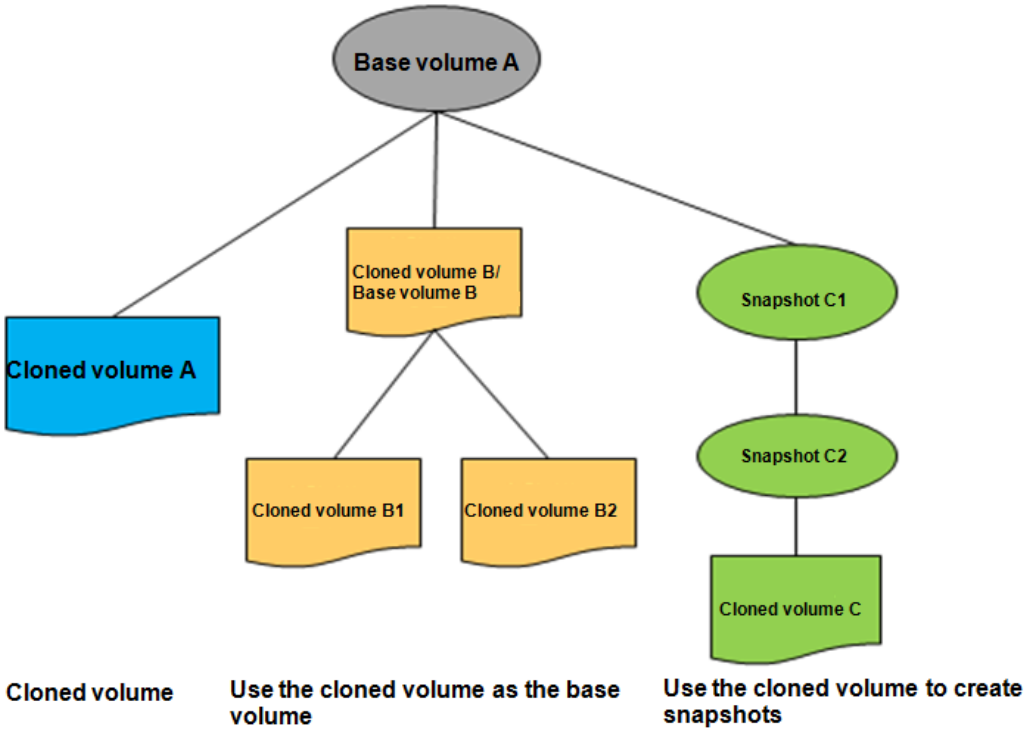
3.2.3.5 Linked Cloning

FusionStorage Block provides the linked cloning mechanism so that multiple cloned volumes can be created for a volume snapshot. The data in a cloned volume is the same as that in its snapshot. Subsequent modifications to a cloned volume do not affect its snapshot or other cloned volumes.

FusionStorage Block supports a linked cloning ratio of 1:256, which can significantly improve storage space utilization.

A cloned volume has all the functions of a common volume. You can create snapshots for a cloned volume, use the snapshot to restore the data in the cloned volume, and clone the data in the cloned volume.

Figure 3-15 Linked cloning mechanism



3.2.3.6 Multiple Resource Pools

To meet the requirements for storage media of different performance and for fault isolation, FusionStorage supports multiple resource pools. A set of FusionStorage Manager manages multiple resource pools. Multiple resource pools share a FusionStorage cluster, incuding the ZooKeepers and primary MDC in the cluster. Each resource pool has a home MDC. When a resource pool is created, an MDC automatically starts as the home MDC of the resource pool. The maximum quantity of resource pools is 128 and that of MDCs is 96. If there are more than 96 resource pools, the existing MDCs will be appointed as the home MDC for the excessive resource pools. Each MDC manages a maximum of 2 resource pools. The home MDC of a resource pool is responsible for the initialization of the resource pool. At the initialization stage, the storage resources are partitioned and the views of the partitions and OSDs are stored in a ZK disk. If the home MDC of a resource pool is faulty, the primary MDC appoints an MDC for the resource pool.

FusionStorage supports offline volume migration between multiple resource pools.

Multiple resource pools are planned according to the following rules:

- Multi-copy (two or three) backup or erasure code (EC, with a redundancy ratio of 2+2, 3+2, 4+2, 8+2, or 12+3) can be used to protect data.
- If two-copy backup is used, a resource pool supports a maximum of 96 hard disks. If three-copy backup is used, a resource pool supports a maximum of 2048 hard disks. For more hard disks, a new resource pool must be planned.
- If EC (with a redundancy ratio of 2+2, 3+2, 4+2, 8+2, or 12+3) is used, a resource pool supports a maximum of 2048 hard disks.
- The hard disks in a resource pool are of the same type. Hard disks of different types are assigned to different resource pools. The hard disks in a resource pool are of the same

capacity; where their capacities are different, they are used as if their capacities are of the smallest.

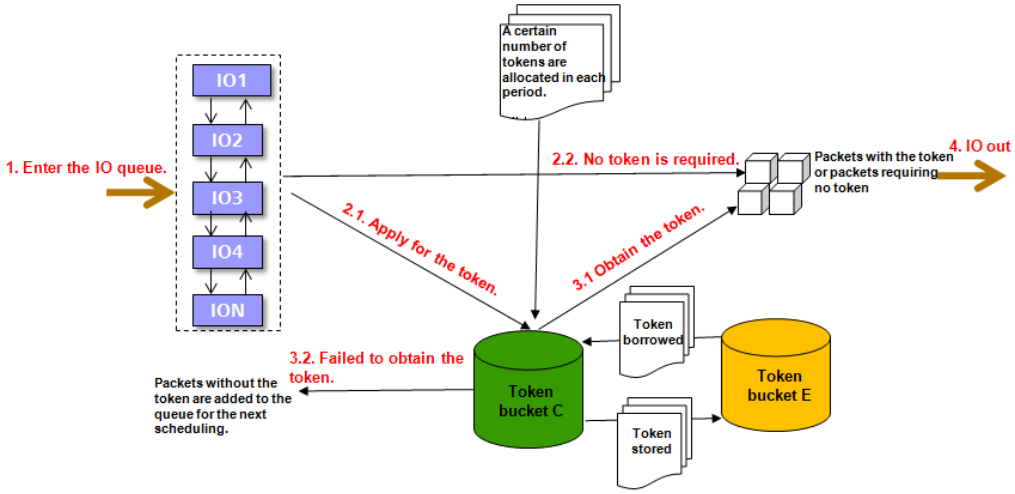
- The cache media in a resource pool are of the same type. Cache media of different types are assigned to different resource pools.
- It is recommended that each storage node in a resource pool have the same number of hard disks. The gap between hard disk quantities on different nodes cannot exceed 2, and the proportion of the gap to the maximum number of hard disks on a node cannot be greater than 33%.

3.2.3.7 QoS

The FusionStorage Block QoS feature provides refined I/O control for volumes and provides the burst function. The burst function means that when the volume requirement exceeds the baseline IOPS (bandwidth), the quota that exceeds the benchmark performance can be used within a certain period.

The QoS feature adopts the dual token bucket algorithm to implement I/O control, bandwidth control, and burst for volumes.

Figure 3-16 QoS mechanism



The token bucket C is used for I/O control. The token bucket E is used to store the remaining tokens. The two buckets work together to implement the burst function.

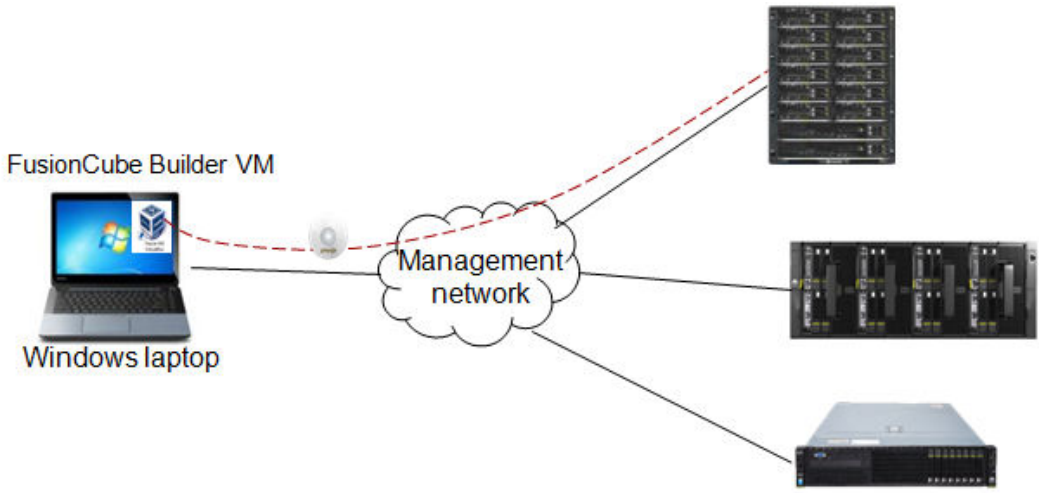
3.3 Automatic Deployment

FusionCube provides the quick installation and deployment tool FusionCube Builder (FCB for short) to install system software. FusionCube supports one-click system initialization. After basic system parameters are configured, the network configuration of each node is automatically completed, and management and storage clusters are created.

3.3.1 FusionCube Builder

The FusionCube Builder (FCB) is a tool used to install and configure FusionCube.

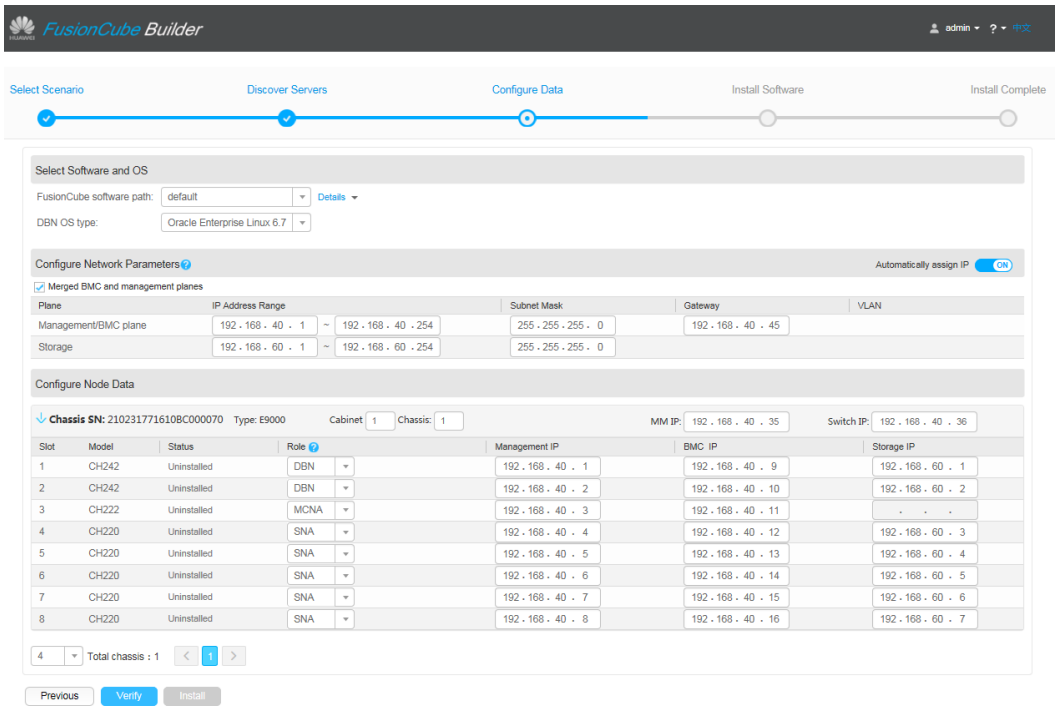
Figure 3-17 FCB-based software installation



- The FCB can be installed on a PC or a VM.
- The FCB detects servers by using the Simple Service Discovery Protocol (SSDP) or scanning server IP addresses.
- The FCB connects to the iBMC and uses the virtual DVD drive of the KVM for software installation. The FCB supports concurrent installation of eight nodes.
- The FCB uses NFS to share and transfer software packages and configurations during the installation process.

The FCB provides installation guidance on a unified configuration GUI to help you quickly complete parameter settings. It then automatically installs related software based on your configuration.

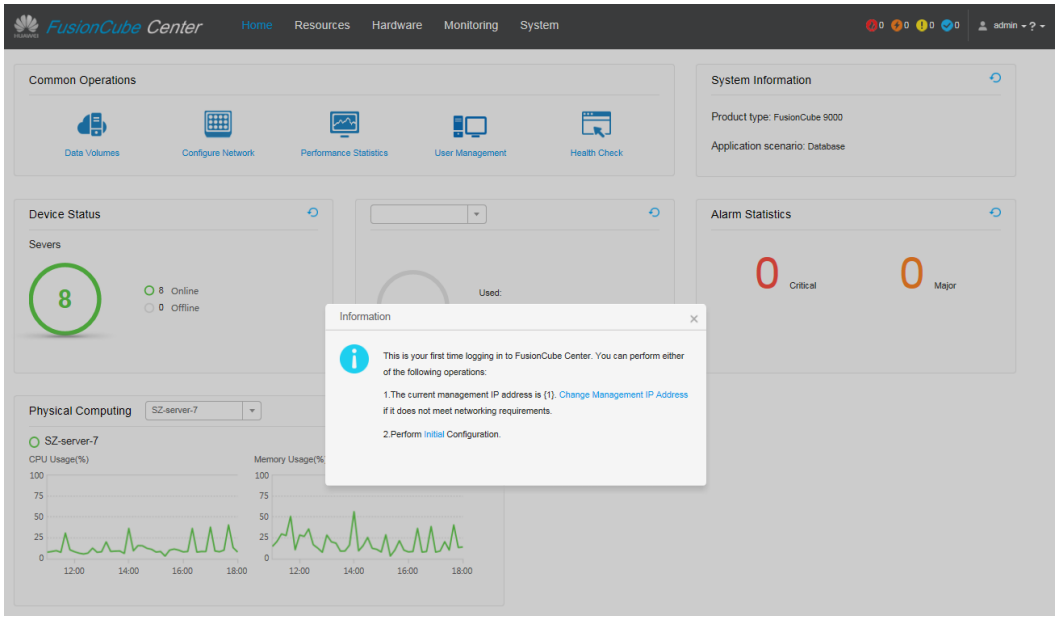
Figure 3-18 FCB installation guidance and configuration GUI



3.3.2 System Initialization

At your initial login to the FusionCube Center, you can manage the system management IP addresses and the system initialization function.

Figure 3-19 Initial login screen of FusionCube Center



The system initialization process is as follows:

1. Access the system initialization screen. The system automatically detects devices and displays the node information on the initialization parameter configuration screen.

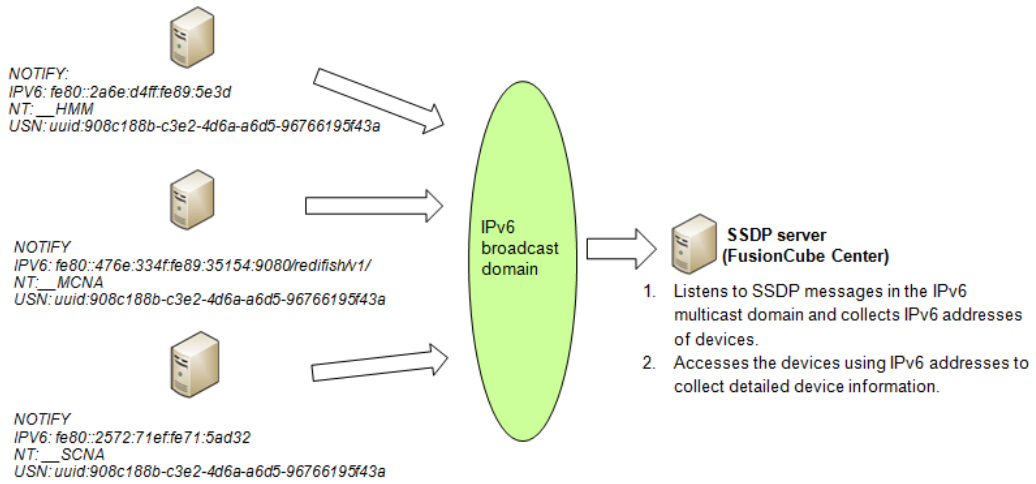
2. Configure the initialization parameters (including network parameters and storage block parameters). System initialization can be started once the parameters are successfully verified.
3. The system automatically completes the initialization, including network configuration of various nodes and creation of management clusters, storage clusters, and storage pools.

You can manage data volumes once the system initialization is complete. After the database is installed, the system is available for service if you mount data volumes to database applications and complete uplink, network, and NTP configurations.

3.3.3 Automatic Device Discovery

FusionCube supports automatic device discovery during system installation, initialization, and capacity expansion. Automatic device discovery is implemented using the Simple Service Discovery Protocol (SSDP) service.

Figure 3-20 Automatic device discovery



The process of automatic device discover during system installation is as follows:

1. The iBMC or MM used by FusionCube is embedded with the SSDP. After being powered on, devices automatically broadcast SSDP messages through IPv6 addresses.
2. The SSDP server deployed on the FCB monitors the SSDP messages in the IPv6 broadcast domain and collects IPv6 addresses of the corresponding devices.
3. The FCB logs in to the devices using the respective IPv6 addresses to obtain detailed device information.

The process of automatic device discover during system initialization and capacity expansion is as follows:

1. After the system is installed, the SSDP client is embedded in the management VM, CVM, and host OS. The SSDP client automatically broadcasts SSDP messages through IPv6 addresses.
2. The SSDP server deployed on the FCB monitors the SSDP messages in the IPv6 broadcast domain on the management plane and collects IPv6 addresses of the corresponding devices.

3. The FCB logs in to the devices using the respective IPv6 addresses to obtain detailed device information.

3.4 Unified O&M Management

3.4.1 Introduction to the Management System

Using FusionCube Center, FusionCube can manage the entire system in a unified manner, including resource management, performance monitoring, alarm management, operation log management, permission management, hardware management, health check, and log collection.

Figure 3-21 FusionCube Center home page

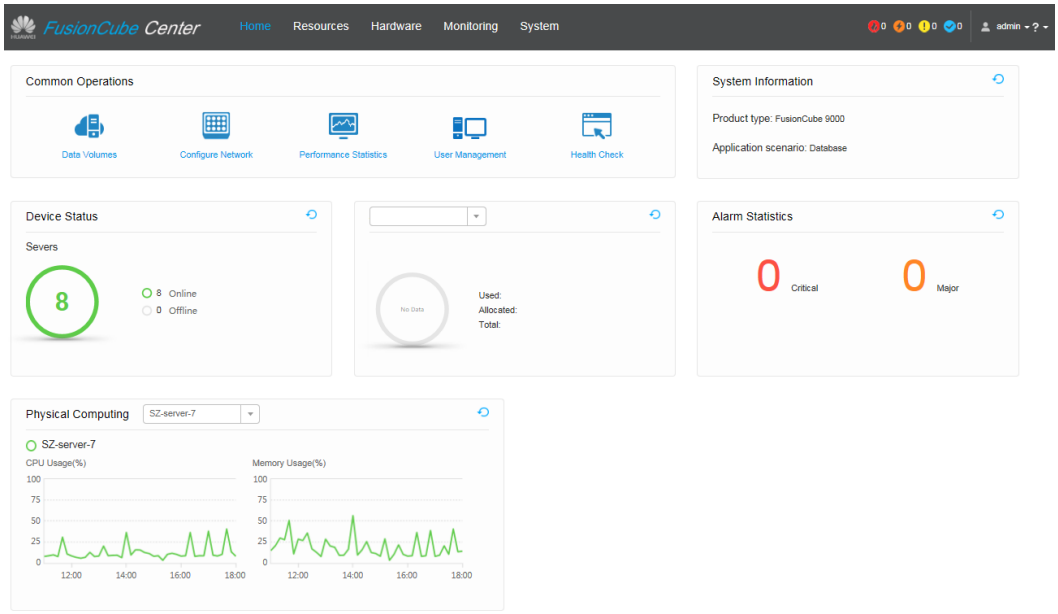


Table 3-5 Functions provided on the FusionCube Center home page

Function	Description
Resource management	Accesses and manages storage and physical computing resources, and manages data volumes by providing functions including volume creation and deletion, and batch mounting and unmounting.
Performance monitoring	Collects statistics on the CPU usage, memory usage, network traffic, and disk I/O of resource clusters and servers.
Alarm management	Provides the functions of viewing alarm information, clearing alarms, and shielding alarms.
Operation log management	Allows users to view and export operation logs.

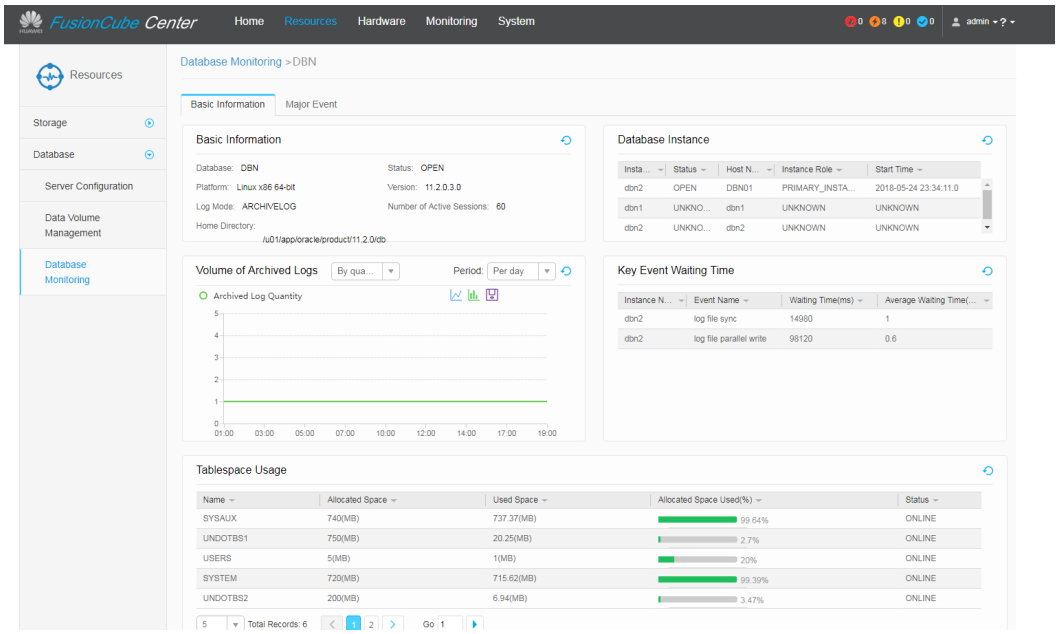
Function	Description
Permission management	Allows users to create and delete users and change passwords, and supports user role configuration, password policy configuration, and domain name authentication and configuration.
Hardware management	Supports access management and modification of the names and IP addresses of chassis, servers, and switches.
Health check	Supports system check and hardware compatibility check for nodes.
Log collection	Collects logs of FusionCube Center and various nodes of FusionStorage Block, and provides log package download functions on the GUI.

3.4.2 Database Monitoring

FusionCube Center also supports Oracle database access and monitoring, and displays database status and running alarms in a unified manner. In addition, the tablespace usage, SQL statement running efficiency, and archiving volume can also be monitored. The functions of database monitoring are as follows:

- Displays basic information and running status of an Oracle database.
- Monitors the tablespace usage.
- Displays database alarms in a centralized manner and transfers alarms to emails.
- Tracks and monitors the running efficiency of SQL statements, and provides SQL statement optimization suggestions based on statistical analysis.
- Monitors the archiving volume of each hour in the last two days and provides the archiving volume trend chart of each day in the last two months.
- Monitors the delay of **Log File Sync**/**Log File Parallel Write**.

Figure 3-22 Database monitoring page



3.5 Hardware Platforms

FusionCube is compatible with multiple compute and storage converged hardware platforms, including the E9000 blade server platform and rack server platform.

3.5.1 E9000 Blade Servers

3.5.1.1 E9000 Server

Huawei E9000 is a 12U blade server that comes with compute nodes, switch modules, and management modules in flexible configuration.

The E9000 has the following functions and features:

- Supports up to 8 full-width or 16 half-width compute nodes in flexible configuration.
- Provides 850 W cooling capacity for a half-width slot.
- Provides 1700 W cooling capacity for a full-width slot.
- Supports up to 2 processors and 24 DIMMs for a half-width compute node.
- Supports up to 4 processors and 48 DIMMs for a full-width compute node.
- Supports a maximum of 32 processors with a maximum memory capacity of 24 TB in one chassis.
- Provides a maximum switch capacity of 5.76 Tbit/s for the midplane.
- Provides two pairs of slots for pass-through or switch modules that support multiple switching protocols such as Ethernet and IB and provide a variety of ports.

Figure 3-23 E9000 appearance

NOTE

FusionCube supports three E9000 chassis in one cabinet.

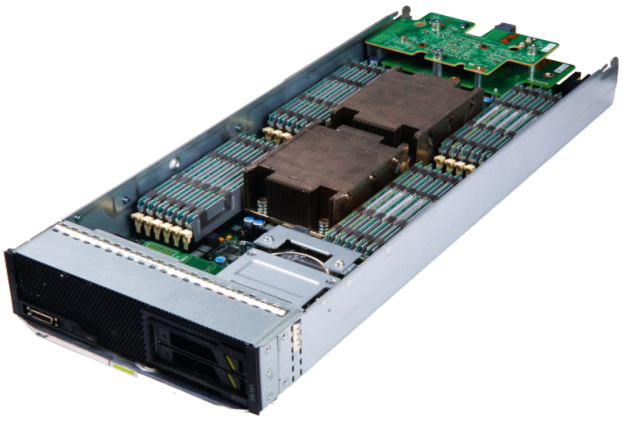
3.5.1.2 E9000 Compute Nodes

FusionCube supports the following E9000 compute nodes:

- 2-socket CH121 V3 compute node
- 2-socket CH220 V3 compute and I/O expansion node
- 2-socket CH222 V3 compute and storage node

- 2-socket CH225 V3 compute and storage node
- 4-socket CH242 V3 compute node
- 2-socket CH121 V5 compute node
- 2-socket CH225 V5 compute and storage node
- 4-socket CH242 V5 compute node

Figure 3-24 CH121 V3 compute node



Form factor	Half-width single-slot 2-socket blade server
Processor	1 or 2 Intel® Xeon® E5-2600 v3/v4 processors
DIMM slot	24 DDR4 DIMM slots
Hard drive	2 x 2.5-inch SAS/SATA drives, or 2 x PCIe SSDs
RAID support	RAID 0 and RAID 1
Built-in flash memory	2 microSD cards, 2 SATADOMs, and 1 USB flash drive (USB 3.0)
PCIe expansion	● Two PCIe x16 mezzanine cards ● One standard FHHL PCIe x16 card

Figure 3-25 CH220 V3 compute and I/O expansion node

Form factor	Full-width single-slot 2-socket blade server
Processor	1 or 2 Intel® Xeon® E5-2600 v3/v4 processors
DIMM slot	16 DDR4 DIMM slots
Drive	2 x 2.5-inch SAS/SATA drives, or 2 x PCIe SSDs
RAID support	RAID 0 and RAID 1
Built-in flash memory	2 microSD cards, 2 SATADOMs, and 1 USB flash drive (USB 3.0)
PCIe expansion	● 4 mezzanine cards (two x16 and two x8) ● 6 standard PCIe 3.0 x16 cards in any of the following combinations: – 6 FHHL single-slot cards – 1 FHFL dual-slot card + 4 FHHL single-slot cards – 2 FHFL dual-slot cards

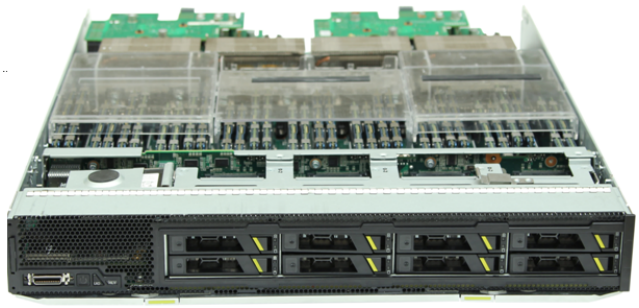
Figure 3-26 CH225 V3 compute and storage node

Form factor	Full-width single-slot 2-socket blade server
Processor	1 or 2 Intel® Xeon® E5-2600 v3/v4 processors
DIMM slot	24 DDR4 DIMM slots
Drive	12 x 2.5-inch NVMe SSDs and 2 x 2.5-inch SSDs or SAS/SATA HDDs
RAID support	RAID 0 and RAID 1
Built-in flash memory	2 microSD cards, 2 SATADOMs, and 1 USB flash drive (USB 3.0)
PCIe expansion	4 PCIe x16 mezzanine cards

Figure 3-27 CH222 V3 compute and storage node

Form factor	Full-width single-slot 2-socket blade server
Processor	1 or 2 Intel® Xeon® E5-2600 v3/v4 processors
DIMM slot	24 DDR4 DIMM slots
Drive	15 x 2.5-inch SSDs or SAS/SATA HDDs
RAID support	RAID 0 and RAID 1
Built-in flash memory	2 microSD cards, 2 SATADOMs, and 1 USB flash drive (USB 3.0)
PCIe expansion	● 2 PCIe x16 mezzanine cards ● 1 standard FHHL PCIe x16 card

Figure 3-28 CH242 V3 compute blade



Form factor	Full-width single-slot 4-socket blade server
Processor	2 or 4 x Intel® Xeon® E7 v2/E7 v3 processors (full series), up to 18 cores, 165 W

DIMM slot	Intel® Xeon® E7 v2 processors: 32 DDR3 DIMM slots, with a maximum bandwidth of 1600 MHz Intel® Xeon® E7 v3 processors: 32 DDR4 DIMM slots, with a maximum bandwidth of 1866 MHz
Drive	8 SSDs or SAS/SATA HDDs
RAID support	RAID 0 and RAID 1
Built-in flash memory	2 microSD cards and 1 USB flash drive (USB 3.0)
PCIe expansion	● 4 PCIe x16 mezzanine cards ● 2 standard FHHL PCIe x16 cards

Figure 3-29 CH121 V5 compute blade

Form factor	Half-width single-slot 2-socket blade server
Processor	1 or 2 Intel® Xeon® scalable processors
DIMM slot	24 DDR4 DIMM slots
Drive	2 x 2.5-inch SAS or SATA HDDs, or 2 x PCIe SSDs
RAID support	RAID 0 and RAID 1
Built-in flash memory	Up to 4 M.2 SSDs (SATA ports)
PCIe expansion	● 2 PCIe x16 mezzanine cards ● 1 standard FHHL PCIe x16 card

Figure 3-30 CH225 V5 compute and storage node

Form factor	Full-width single-slot 2-socket blade server
Processor	1 or 2 Intel® Xeon® scalable processors
DIMM slot	24 DDR4 DIMM slots
Drive	● 12 x 2.5-inch SATA/SAS/NVMe HDDs or SSDs, mixed configuration supported ● 2 x 2.5-inch SAS/SATA HDDs or SSDs
RAID support	RAID 0 and RAID 1
Built-in flash memory	Up to 6 M.2 SSDs (including 2 LOMs)
PCIe expansion	4 PCIe x16 mezzanine cards

Figure 3-31 CH242 V5 compute node

Form factor	Full-width single-slot 4-socket blade server
Processor	2 or 4 Intel® Xeon® scalable processors
DIMM slot	48 DDR4 DIMM slots, with a maximum memory speed of 2666 MT/s

Drive	4 x 2.5-inch SSDs or SAS/SATA HDDs, or 4 x NVMe SSDs, or a maximum of 8 x M.2 SSDs (SATA ports). Hot swap of a single drive is supported.
RAID support	RAID 0 and RAID 1
Built-in flash memory	2 microSD cards and 1 USB flash drive (USB 3.0)
PCIe expansion	● 4 PCIe x16 mezzanine cards ● 1 standard HHHL PCIe x16 card

3.5.1.3 High-Performance Switch Modules

FusionCube uses CX310 switch modules, which support 10GE network. Each chassis can be configured with two switch modules.

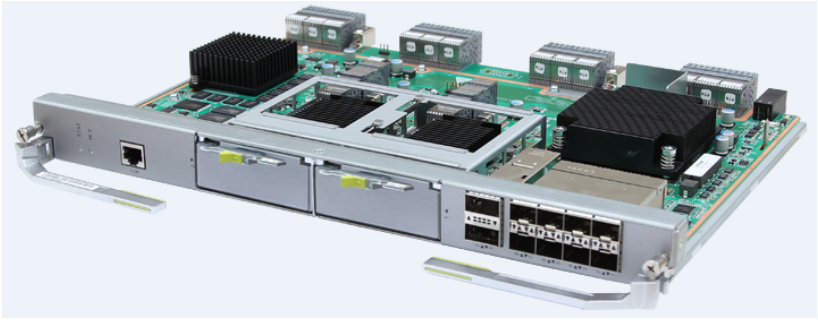
Figure 3-32 CX310 10GE switch module

Model	CX310 10GE switch module
Network port	16 x 10GE uplink ports 32 x 10GE downlink ports
Network feature	L2: VLAN/MSTP/LACP/TRILL/Stack/IGMP L3: RIP/OSPF/ISIS/BGP/VRRP/BFD/PIM QoS: DCBX/PFC/ETS/ACL/CAR/DiffServ Security: IPSG/MFF/DAI/FSB/DHCP Snooping
Management port	2 x RS232 management serial ports (one for service management and one for device management)

Figure 3-33 CX611 InfiniBand switch module

Model	CX611 IB switch module
Network port	18 x QDR or FDR uplink ports 16 x 4X QDR or FDR downlink ports (one 4X QDR or FDR downlink port in each half-width slot)
Network feature	QDR/FDR auto-negotiation Ideal for applications demanding low delay and high bandwidth.
Management port	2 x RS232 management serial ports (one for service management and one for device management)

Figure 3-34 CX320 10GE switch module



Model	CX320 10GE switch module
Network port	8 x 10GE uplink ports and 2 x 40GE uplink ports 32 x 10GE downlink ports
Network feature	L2: VLAN/MSTP/LACP/TRILL/Stack/IGMP L3: RIP/OSPF/ISIS/BGP/VRRP/BFD/PIM QoS: DCBX/PFC/ETS/ACL/CAR/DiffServ Security: IPSG/MFF/DAI/FSB/DHCP Snooping

Management port	2 x RS232 management serial ports (one for service management and one for device management)

Figure 3-35 CX620/CX621 IB switch module

Model	CX620/CX621 IB switch module
Network port	18 x FDR or EDR uplink ports 16 x FDR/EDR downlink ports
Network feature	FDR/EDR autonegotiation Ideal for applications demanding low delay and high bandwidth.
Management port	1 x RS232 management serial port

3.5.2 Rack Servers

3.5.2.1 RH2288H V3

Figure 3-36 RH2288H V3 compute and storage server (with 12 hard disks)

Form factor	2-socket rack server
Processor	1 or 2 x Intel® Xeon® E5-2600 v3/v4 processors
DIMM slot	24 x DDR4 DIMM slots
Hard disk	12 x 3.5-inch SATA/NL-SAS HDDs + 2 x 2.5-inch SAS HDDs
RAID support	RAID 0 and RAID 1
Built-in flash	2 x microSD cards, 2 x SATADOMs, and 1 x USB flash drive (USB 3.0)
PCIe expansion	● 1 x LOM, with 2 x GE ports, 4 x GE ports, or 2 x 10GE ports ● 4 x standard PCIe slots (supporting 2 x standard FHHL cards and 2 x standard HHHL cards)

3.5.2.2 RH5885H V3

Figure 3-37 RH5885H V3 compute server

Form factor	4-socket rack server
Processor	2 or 4 Intel® Xeon® E7 v3/v4 processors
DIMM slot	96 x DDR4 DIMM slots
Hard disk	8 x 2.5-inch SAS/SATA/NL-SAS HDDs

RAID support	RAID 0 and RAID 1
LOM	4 x GE ports, 2 x GE ports, or 2 x 10GE ports
PCIe expansion	17 x PCIe slots (one for the RAID controller card and four supporting hot swap)

3.5.2.3 1288H V5

Figure 3-38 1288H V5 compute server

Form factor	2-socket rack server
Processor	1 or 2 x Intel® Xeon® scalable processors
DIMM slot	24 x DDR4 DIMM slots, with a maximum memory speed of 2666 MT/s
Hard disk	8 x 2.5-inch SAS/SATA HDDs or SSDs
RAID support	RAID 0 and RAID 1
LOM	Up to 3 x standard PCIe slots
PCIe expansion	1 x LOM, with 2 x GE ports and 2 x 10GE ports 1 x flexible LOM, with 2 x GE ports, 4 x GE ports, or 2 x 10GE ports 2 x standard HHHL PCIe x16 slots and 1 x standard FHHL PCIe x8 slot

3.5.2.4 2288H V5

Figure 3-39 2288H V5 compute and storage server

Form factor	2-socket rack server
Processor	1 or 2 x Intel® Xeon® scalable processors
DIMM slot	24 x DDR4 DIMM slots, with a maximum memory speed of 2666 MT/s
Hard disk	8 x 2.5-inch SAS/SATA/NL-SAS HDDs or SSDs 12 or 16 x 3.5-inch SAS/SATA/NL-SAS HDDs or SSDs 25 x 2.5-inch SAS/SATA/NL-SAS HDDs or SSDs 12 or 24 x NVMe SSDs
RAID support	RAID 0 and RAID 1
LOM	2 x GE ports and 2 x 10GE ports
PCIe expansion	1 x flexible LOM, with 2 x GE ports, 4 x GE ports, or 2 x 10GE ports or 1/2 x 56 Gbit/s FDR IB ports Up to 8 x standard PCIe slots: 4 x standard FHFL PCIe 3.0 x16 cards (bandwidth: x8), 3 x standard FHHL PCIe 3.0 x16 cards (bandwidth: x8), and 1 x standard FHHL PCIe 3.0 x8 standard card (bandwidth: x8), 1 x RAID controller card, and 1 x flexible LOM

3.5.2.5 2488H V5

Figure 3-40 2488H V5 compute server

Form factor	2-socket rack server
Processor	2 or 4 x Intel® Xeon® scalable processors
DIMM slot	48 x DDR4 DIMM slots
Hard disk	8 x 2.5-inch SAS/SATA HDDs
RAID support	RAID 0 and RAID 1
LOM	2 × GE + 2 x 10GE optical or electrical ports
PCIe expansion	3 x standard HHHL PCIe 3.0 x16 cards and 7 x standard HHHL PCIe 3.0 x8 cards

3.5.3 Typical Configuration

Type	Model	Specifications	Function	CPU	Memory	Storage	Network
Management	CH121 V5	Half-width blade	2-socket blade management node	1 x scalable processor	24 x DDR4 DIMMs	2 x 2.5" SAS system disks	10GE

node	1288H V5	1U rack	2-socket rack management node	1 x scalable processor	24 x DDR4 DIMMs	2 x 2.5" SAS system disks	GE
Compute node	CH121 V5	Half-width blade	2-socket blade compute node	2 x scalable processors	24 x DDR4 DIMMs	2 x 2.5" SAS system disks	10GE 56 Gbit/s or 100 Gbit/s IB
	CH242 V5	Full-width blade	4-socket blade compute node	4 x scalable processors	48 x DDR4 DIMMs	2 or 4 x 2.5" SAS system disks	10GE 56 Gbit/s or 100 Gbit/s IB
	1288H V5	1U rack	2-socket rack management node	2 x scalable processors	24 x DDR4 DIMMs	2, 4, 6, or 8 x 2.5" SAS system disks	GE, 10GE 56 Gbit/s or 100 Gbit/s IB
	2288H V5	2U rack	2-socket rack management node	2 x scalable processors	24 x DDR4 DIMMs	2, 4, 6, or 8 x 2.5" SAS system disks	GE, 10GE 56 Gbit/s or 100 Gbit/s IB
	2488H V5	2U rack	4-socket rack management node	4 x scalable processors	48 x DDR4 DIMMs	2, 4, 6, or 8 x 2.5" SAS system disks	GE, 10GE 56 Gbit/s or 100 Gbit/s IB

Storage node	CH225 V5	Full-width blade	2-socket blade storage node	2 x scalable processors	24 x DDR4 DIMMs	11 x SAS data disks + 1 x NVMe SSD cache + 2 x 2.5" SAS system disks 12 x NVMe SSD data disks + 2 x 2.5" SAS system disks	10GE 56 Gbit/s or 100 Gbit/s IB
	2288H V5	2U rack	2-socket rack management node	2 x scalable processors	24 x DDR4 DIMMs	12 or 16 x SAS/SATA/NL-SAS data disks + 4 x NVMe SSD caches + 2 x 2.5" SAS system disks 12 x NVMe SSD data disks + 2 x 2.5" SAS system disks	GE, 10GE 56 Gbit/s or 100 Gbit/s IB
Switch device	Storage network: 56 Gbit/s or 100 Gbit/s IB network Management or service network: GE or 10GE network						

Note: Only Huawei ES3000 NVMe SSDs and Intel Optane NVMe SSDs are supported.

3.6 Networking Schemes

The FusionCube network is logically divided into three planes: service plane, storage plane, and management plane. The three planes are isolated to protect FusionCube from external attacks.

Table 3-6 Description of the three planes

Plane	Description
Storage plane	Storage devices on the server communicate with each other over layer 2 of the storage plane. Storage devices provide storage resources for virtual machines (VMs) through the virtualization platform but do not communicate with VMs directly.

Plane	Description
Service plane	The service plane provides a channel for users to obtain services, for virtual NICs of VMs to communicate with each other, and for external applications to interact with FusionCube. Access can be separated by the VLANs configured for VMs.
Management plane	The management plane carries the traffic of system management, service deployment, and system loading. The BMC plane is responsible for server management. It can be independent of the management plane and uses a network segment different from that of the management plane. The BMC plane can also be integrated with the management plane and uses the same network segment as the management plane as long as it is interconnected with layer 2 of the management plane.

The service, management, and storage planes can be converged at a port and logically isolated through VLANs and can also be physically isolated through independent network ports.

3.6.1 Internal Network

The following figure shows the internal network of a FusionCube database infrastructure.

Figure 3-41 Internal network

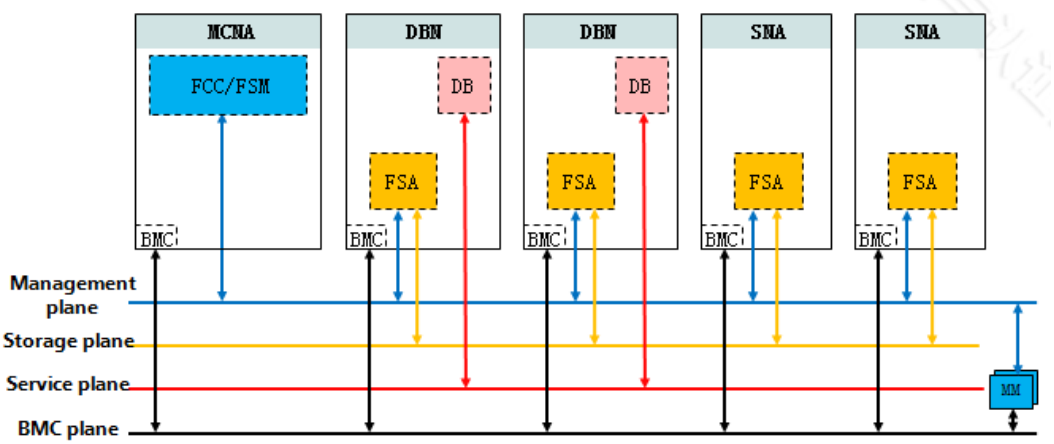


Table 3-7 Node types

Node	Description	Deployment Principle
Management Computing Node Agent (MCNA)	A host that manages the system. FusionCube Center and FusionStorage Block Manager are deployed on the MCNA.	One or two MCNAs can be deployed based on service requirements.

Node	Description	Deployment Principle
Database Node (DBN)	A node in the database cluster that provides database services.	Two or more DBNs can be deployed based on service requirements.
Storage Node Agent (SNA)	A node that provides storage functions but not virtualization resources. SNAs and SCNAs cannot be deployed together in one system.	Three or more SNAs can be deployed based on service requirements.

Table 3-8 Communication plane types

Plane	Description
Management plane	Device management plane
BMC plane	Server hardware management plane
Storage plane	Internal communication plane of FusionStorage Block
Service plane	Any service-related communication plane except the BMC, management, and storage planes. This plane can be divided into one or multiple networks based on service requirements.

NOTE

The management and BMC planes can share the same network segment or use different network segments.

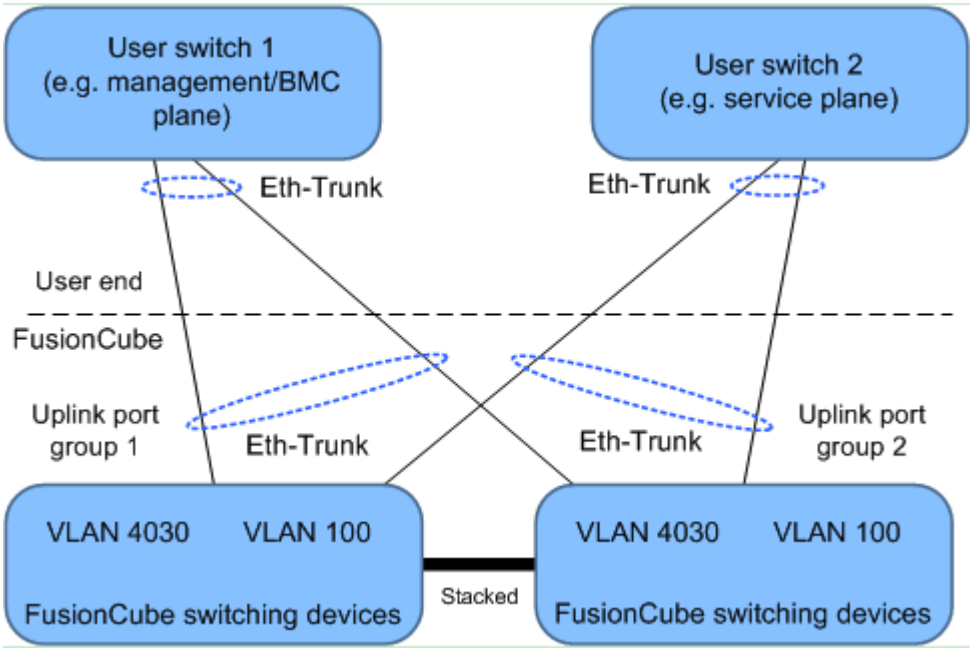
3.6.2 Interconnection Network

FusionCube connects to external networks through a layer 2 network in multiple interconnection modes.

Physical Isolation Between Management and Service Planes

Planes on external networks, for example, the management and service planes, are physically isolated by different switches.

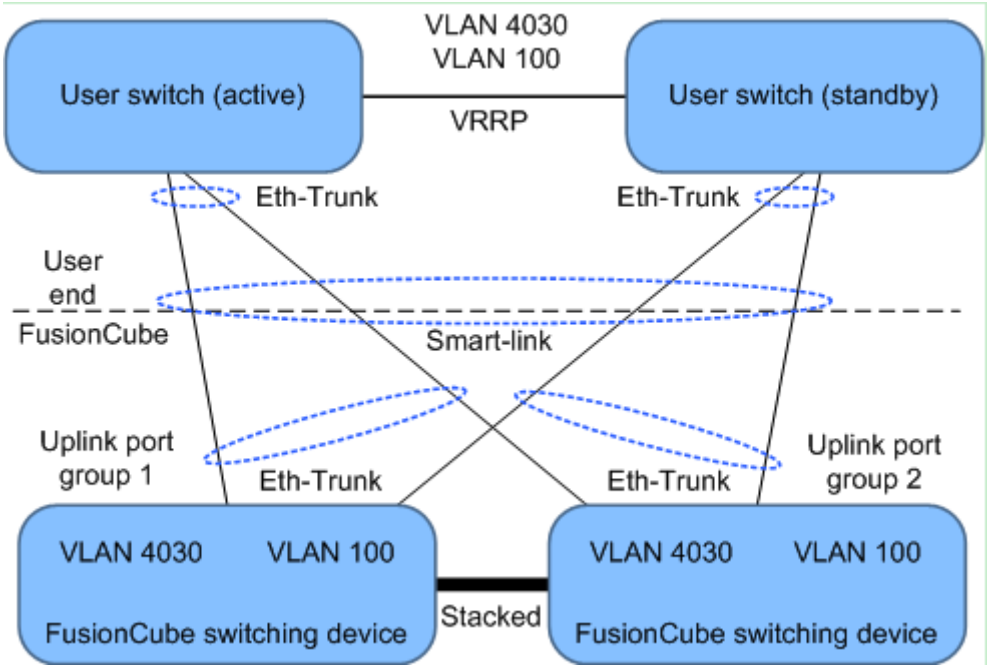
Figure 3-42 Uplink interconnection when management and service planes are physically isolated



Convergence of Management and Service Planes

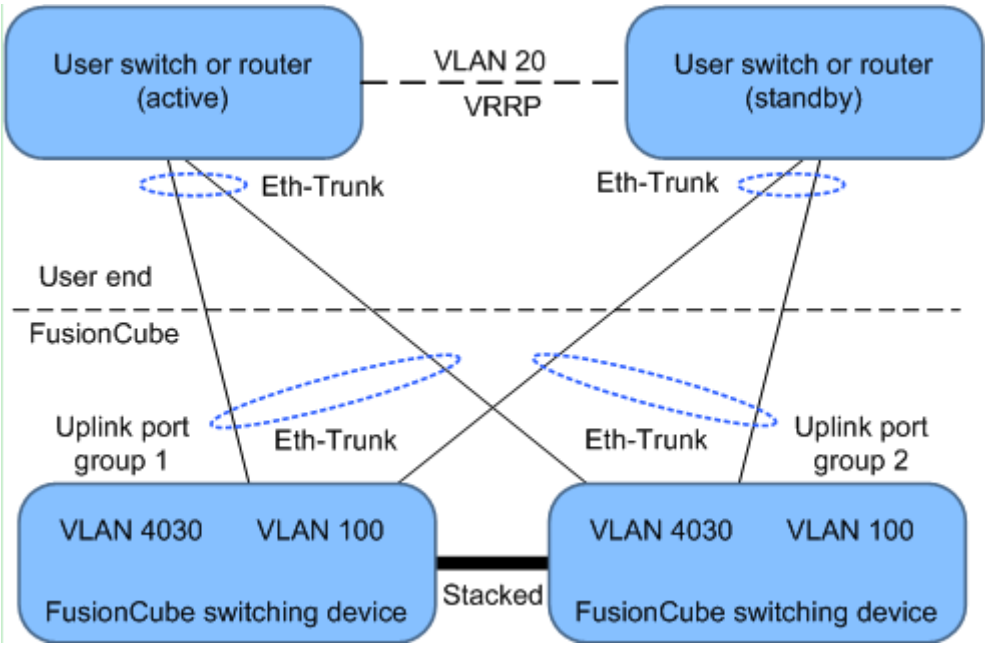
- User switches work in redundancy mode using VRRP. The user switches and FusionCube switching devices support Eth-Trunk (or port aggregation) and SmartLink.

  Figure 3-43 Uplink interconnection using Smart-link when management and service planes are converged



- User switches or routers work in active/standby mode using VRRP. An independent VLAN is used to control VRRP heartbeats on the switches. User switches and the FusionCube switching devices support Eth-Trunk (or port aggregation).

Figure 3-44 Uplink interconnection using an independent VLAN for VRRP heartbeat control when management and service planes are converged



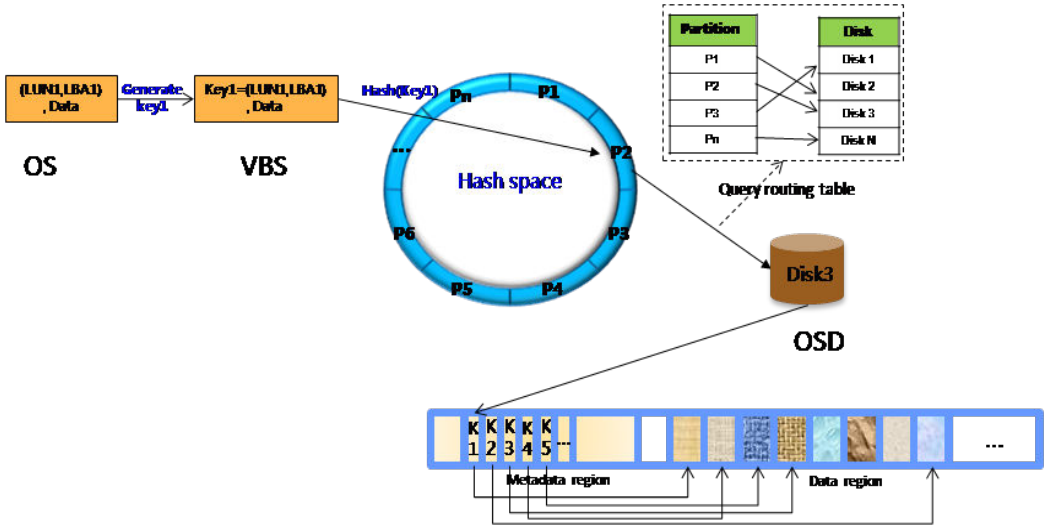
4 High Performance

The FusionStorage block storage uses completely distributed large resource pool architecture. It does not have centrally accessed components or modules, which eliminates the performance insufficiency caused by the bottleneck of a single component or module. It uses the distributed hash data routing algorithm to evenly store service data in all disks in the resource pool. In addition, all disks in the resource pool can be used as hot spare disks of the resource pool. This enables rapid restoration when a component or module is faulty and consistent service performance.

4.1 Distributed I/O Ring

FusionStorage Block uses the distributed hash table (DHT) routing technology, which allows the service I/O to rapidly locate the specific place where the data is stored on the disk. This prevents searching and calculation of a great deal of data. The DHT technology uses Huawei homegrown algorithm to ensure balanced data distribution among disks and rapid, automatic data adjustment when the hardware quantity increases (due to capacity expansion) or decreases (due to a hardware fault). The DHT technology also ensures data migration validity, rapid, automatic self-healing, and automatic resource balancing.

Figure 4-1 FusionStorage block storage data routing process



During the system initialization process, FusionStorage Block divides the hash space into N partitions based on the number of hard disks. For example, in a scenario where two data copies are retained and N is 3600 by default. If there are 36 hard disks in the system, each hard disk provides 100 partitions. The mapping between hard disks and partitions is configured during system initialization and is dynamically adjusted with the number of hard disks in the system. The mapping table occupies a small space. The nodes in FusionStorage Block store the mapping in memory for fast routing.

FusionStorage Block logically divides each logical unit number (LUN) into slices of 1 MB. For example, a LUN of 1 GB is divided into 1024 slices of 1 MB. When an application accesses FusionStorage Block, the SCSI command carries the LUN ID, logical block addressing (LBA) ID, and read/write data content. The OS forwards the message to the VBS of the local node. The VBS generates a key based on the LUN ID and LBA ID. The key contains the roundup result of the LBA ID divided by 1 MB. The partition ID is calculated using the DHT Hash. The specific hard disk is located based on the mapping between the partitions and hard disks stored in the memory. The VBS forwards the I/O request to the OSD to which the hard disk belongs.

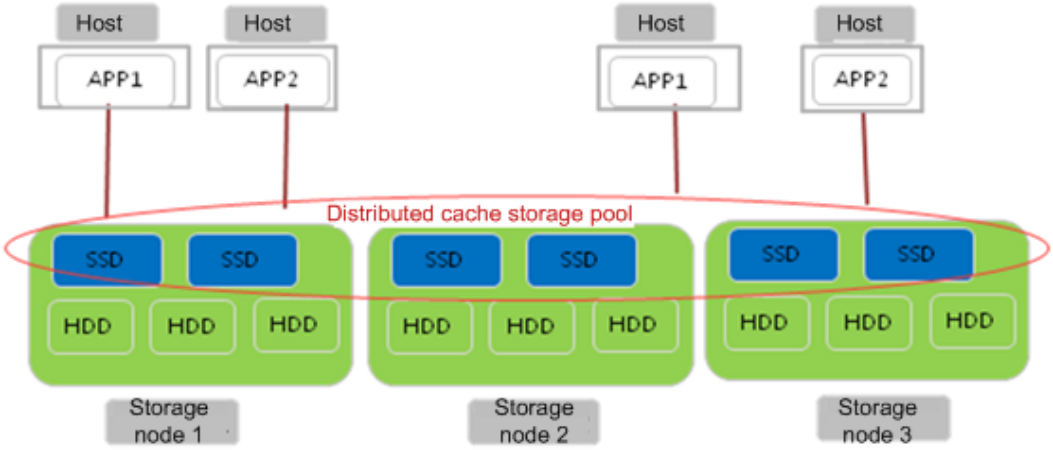
Each OSD manages a hard disk. During system initialization, the OSD divides the hard disk into slices of 1 MB and records the slice allocation information in the metadata management area of the hard disk. After receiving the I/O request from the VBS, the OSD searches the hard disk for data slice information based on the key, obtains the data, and returns the data to the VBS. The data routing process is complete.

4.2 Distributed SSD Cache Acceleration

Limited by the mechanical nature, traditional hard disk drives (HDDs) have never seen performance improvement in dozens of years although the capacity increases greatly. The random I/O delay is from several milliseconds to tens of milliseconds, severely affecting user experience and performance. Compared with HDDs, the solid state drives (SSDs) provide higher performance but higher cost per bit. Nowadays, SSDs are used as the system cache or in the Tier layer to balance the performance and cost.

In FusionStorage Block, the SSDs on each storage node constitute a distributed cache resource pool shared by all services. In this way, the SSD resources are fully utilized.

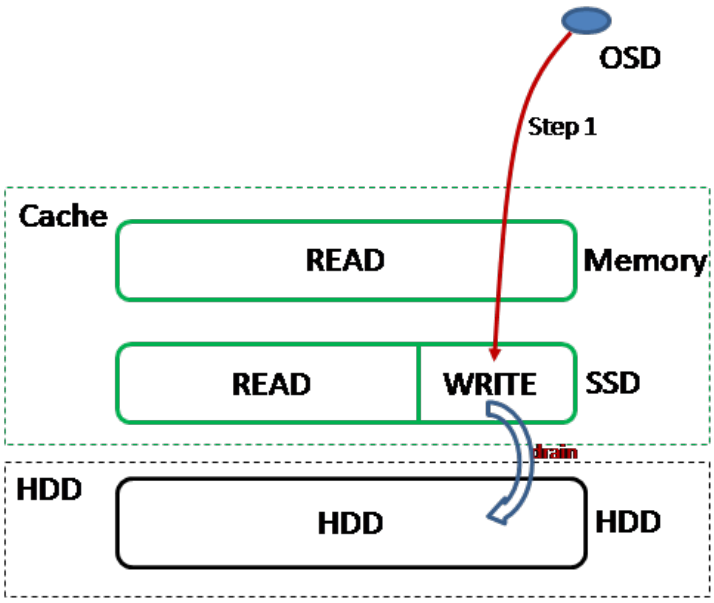
Figure 4-2 Logical architecture of FusionStorage Block distributed cache



4.2.1 Write Cache

When receiving write I/O requests from the VBS, the OSD caches the write I/O requests in the SSD cache. The OSD sorts and arranges the I/O requests in the background and then write data to hard disks.

Figure 4-3 Write cache mechanism



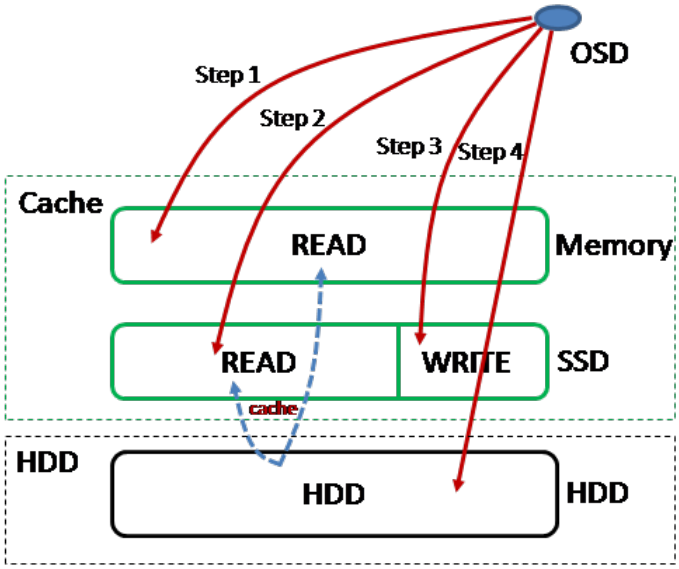
4.2.2 Read Cache

FusionStorage Block adopts a hierarchical mechanism for read cache. The first layer is the memory cache, which caches data using the LRU mechanism. The second layer is the SSD cache, which functions based on the hotspot read mechanism. The system collects statistics on each piece of read data and the hotspot access factor. When the threshold is reached, the system automatically caches data to the SSD and removes the data, which has not been accessed for a long time, from the SSD.

When receiving a read I/O request from the VBS, the OSD performs the following operations:

1. Search the read cache of the memory for the required I/O data.
   - If the I/O data is found, return it to the VBS and move the I/O data to the LRU queue head of the read cache.
   - If the I/O data is not found, perform 2.
2. Search the read cache of the SSD for the required I/O data.
   - If the I/O data is found, return it directly in a copy scenario. In an EC scenario, obtain data blocks on other nodes, combine data by using the EC algorithm, return data, and add the hotspot access factor of the I/O data.
   - If the I/O data is not found, perform 3.
3. Search the write cache of the SSD for the required I/O data.
   - If the I/O data is found, return it directly in a copy scenario. In an EC scenario, obtain data blocks on other nodes, combine data by using the EC algorithm, return data, and add the hotspot access factor of the I/O data. If the hotspot access factor reaches the threshold, the I/O data is recorded to the read cache of the SSD.
   - If the I/O data is not found, perform 4.
4. Search the hard disk for the required I/O data. Return the data directly in a copy scenario. In an EC scenario, obtain data blocks on other nodes, combine data by using the EC algorithm, return data, and add the hotspot access factor of the I/O data. If the hotspot access factor reaches the threshold, the I/O data is recorded to the read cache of the SSD.

Figure 4-4 FusionStorage Block read cache mechanism



4.2.3 Pass-Through of Large Blocks

The following table provides performance comparison of different media. For the random small I/O, SSDs provide a performance advantage of tens to hundreds of times than HDDs. However, the advantages in sequential I/O are not obvious.

Table 4-1 Performance comparison

Medium Type	4 KB Random Write IOPS	4 KB Random Read IOPS	1 MB Write Bandwidth	1 MB Read Bandwidth	Average Delay (ms)
SAS	180	200	150 MB	150 MB	3 to 5
NL-SAS	100	100	100 MB	100 MB	7 to 8
SATA	100	100	80 MB	80 MB	8 to 10
SSD disk	70,000	40,000	500 MB	500 MB	< 1
SSD card	600,000	800,000	2 GB	3 GB	< 1

The HDD disk performance data is measured in the condition that the HDD write cache is disabled. For the HDDs used to set up a storage system, the write cache must be disabled to ensure reliability.

The working principle of the HDD disks is similar. So is the performance of the HDDs from different vendors. The difference in performance is less than 10%.

There is a big difference in the performance of SSD disks and SSD cards. This table uses one type of SSD disks and SSD cards as an example. The SSD disk bandwidth performance is limited by the SAS/SATA interface bandwidth. The 6 Gbit/s SATA interface is commonly used for tests.

As indicated by the preceding performance data, SSDs have obvious performance advantages over HDDs in small-block random IOPS. In large-block sequential I/O, SSD cards provide large bandwidth advantages, but SSD disks do not have obvious advantages over HDDs due to the bandwidth limit on the SAS/SATA interface. If an SSD disk serves as the cache for more than five HDDs, directly accessing HDDs provides higher performance than accessing the SSD.

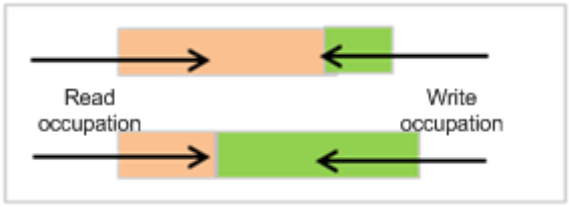
Huawei FusionStorage supports bypass SSD cache in large-block I/O. This feature provides the following advantages:

Higher performance in large-block I/O operations. The cache resources originally occupied by large-block I/O operations are released, and more small-block I/O operations can be cached. This increases the cache hit rate of small-block random I/O operations, enhances the overall system performance, allows more write I/O operations, and extends the service life of SSD cards.

4.2.4 Dynamic Cache Adjustment

Many distributed storage devices use SSDs as the cache. Some vendors provide the read cache only, and some vendors provide read/write cache. However, the read/write caches of most vendors are configured with a 70:30 ratio by default. In the read and write balanced scenario, this configuration makes the entire system run properly. For the read-intensive scenarios (for example OLAP) or write-intensive scenarios, however, this configuration will cause waste of SSD resources. Huawei FusionStorage Block supports automatic adjustment of the read/write cache ratio based on the system running status. This feature helps maximize the SSD cache resources.

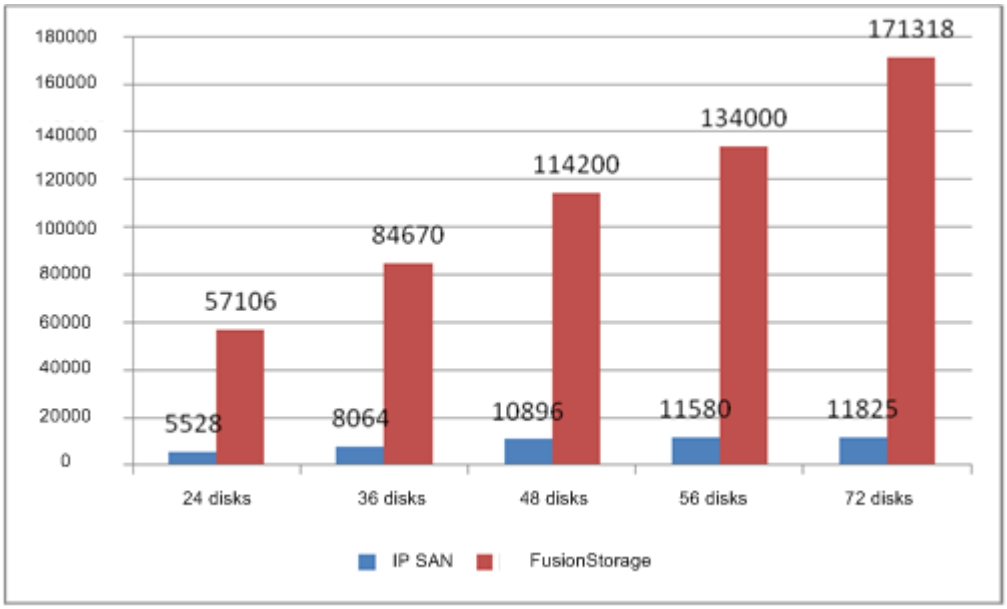
Figure 4-5 Dynamic cache adjustment



4.3 Performance Advantages of FusionStorage Block over SAN

4.3.1 Higher Performance

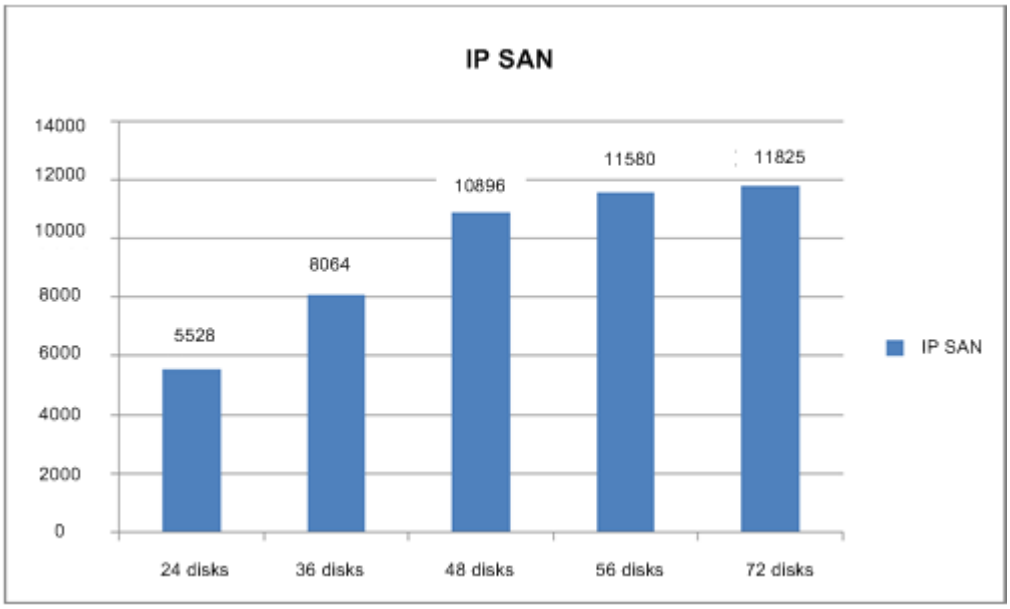
Figure 4-6 FusionStorage and IP SAN comparison test



Under the same test conditions, FusionStorage provides more than 10 times higher performance than IP SAN. The performance increases as the number of disks.

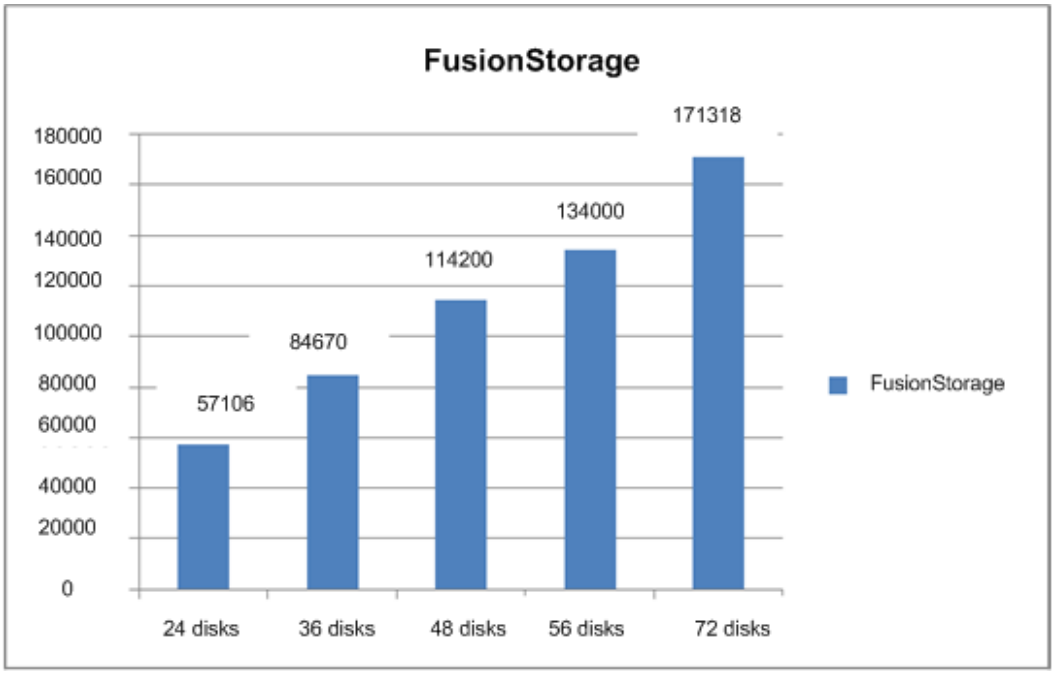
4.3.2 Linear Scale-Up and Scale-Out

Figure 4-7 IP SAN



IP SAN supports scale-up only and does not ensure linear improvement even in scale-up. As shown in the preceding figure, IP SAN can maintain linear improvement in 48 disks. However, as the number of disks increases, the processing capability cannot be expanded linearly due to the limit in the processing capability of the heads. Therefore, the high-performance configuration for IP SAN is generally not configured with the maximum number of disk enclosures.

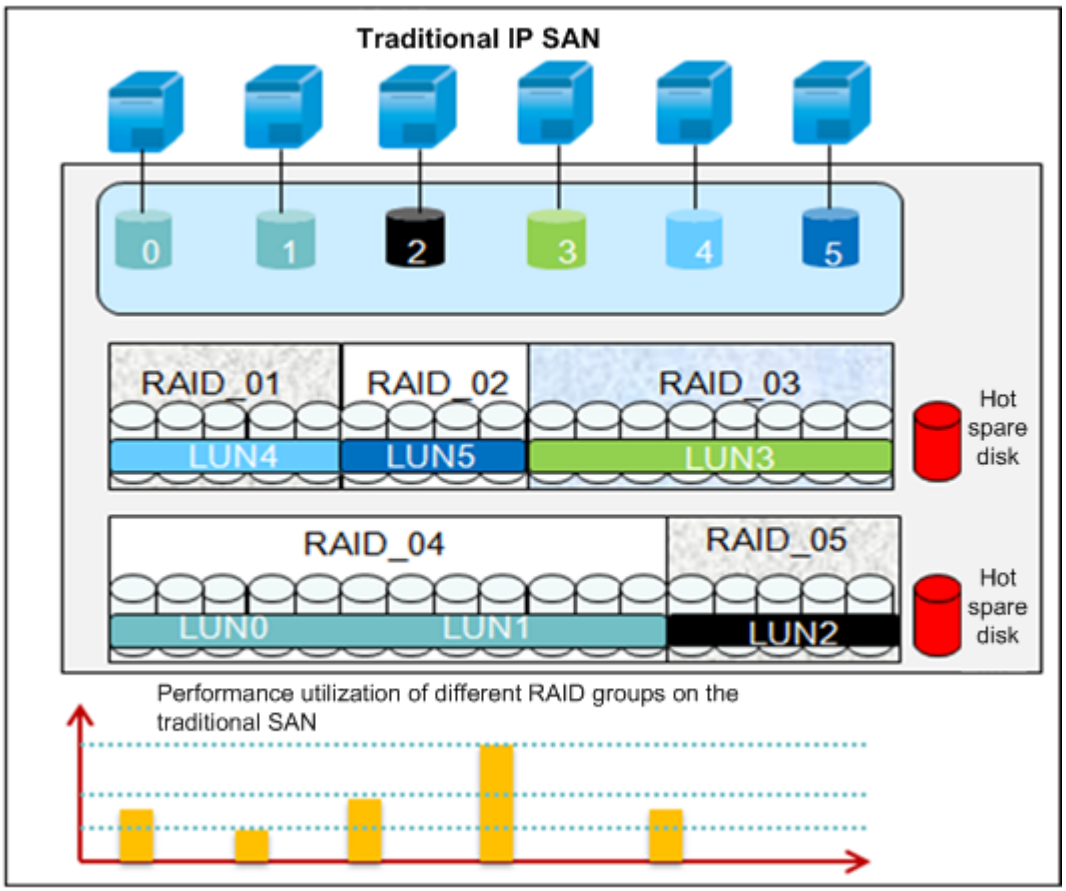
Figure 4-8 FusionStorage



FusionStorage Block provides linear performance improvement as the number of hard disks increases.

4.3.3 Large Pool

Figure 4-9 Traditional IP SAN RAID



Before providing services, the traditional IP SAN needs to be configured with RAID arrays, which are divided into LUNs. For reliability purposes, a RAID array is generally composed of a few disks. Therefore, the performance of a single RAID array is limited. The traditional IP SAN has the following drawbacks:

Complex service planning and difficult adjustment: When the RAID array cannot provide required performance, you need to migrate LUNs to another RAID array. You have to adjust multiple RAID arrays and LUNs especially when no RAID array can meet the LUN performance requirements while the overall system performance still has allowance. The adjustment may pose risks to normal system running. In many cases, even large-scale adjustment cannot meet performance requirements and causes waste of resources. The system performance is wasted, and the performance of different RAID arrays cannot be shared. As shown in the preceding figure, the RAID arrays have different performance requirements. As a result, the system has redundant resources, but some services still have no resources to use. The following problems may exist:

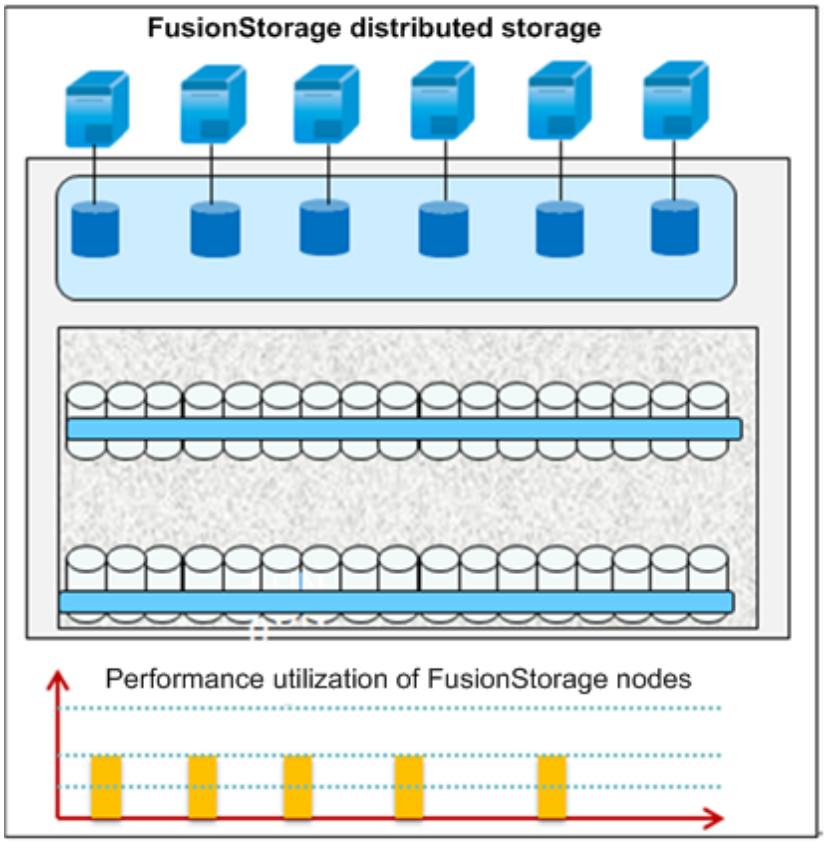
- Services have different performance requirements.

  RAID arrays are planned based on service requirements. The resources of different RAID arrays cannot be shared, resulting in waste of performance resources.

- A service has different performance requirements in different time periods.

  Each RAID array is planned based on its maximum performance. The maximum performance, however, may be required only in one day, week, or month. As a result, the resources are underused and wasted in most of the time.

Figure 4-10 FusionStorage Block large resource pool architecture



FusionStorage Block uses large resource pools. All hardware resources are used for any service. If the overall system performance and capacity meet requirements, users can add services (new services or enhanced services) without extra performance planning or adjustment. Huawei FusionStorage can easily cope with IP SAN problems. It can maximize the system performance, minimize the maintenance investment, and reduce the service interruption risks.

4.3.4 SSD Cache vs SSD Tier

The traditional SAN storage uses the memory in the engine as the storage I/O cache. However, restricted by the memory size, the typical cache capacity is only 8 GB, 16 GB, or 32 GB, which is insufficient to meet service requirements, especially for scenarios with complex hybrid services and high performance requirements. The memory in the engine can cache little data and cannot play a significant role. Therefore, an increasing number of traditional storage vendors use SSDs to accelerate the storage service processing. However, most storage vendors use SSDs as SSD tier rather than SSD cache. SSD tier is helpful for stable and simple services with fixed and long-duration hotspots, but cannot meet the requirements of scenarios with multiple services and rapidly changing hotspots.

Huawei FusionStorage Block uses SSDs as cache. It can promptly identify service hotspot changes, quickly respond to changes, and continuously ensure high performance.

Table 4-2 SSD cache vs SSD tier

Item	SSD Cache	SSD Tier
Benefits	Stores hotspot data on high-speed storage media (SSD cards or disks) to improve processing performance of the storage system.	Stores hotspot data on high-speed storage media (SSD cards or disks) to improve processing performance of the storage system.
Data change	Migrates hotspot data from HDDs to SSDs, and does not delete data from HDDs. When the hotspot data is not frequently accessed, the SSD space will be released.	Deletes data from HDDs after migrating the hotspot data to SSDs. When the hotspot data is not frequently accessed, data will be written to the HDDs and the SSD space will be released.
Capacity	SSDs are only used as system cache, and the total system capacity is not increased.	SSDs are used as storage tier, and the total system capacity is increased.
SSD space utilization	The hotspot data is backed up in HDDs, and it is unnecessary to use RAID technology in SSDs to ensure reliability. Therefore, the SSD space utilization is high.	When the hotspot data is migrated to SSDs, the hotspot data is deleted from HDDs. RAID technology must be used to ensure reliability.
SSD performance utilization	The SSD performance utilization is high. The data is written directly into SSDs, without write penalty. The original hotspot data that is not frequently accessed will be directly overwritten by new hotspot data.	RAID technology is used, and a write penalty on the RAID array occurs when data is migrated to the SSDs. The original hotspot data that is not frequently accessed will be overwritten by new hotspot data only after the data is written back to the HDDs.
Statistical period of hotspot	It takes only several minutes to capture changes of hotspots.	It takes several hours to capture changes of hotspots.
Data block size	Generally, the data blocks are 8 KB, 16 KB, or 32 KB. The cache resources are fully used.	The data blocks are 1 MB, 2 MB, or 4 MB. The cache resources are not fully used.

Item	SSD Cache	SSD Tier
Application scenarios	Suitable for scenarios with frequent hotspot changes. Suitable for hybrid service scenarios, especially the scenarios involving great changes in service hotspots.	Suitable for scenarios where the hotspots seldom change. Suitable for single service scenarios where the hotspots seldom change.

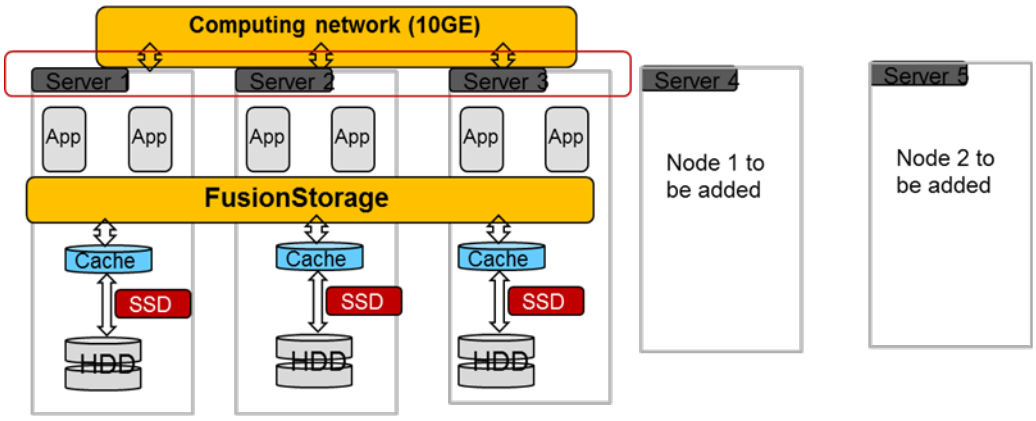
5 Linear Scaling

5.1 Storage Smooth Expansion

The distributed architecture of FusionStorage Block allows easy capacity expansion and supports ultra-large capacity storage.

- After capacity expansion of storage nodes, the system can rapidly achieve load balancing without migration of a large amount of data.
- FusionStorage supports flexible capacity expansion and allows concurrent and separate expansion of compute nodes, hard disks, and storage nodes. Storage capacity can be expanded while the compute nodes are added. After capacity expansion, compute and storage resources are still converged.
- The software engine, storage bandwidth, and cache are evenly distributed to all nodes. The system IOPS, throughput, and cache are increased linearly as the number of nodes are added.

Figure 5-1 Storage smooth expansion



5.2 Performance Linear Expansion

FusionStorage uses an innovative architecture to organize the dispersedly distributed SATA hard disks into an efficient SAN-like storage pool. It provides higher IO throughput than SAN storage devices and extreme storage performance.

Distributed Engine

FusionStorage uses distributed stateless engines, each of which are deployed on a node. This eliminates the performance bottleneck of central engines. The software engines on each node consume little CPU resources and provide more IOPS and throughput than central engines. For example, the system has 20 nodes to access storage resources provided by FusionStorage, and each node provides 2 x 10 Gbit/s bandwidth to the storage plane. If each node is deployed with a VBS module (a storage engine), there are 20 storage engines in the system and the total throughput can reach 400 Gbit/s (20 x 2 x 10 Gbit/s = 400 Gbit/s). The storage engines can be added linearly as the cluster capacity is expanded. This breaks the performance bottleneck of the centralized head of the traditional dual-controller or multi-controller storage systems.

Distributed Cache

FusionStorage evenly distributes cache and bandwidth among nodes.

Different from the independent storage system where a large number of hard disks share the limited bandwidth between compute devices and storage devices, the hard disks on each node in a FusionStorage storage cluster use independent input/output bandwidth.

FusionStorage allows the use of part of node memory as the read cache and SSDs as the write cache. Data caches are evenly distributed to all nodes. The total cache capacity of all nodes is far larger than the cache capacity provided by external storage devices. Even if large-capacity and low-cost SATA hard disks are used, FusionStorage can still provide high I/O performance and 1 to 3 times higher overall performance.

FusionStorage can use SSDs for caching data. In addition to providing high capacity and the write cache function, the SSDs can collect statistics on and cache hotspot data, further improving system performance.

Global Load Balancing

The **DHT** mechanism of FusionStorage allows the I/O operations from the upper-layer applications to be evenly distributed and performed on different hard disks of different nodes to achieve global load balancing. The global load balancing is implemented as follows:

- The system automatically disperses data blocks on each volume and stores them on different hard disks. As a result, the data frequently or seldom accessed is evenly distributed on different nodes to prevent hot spots.
- The data slicing and distribution algorithm enables primary and secondary data copies to be evenly distributed on different hard disks of different nodes. In this way, the number of primary copies distributed on each hard disk is equal to the number of secondary copies.
- When a node is added or deleted due to a failure, FusionStorage employs the data rebuild algorithm to balance load among all nodes after system rebuild.

Distributed SSD Storage

Huawei FusionStorage supports distributed SSDs, which provide higher read and write performance than traditional SATA or SAS HDDs.

FusionStorage virtualizes the PCIe SSD cards configured on storage nodes into a virtual storage resource pool to provide high-performance read and write for applications.

FusionStorage supports Huawei SSD cards and mainstream PCIe SSD cards.

High-Speed InfiniBand Network

FusionStorage provides IB network for applications demanding large bandwidth and low delay. FusionStorage provides the following functions:

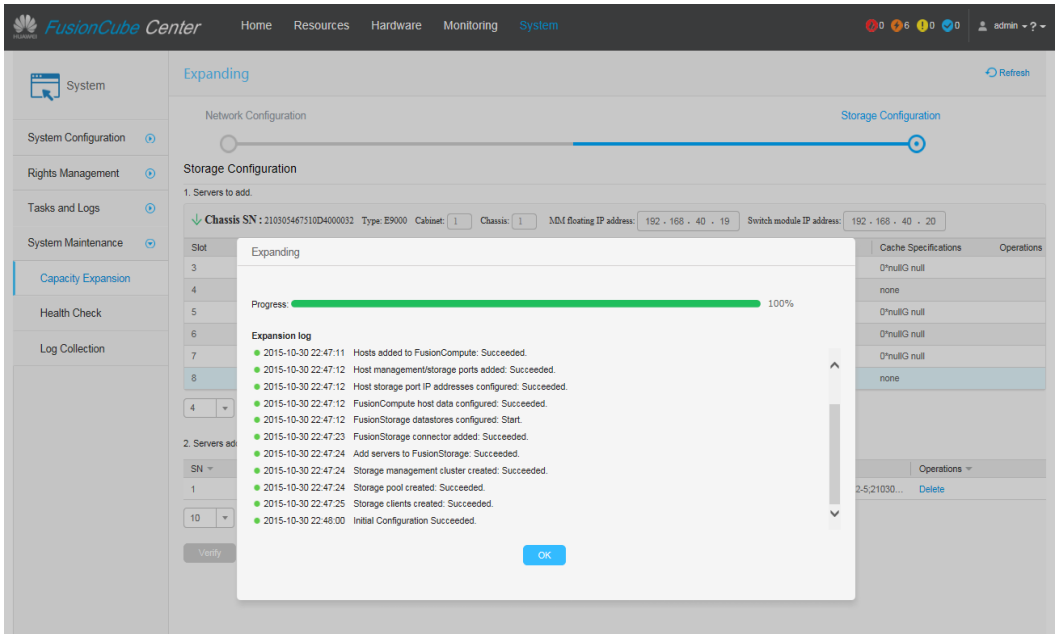
- Supports 56 Gbit/s FDR and 100 Gbit/s EDR IB and RDMA, which provides ultra-high-speed communication between nodes.
- Uses standard multilayer fat-tree networking for smooth capacity expansion.
- Provides a communication network that is almost free from congestion and switching bottlenecks.
- Transmits compute and storage information with a delay in nanoseconds.
- Provides lossless network QoS to ensure data integrity during transmission.
- Allows multi-plane communication for active and standby ports to improve transmission reliability.

5.3 One-Click Capacity Expansion

FusionCube provides the one-click capacity expansion function to simplify system capacity expansion operations.

1. On the FusionCube Center WebUI, choose **System** > **System Maintenance** > **Capacity Expansion**.
2. The system automatically discovers devices and displays the discovered devices.
3. Configure network and storage parameters and click **Verify**. If the verification is successful, click the **Capacity Expansion** button.
4. The system automatically completes the expansion configuration, including configuring network settings for nodes, adding nodes to storage clusters, and expanding the storage pool or creating a storage pool, based on the data configured.

Figure 5-2 FusionCube Center capacity expansion



6 System Security

6.1 System Security Threats

Security Threats from External Networks

- Traditional IP attacks

  Traditional IP attacks include port scans, IP address spoofing, land attacks, IP option attacks, IP routing attacks, IP fragmentation attacks, IP fragment packet attacks, and teardrop attacks.

- OS and software vulnerabilities

  Numerous security bugs have been found in compute software, including third-part, commercial and free software. Hackers can control the OS by making use of minor programming errors or context dependency. Common OS and software vulnerabilities include buffer overflows, operations abusing privilege, and code download without integrity verification.

- Viruses, Trojan horses, and worms.
- SQL injection attacks

  Attackers include portions of SQL statements in a web form entry field or query character strings of a page request in an attempt to get the node to execute malicious SQL statements. The forms, in which the contents entered by users are directly used to construct (or affect) dynamic SQL commands or the contents are used as input parameters for stored procedures, are particularly vulnerable to SQL injection attacks.

- Phishing attacks

  Phishing is the act of attempting to acquire information, such as user names, passwords, and credit card details, by masquerading as a trustworthy entity in an electronic communication. Communications purporting to be from popular social web sites, auction sites, online payment processors or IT administrators are commonly used to lure the public. Phishing is typically carried out via e-mail or instant messaging.

- Zero-day attacks

  Zero-day vulnerabilities are security vulnerabilities that have not been patched. Zero-day attacks are attacks that exploit zero-day vulnerabilities. It is difficult to install patches

immediately after a security vulnerability is found because it takes time to confirm, verify, evaluate, and fix the vulnerability. Therefore, zero-day vulnerabilities pose a great threat to network security.

Security Threats from the Internal Network

- Ever-changing attacks pose security risks.

  Internal network ARP spoofing and abuse of malicious plug-ins pose new security threats. The attacked intranet host may be used as "zombies" for penetration attacks or used as a DDOS tool to send a large number of attack packets to occupy network bandwidth. The system is vulnerable to attacks if malicious plug-ins are used or web pages that are implanted with viruses or Trojan horses are viewed.
- Worms and viruses are spread through loopholes if patches are not upgraded or the antivirus database is not updated in a timely manner.

  If the OS, database, and application software of the hosts and devices on the network have security vulnerabilities and are not patched and the antivirus database of the hosts is not updated in a timely manner, viruses and worms will be spread. A large-scale worm outbreak may paralyze the intranet and interrupt services.
- Confidential information disclosure happens frequently because of unauthorized Internet access activities.

  Enterprise employees can bypass the firewall and directly connect to the external network through the telephone, VPN, or GPRS. This may cause disclosure of important confidential information.
- Uncontrolled mobile device access challenges network border security.

  The notebooks, pocket PCs, and other mobile devices of employees or temporary visitors are used in various network environments and may carry viruses and Trojan. If these devices access the intranet without being scanned, the intranet security will be threatened.
- Uncontrolled use of hardware and software threatens asset security.

  If internal assets (such as CPUs, DIMMs, and hard disks) are replaced and modified without effective tracing measures and unified management, it is difficult to locate the fault once an attack or security incident occurs.
- Application software without monitoring mechanism poses new security risks.

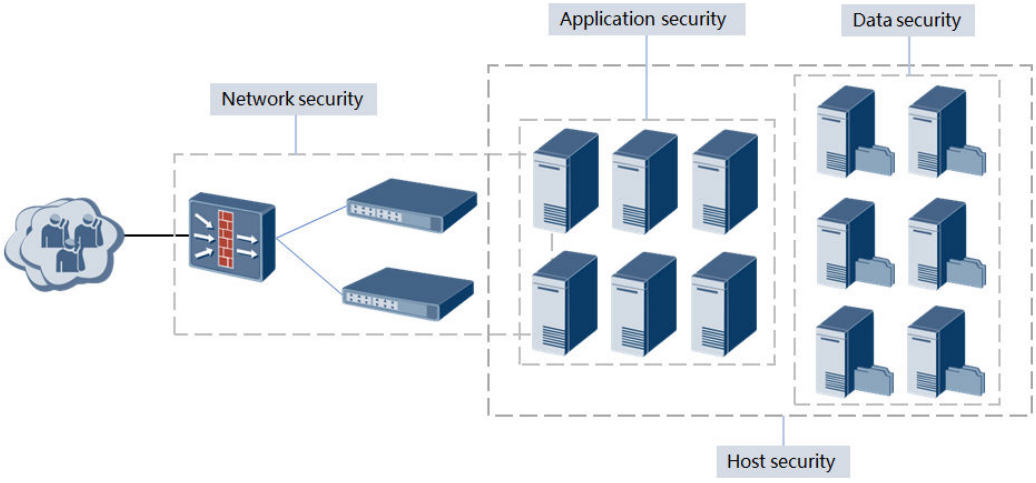
  Popular social communication applications, such as QQ and MSN, may spread viruses, worms, and Trojan horses. Using network tools, such as BitTorrent and eMule, to download movies, games, and software will affect the bandwidth for mission-critical services.
- Ineffective management of peripherals causes data leakage and virus spreading.

  Peripherals, such as USB flash drives, CD drives, printer, infrared, serial port, and parallel ports, are prone to data leakage and virus infection. The peripherals, especially the USB flash drives, cannot be effectively managed by sealing the ports or introducing regulations. Technical measures are required to manage and control the peripherals.
- Security regulations without support of technical measures cannot be put into practice.

6.2 Overall Security Framework

FusionCube provides a security solution to address security risks and issues. **Figure 6-1** shows the security framework. The FusionCube security framework ensures system security from the network, host, application, and data aspects.

Figure 6-1 FusionCube security solution framework



The FusionCube security is ensured from the following aspects:

- Network security

  Network isolation is used to ensure normal data processing, storage security, and maintenance.
- Application security

  Login user authentication, rights control, and audit control are adopted to ensure application security.
- Host security

  OS security hardening has been performed to ensure normal operation of hosts.
- Data security

  Cluster disaster recovery (DR), cluster backup, data integrity, and data confidentiality are used to ensure data security.

6.2.1 Network Security

The FusionCube network communication plane is divided into the following planes:

- Service plane

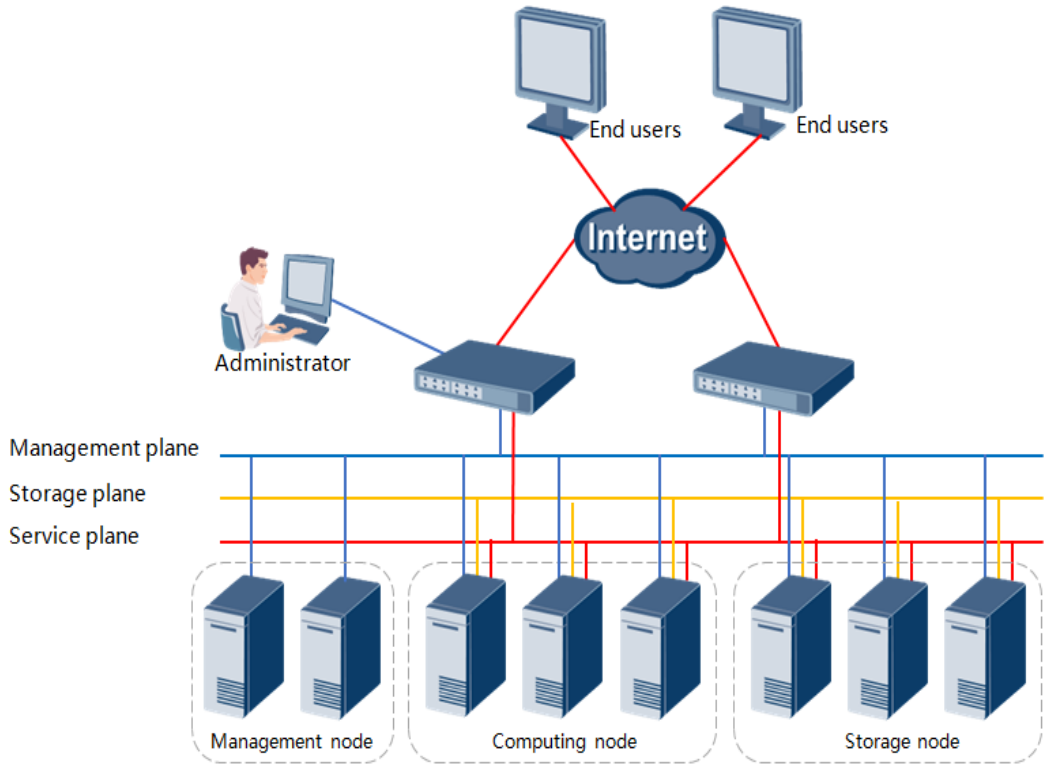
  The service plane provides service channels and the communication plane for the virtual NICs of VMs.
- Storage plane

  The storage plane enables VMs to access storage resources. The storage plane communicates with VMs through the virtualization platform.
- Management plane

  The management plane provides communication channels for system management, routine maintenance, service configuration, and system loading.

For security purposes, isolate the three planes from each other. **Figure 6-2** illustrates the isolation of the planes.

Figure 6-2 Plane isolation



6.2.2 Application Security

6.2.2.1 Rights Management

FusionCube supports rights-based management. Users are assigned different rights to ensure system security.

The operation rights of a user are defined by the role of the user. A user can have multiple roles, and a role can have various operation rights. The binding between a user and a role determines the operations that the user can perform. If a user has multiple roles, the user can perform all the operations defined for these roles.

6.2.2.2 Web Security

FusionCube implements the following web service security functions:

- Automatically converts user requests into HTTPS links.

  The web service platform automatically redirects user requests to HTTPS links. When a user accesses the web service platform using HTTP, the web service platform automatically converts the user requests into HTTPS requests to enhance access security.
- Prevents cross-site scripting.

  Cross-site scripting is a type of injection, in which attackers use insecure websites to attack website visitors.
- Prevents SQL injections.

  Attackers inject SQL commands to entry fields of web sheets or query character strings of page requests to enable servers to execute malicious SQL commands.

- Prevents cross-site request forgeries.

  Malicious requests can exploit the browser's function of automatically sending authentication certificates. A cross-site request forgery attack deceives a logged-in user into loading a page with a malicious request in order to inherit the identity and privileges of the user. In this way, the attacker can make mischief for their own purposes, for example, changing the user's password or address information.

- Hides sensitive information for security purposes.

  Sensitive information is hidden to prevent attackers' access.

- Restricts file upload and download.

  Measures are taken to prevent confidential files from being downloaded and insecure files from being uploaded.

- Prevents unauthorized uniform resource locator (URL) access.

  Users are prevented from accessing unauthorized URLs.

- Supports graphic verification codes for logins.

  When a user attempts to log in to the web system, a random verification code will be generated. The user can log in to the system only when the user name, password, and random verification code are correct.

6.2.2.3 Database Hardening

The FusionCube management nodes use GaussDB database.

Basic security configurations are required to ensure database security. The security configurations for a GaussDB database include the following:

- Access source control

  According to service requirements and security standards, the database allows local access only. All cross-server access requests are rejected to prevent external attacks.

- Principle of least privilege

  Except the database super administrator, all the users are assigned roles based on the principle of least privilege.

- Folder protection

  The owner of the data installation folder and its data area folders is the user who performs the installation, and the permission on the folders and its subfolders includes read, write, and execute.

- Protection of sensitive files

  The owner of the database core configuration files is the user who performs the installation, and the permission on the files includes read and write.

- Restriction on the number of concurrent connections

  By default, the system supports a maximum of 300 connections. The maximum number of connections can be modified in the configuration file to prevent malicious attacks.

To ensure data security, the data in the database must be backed up periodically. The database supports local online back and remote backup.

- Local backup: A script is executed at the specified time to back up data.
- Remote backup: Data is backed up to a third-party server.

6.2.2.4 Log Management

The following measures are used to ensure log security:

- Logs cannot be modified or deleted on the management systems.
- Only the users authorized to query log information can export logs.

6.2.3 Host Security

6.2.3.1 OS Security Hardening

All the compute nodes, storage nodes, and management nodes of FusionCube use the Linux OS. The following configuration must be performed to ensure OS security:

- Stop unnecessary services, such as Telnet service and file transfer protocol (FTP) service.
- Perform security hardening of the secrete shell (SSH) service.
- Control the access permission on files and directories.
- Allow system access only for authorized users.
- Manage user passwords.
- Record operation logs.
- Detect system exceptions.

6.2.4 Data Security

FusionCube uses a variety of storage security technologies to ensure the security and reliability of user data.

- Data fragment storage

  Data on FusionCube storage nodes is automatically stored in multiple copies. Different copies of each data slice are stored on different storage nodes. As a result, malicious users cannot obtain user data from a single storage node or physical disk.

- Encrypted storage of sensitive data

  The AES-256 or SHA-256 algorithm is used to encrypt sensitive data (such as authentication information) stored in the database.

7 System Reliability

The FusionCube distributed storage system provides cross-node data protection. When multiple hard disks or nodes are faulty, the FusionCube distributed storage system can still provide services. Data is stored on different hard disks of different nodes in the same pool, achieving cross-node reliability and fast fault recovery. The hardware redundancy configuration also ensures high system availability.

7.1 Data Reliability

7.1.1 Block Storage Cluster Reliability

FusionStorage Block employs cluster management to prevent single point of failure. If a node or hard drive is faulty, it will be automatically isolated from the cluster. The entire system services will not be affected. The cluster management is implemented as follows:

- Zookeeper

  ZooKeeper chooses the master MetaData Controller (MDC) and stores metadata generated during system initialization. The metadata includes data routing information, such as the mapping between partitions and hard drives. An odd number of Zookeepers are deployed in a system to form a cluster. A system must be deployed with at least three ZooKeeper nodes and have more than half of the ZooKeeper nodes active and accessible. The number of ZooKeeper nodes cannot be added once the system is deployed.

- MDC

  The MDC controls the status of the distributed clusters. A system must be deployed with at least three MDCs. When a resource pool is added, an MDC will be automatically started or specified for the resource pool. Multiple MDCs elect a master MDC through the ZooKeeper . The master MDC monitors other MDCs. If the master MDC detects the fault of an MDC, it restarts the MDC or specifies an MDC for the resource pool. When the master MDC is faulty, a new master MDC will be elected.

- OSD

  The OSD performs input/output operations. The OSDs work in active/standby mode. The MDC monitors the OSD status on a real-time basis. When the active OSD where a

specified partition resides is faulty, services will be automatically switched over to the standby OSD in real-time to ensure service continuity.

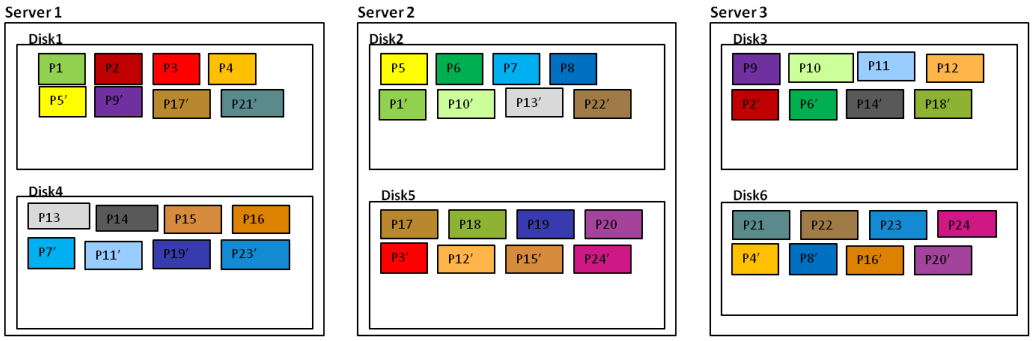
Each node has multiple OSDs to manage drives or SSD virtual drives on the node. The OSDs are in one-to-one mapping with drives or SSD virtual drives, but do not bind with the drives or SSDs. The positions of the storage drives/SSDs in a node can be swapped. This can prevent misoperation during maintenance and improve system reliability.

7.1.2 Multiple Data Copies

To ensure data reliability, FusionStorage Block stores two or three identical data copies for each piece of data. Before storing data, FusionStorage divides the data on each volume into slices of 1 MB and then stores the data slices on nodes in the cluster based on the DHT algorithm.

Figure 7-1 illustrates the multi-data-copy mechanism of FusionStorage Block. As shown in **Figure 7-1**, P1' on disk 2 of server 2 is a copy of data block P1 on disk 1 of server 1. P1 and P1' are two data copies of the same data block. If disk 1 becomes faulty, P1' can be used.

Figure 7-1 FusionStorage Block multiple data copies

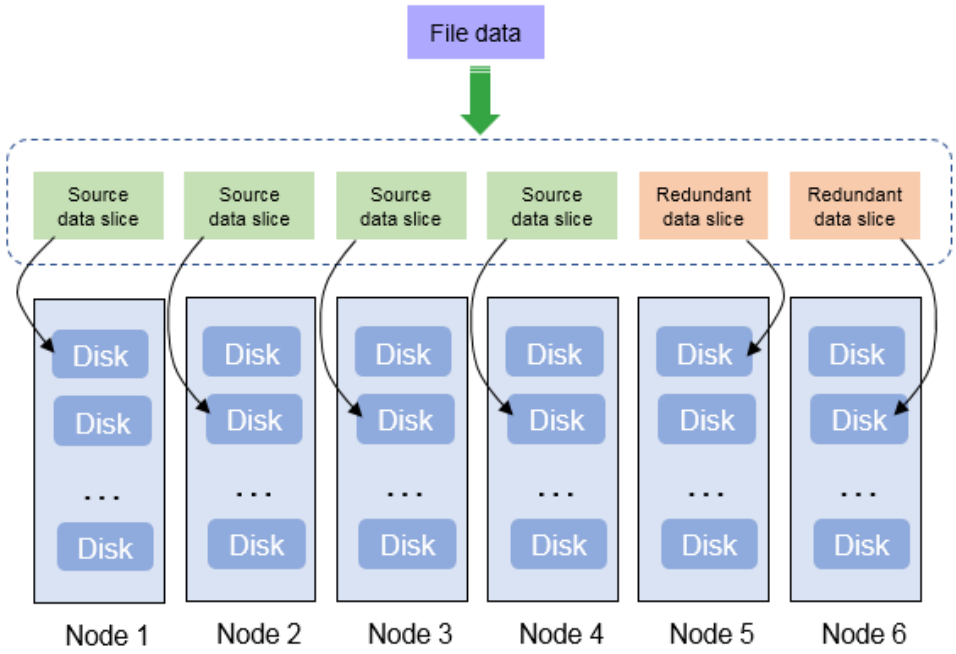


7.1.3 Erasure Code

FusionStorage can also use Erasure Code (EC) to ensure data reliability. Compared with multi-copy backup, EC provides higher disk utilization in addition to high reliability.

The EC-based data protection technology used by FusionStorage is based on distributed and inter-node redundancy. FusionStorage uses the Huawei-developed Low Density Erasure Code (LDEC) algorithm. It is an MDS array code based on the XOR and Galois field multiplication. The minimum granularity is 512 B. It supports Intel instruction acceleration and various mainstream ratios. Data written into the system is divided into N data strips, and then M redundant data strips are generated (both N and M are integers). These data strips are stored on N+M nodes.

Figure 7-2 FusionStorage EC diagram



Data in the same strip is stored on different nodes. Therefore, data in the FusionStorage system not only supports disk-level faults, but also supports node-level faults to ensure data integrity. As long as the number of concurrently failed nodes is smaller than M, the system can continue to provide services properly. Through data reconstruction, the system is able to restore damaged data to protect data reliability.

The EC data protection mode provided by FusionStorage achieves high reliability similar to that provided by traditional RAID based on data replication among multiple nodes. Furthermore, the data protection mode maintains a high disk utilization rate of up to N/(N + M). Different from traditional RAID that requires hot spare disks to be allocated in advance, the system allows any available space to serve as hot spare space, further improving storage utilization.

FusionStorage provides multiple N+M redundancy ratios. Users can set redundancy ratios based on service requirements. FusionStorage supports EC redundancy ratios of 2+2, 3+2, 4+2, 8+2, and 12+3 and data slices of 8 KB, 16 KB, 32 KB, and 64 KB. In this way, users can flexibly configure data redundancy ratios and data slice sizes based on actual service requirements to obtain desired reliability levels.

7.1.4 Data Consistency

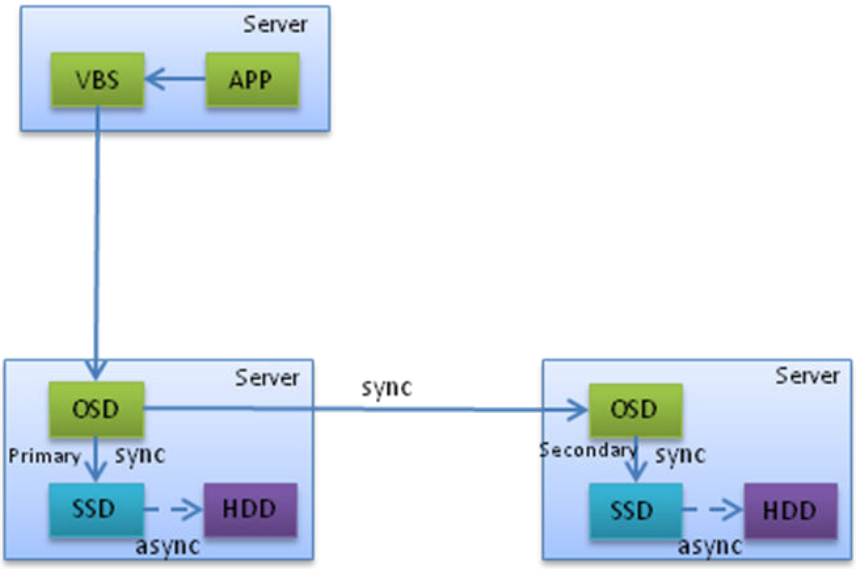
When a piece of data is written into the storage system, the data copies in the storage system must be consistent.

FusionStorage Block uses the following means to ensure data consistence in the system:

- Synchronous write of data copies

When the VBS module sends a write operation request to the active OSD, the OSD writes data to the hard disk of the local node and synchronizes the write operation to the standby OSD. To ensure data consistency, the I/O number for a write operation sent to the active and standby OSDs must be the same. A success message is returned only when the write operation performed on the active and standby OSDs are successful. **Figure 7-3** shows the synchronous write process.

Figure 7-3 Synchronous write of data copies



- Read repair

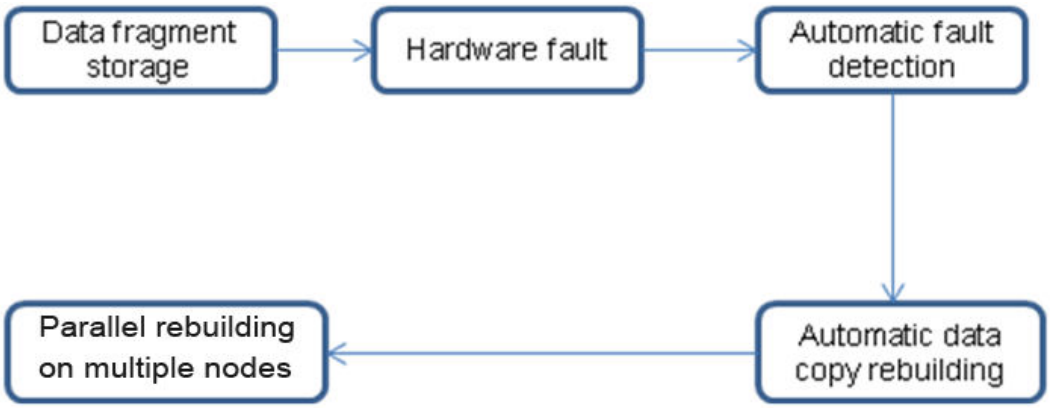
  FusionStorage supports the read repair mechanism. When a data read operation fails, FusionStorage automatically identifies the failure type. If data cannot be read from a disk sector, the system retrieves the data from the data copy stored on another node and writes the data back into the original disk sector. This mechanism ensures correct number of data copies and data consistency between data copies.

7.1.5 Rapid Data Rebuild

Each disk of FusionStorage Block stores multiple data blocks (partitions). The copies of these data blocks are distributed to other nodes in the system based on specified policies. When detecting a hardware fault on a disk or node, FusionStorage Block automatically starts data repair in the background. Because data copies are evenly distributed to different storage nodes, data rebuild starts on different nodes at the same time when a data repair is triggered. Only a small amount of data needs to be rebuilt on each node. This mechanism prevents performance deterioration caused by rebuild of a large amount of data on a single node, and therefore minimizes adverse impacts on upper-layer services.

Figure 7-4 shows the automatic data rebuild process.

Figure 7-4 FusionStorage Block data rebuild process



FusionStorage Block supports parallel and rapid data rebuild as follows:

- Data blocks (partitions) and their copies are scattered in a resource pool. If a hard disk is faulty, its data can be automatically rebuilt in the resource pool rapidly.
- Data is distributed on different nodes. Therefore, data can still be obtained or rebuilt even if a node is faulty.
- Load balancing can be automatically achieved between existing nodes in the event of node failures or capacity expansion. Optimal capacity and performance can be obtained without adjusting application configuration.

7.2 Hardware Reliability

FusionCube uses highly reliable Huawei hardware in redundancy design to ensure system reliability. It has the following features:

- The cache is protected against power failure to ensure data security.
- Hot-swappable SAS disks in RAID 1 array are used as system disks.
- Servers are configured with redundant power supply modules and fans to ensure high system availability.
- Dual-plane design is used for network communication.

8 Compatibility

8.1 Database Compatibility

FusionCube is an open, hyper-converged infrastructure platform independent from upper-layer applications. It is compatible with the following databases:

- Oracle databases: FusionCube supports Oracle RAC One Node and Oracle RAC databases. FusionStorage Block provides shared block storage services. Oracle ASM implements central management of the storage volumes provided by FusionStorage.
- IBM DB2: FusionCube supports DB2 databases in one-node and HA modes. FusionStorage Block provides block storage services.
- SAP HANA: FusionCube supports the SAP HANA Database Cluster edition. FusionStorage Block provides logo volumes, shared volumes, and data volumes for the SAP HANA databases.
- Sybase IQ: FusionCube supports Sybase IQ cluster mode. FusionStorage Block provides block storage services.

