天翼云
eSurfing Cloud

Storage Resource Reutilization System

Product Whitepaper

China Telecom Cloud Technology Co., Ltd.

Revision History

Version	Release Date	Description
4.0	March 24, 2026	1. Support for the free edition. 2. Support for configuring server memory usage parameters for the HBlock service. 3. Support for suspending cloud LUNs. 4. Support for setting extended attributes of local LUNs via API. 5. Support for configuring authentication methods.
3.10	September 25, 2025	1. Support target access permissions to implement permission management for clients and targets. 2. Support backup, generate data backup files independent of source LUNs based on snapshots. 3. Support for QoS policy to regulate traffic in terms of bandwidth and IOPS.
3.9	April 21, 2025	1. Support for snapshot functionality to achieve rapid data backup and recovery. 2. Support for LUN cloning, for scenarios such as data replication and testing/validation.
3.8	February 14, 2025	1. Update API signature method to enhance security. 2. Cloud LUNs support uploading data to S3-compatible object storage. 3. Support for setting the data storage directory of base services. 4. Add reclaim policy for targets, supporting automatic deletion after disassociation from LUNs. 5. Add performance parameter adjustment for scenarios such as intelligent computing, virtualization, and high-sensitivity scenarios. 6. Support for setting redundancy overlap, allowing data copies/fragments to be placed in the same fault domain.
3.7	August 8, 2024	1. Support for configuring cluster topology. 2. Support for creating and managing multiple storage pools. 3. Support for room and rack level fault domains. 4. Support for setting cache pool for LUNs. 5. Support for base service migration.
3.6	June 3, 2024	1. Support for uploading data from backend storage to OOS for storage LUNs and cache LUNs. 2. Hardware and HBlock monitoring data now support integration with Prometheus. 3. Optimize read/write performance. 4. Support for Loongson servers.
3.5	March 4, 2024	1. Support for server and disk path level fault domains, and disk-level data services. 2. Support for specifying installation nodes for base services. 3. Support for capacity quota of the disk path, setting the maximum amount of data that HBlock can write.

		4. Support for targets to be discovered and connected by multiple clients. 5. Support for setting the minimum number of write replicas for LUNs to improve data write security. 6. Expand erasure coding (EC) N+M support range to satisfy N+M <= 128.
3.4	July 12, 2023	1. LUN connections support one primary with multiple backups, improving business availability. 2. Support for IPv6 environments. 3. Provide a one-page Dashboard overview. 4. Support for querying CHAP passwords via command line.
3.3	December 23, 2022	Support for safely removing servers.
3.2	September 26, 2022	1. Add "Basic Concepts" to the "Product Overview" chapter. 2. Expand monitoring items to broaden coverage. 3. Support for alarm and log management. 4. Support for system events in the Events section. 5. Add to the "Appendix" chapter: HBlock Service, System Event List, Monitoring Metrics, and Alarm List.
3.1	June 14, 2022	1. Support for multiple disk paths in standalone mode. 2. Support for specifying servers when creating targets in cluster mode, and support target migration. 3. Support for recording and querying user events.
3.0	January 18, 2022	1. Change command line to non-interactive mode. 2. Support for WEB and API calling methods. 3. LUN operations: Support for setting LUN high availability types and LUN write policies.
2.1	August 27, 2021	1. Support for ARM servers. 2. Software license: Display allowed capacity when viewing licenses. 3. LUN operations: Support for LUN active/standby switchover, i.e., switching between ACTIVE target and STANDBY target for LUNs.
2.0	May 28, 2021	1. Support for cluster deployment. 2. Support for multi-replica and erasure coding (EC) data redundancy.

Contents

1 Product Overview

1.1 Overview

HBlock is a Storage Resource Reutilization System (SRRS) independently developed by eSurfing Cloud. It is a lightweight storage cluster controller and a fully user-mode software-defined storage controller that converts commodity servers and managed storage resources into HA virtual storage disks. It provides distributed block storage services through the standard Internet Small Computer System Interface (iSCSI) protocol and the disks can be mounted to a local server (or other remote servers) for intensive use of resources. Meanwhile, HBlock is compatible with heterogeneous devices and applies to various scenarios. HBlock helps users tackle the challenges of IT architecture upgrade, reduce costs, and increase efficiency, and assists enterprise users in achieving green transformation.

In non-networked mode, HBlock can be seen as a substitute for local disk arrays for local data storage. In networking mode, HBlock can also serve as a bridge between local and cloud storage, automatically synchronizing all data to S3-compatible object storage, retaining only hot data locally to save local storage space, or retaining all data to ensure local I/O performance, achieving hybrid cloud storage.

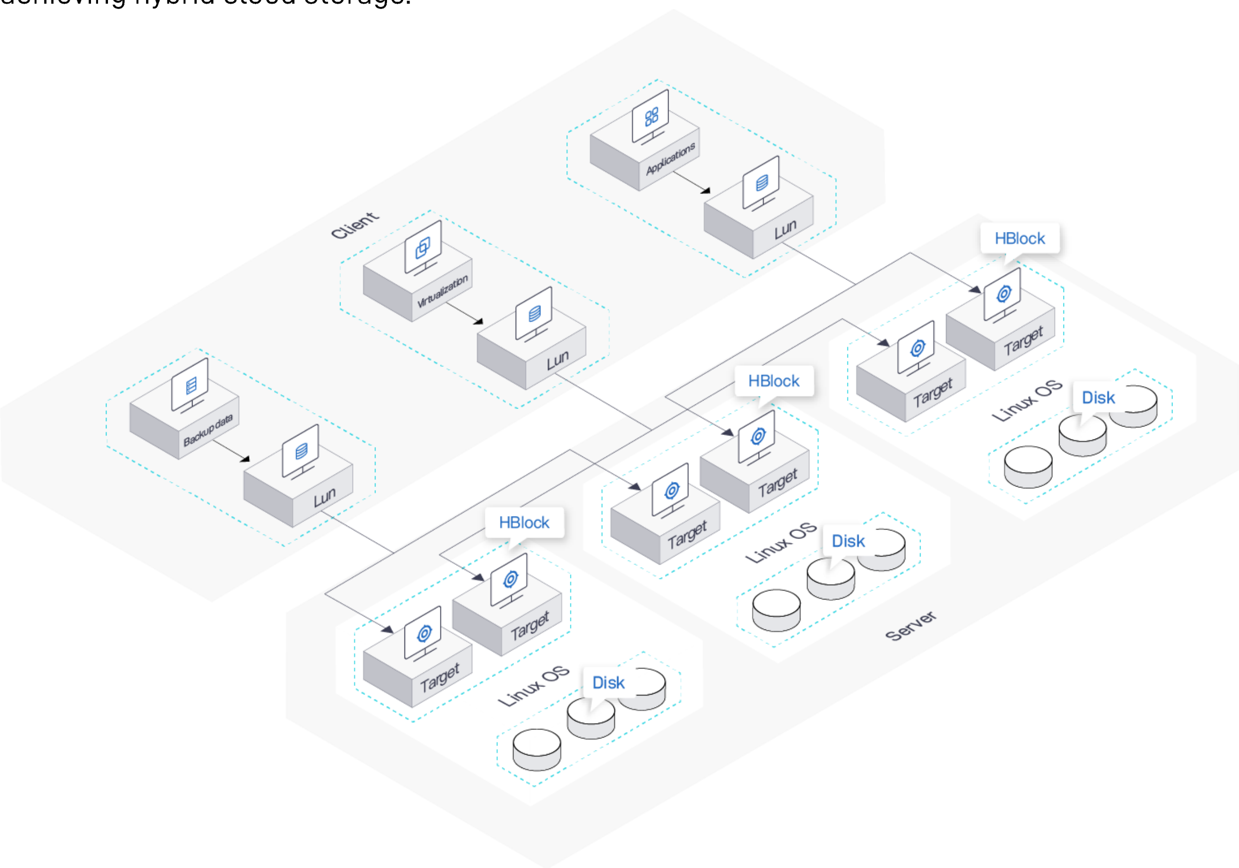

Figure 1 HBlock Architecture Diagram

HBlock can be installed on Linux operating systems as a common application without root permissions. You can deploy HBlock with other applications in a server in a hybrid manner to

form high-performance and HA virtual hard disks for business. This way, HBlock directly re-uses storage resources without affecting user business or purchasing additional equipment!
The traditional hardware storage array provides low latency and high availability for each logical volume, but has poor horizontal scalability, high cost, and may form many "isolated data islands". That leads to high cost and low utilization of storage resource. Traditional distributed storage, while attractive, often suffers from issues such as complex deployment, poor performance, and low stability.
HBlock delivers storage arrays in a quite different approach:

- **Easy Installation**: The HBlock installation package is a zip file that can be installed on mainstream Linux operating systems running on commonly used 64-bit x86 servers, ARM servers, or Loongson servers. It supports physical servers, bare metal servers, and virtual machines. HBlock is completely decoupled with hardware drivers, so users can freely use the latest hardware on the market, with less vendor lock-in.
- **Green**: HBlock runs as a group of user-mode processes, without relying on any specific version of Linux kernel or distributions. It does not rely on or modify the operating system environment, neither does it monopolize the entire hard drive or interfere with the execution of any other processes. Therefore, HBlock can run in the same Linux operating system instance concurrently with other applications. On the one hand, it can help users improve the utilization of existing hardware resource, on the other hand, it also lowers the barriers for potential users to try HBlock - even a virtual machine is not needed.
- **High Utilization**: HBlock supports heterogeneous hardware. HBlock allows each Linux operating system instance in the cluster to have different hardware configurations, such as different numbers of CPUs, different sizes of memory, and different capacities of local hard drives. Therefore, it can improve the utilization of existing hardware resources.
- **High Performance**: HBlock adopts a distributed multi-controller architecture to provide the low latency and high availability just like traditional hardware storage arrays, as well as high scalability and high throughput just like traditional distributed storage. It can scale up from three servers to thousands of servers or shrink from thousands of servers to three servers one by one without service unavailability.
- **High Quality**: HBlock is designed to guarantee the data durability as long as the number of simultaneous disk faults in the cluster is not greater than the allowed faulty number of logical volumes in redundancy mode (for three-copy mode, the allowed number is two; for Erasure Code N+M mode, the allowed number is M). HBlock can ensure the service availability when any single server, link, or disk in the cluster experiences faults. HBlock is specifically designed for Chaos environment and can be applied to various complicated environments such as unstable networks, unstable computing power, and unstable disks. It has been sufficiently tested in complex and large-scale environments before release. HBlock supports snapshot and clone functions. Users can create time-point snapshots for logical volumes to quickly back up and restore data. The clone function enables quick generation of new volume copies based on snapshots, greatly improving the efficiency of data replication and test environment setup.

Meanwhile, as a bridge between local and cloud storage, HBlock also has the following features:

- **Data Cloud Migration:** HBlock can be integrated with S3-compatible object storage to create local LUNs (local mode) and cloud LUNs (cache mode and storage mode):
  - Local mode LUNs: All data is stored locally, which not only has the low latency and high availability of traditional hardware storage arrays, but also combines the high scalability and throughput of traditional distributed storage. The green feature greatly reduces the cost of user deployment, while the heterogeneous hardware feature improves the utilization of existing hardware resources.

- Storage mode LUNs: All data is not only stored locally, but also asynchronously replicated to object storage, achieving high local performance and remote data disaster recovery.
- Cache mode LUNs: Recently read and written data will be cached locally to maximize performance, while the full amount of data will be saved on object storage to reduce costs, allowing a small local capacity to store massive amounts of data. This is particularly suitable for businesses that do not require high real-time performance, such as data backup and archiving, as well as businesses that require more writing and less access, such as document files, medical imaging, video surveillance, etc. HBlock can seamlessly connect local applications with cloud storage, enabling on-demand use of storage space and elastic expansion.

- **Consistency:** HBlock utilizes the atomic operations of object storage, which can truly ensure the consistency of data on the cloud (i.e., data on the cloud is always a snapshot of local data), and there will be no situation where the entire business cannot be restored due to inconsistency of data on the cloud, thus ensuring data security.

1.2 HBlock vs Traditional Hardware Storage Array

	HBlock	Traditional Hardware Storage Array
Installation	Simple Delivered as a highly optimized zip package, requiring only 3 minutes from the unzipping of installation package to the initialization of the cluster.	Complex Involves data center, rack space, power, and network teams to prepare the environment, and requires on-site deployment. Several days are needed at least.
Hardware Requirements	Low With a process-level software-defined storage controller, commodity servers, bare metal servers, and virtual machines are supported.	High Dedicated hardware controller.
Hardware Resource Utilization	High Runs in the same Linux operating system instance concurrently with other applications. Allows each Linux operating system instance in the cluster to have different hardware configurations, such as different numbers of CPUs, different sizes of memory, and different capacities of local hard drives. A 2 GB memory is qualified for the minimum configuration.	Low Requires new hardware and monopolizes the entire hardware resource.
Horizontal Scalability	High Three to thousands of software-defined controllers share a capacity-unlimited virtual storage pool.	Low Generally, no more than 8 hardware controllers share a capacity-limited physical disk enclosure.
Aggregate Throughput	High No single bottleneck of throughput.	Low Limited bandwidth between controllers and disk enclosures is a bottleneck of throughput.
Delay	Low Sub-millisecond 4KiB random writes when using 10G ethernet interconnection and HDDs.	Low <0.1ms for 4KiB random writes when using backboard interconnection and SSDs.
Availability	High Failover in seconds without service disconnection.	High Failover in seconds without service disconnection.
Data Redundancy Protection	High Software RAID. Support both multi-copy and Erasure Code modes. Copies are placed across different servers.	Medium Hardware RAID only. Copies are placed in the same enclosure.
Total Costs	Low	High

Troubleshooting	Remote diagnosis	On-site vendor diagnosis
Application Scenarios	HA storage for small-scale distributed systems, utilization of existing resources, and independent management and control of new resources.	Critical workloads of enterprises.

1.3 HBlock vs Traditional Distributed Storage

	HBlock	Traditional Distributed Storage
Installation	Simple Delivered as a highly optimized zip package. Requiring no more than three minutes from the unzipping of installation package to the initialization of the cluster.	Complex Needs a lot of preparation work, such as configuring network connection, NTP server, and server access authentication.
Hardware Compatibility	High Compatible with any x86, ARM and Loongson hardware supported by Linux operating system. No device driver compatibility issue.	Medium Depending on specific hardware in the Hardware Compatibility List (HCL), driver compatibility issues often occur, and hardware and software vendors often blame each other.
Hardware Resource Utilization	High Runs in the same Linux operating system instance concurrently with other applications. Allows each Linux operating system instance in the cluster to have different hardware configurations, such as different numbers of CPUs, different sizes of memory, and different capacities of local hard drives. A 2 GB memory is qualified for the minimum configuration.	Low Dedicated VMs are needed. Each instance in the cluster has the same or similar hardware configuration.
Availability	High Failover in seconds without session disconnection.	Medium Virtual IPs (VIPs) are generally required. The traditional VIP mode adopts the design method of "nodes correspond to virtual IPs", which requires the authority to change the network configuration of the nodes. The tight coupling between the storage system and the network system is not conducive to system expansion and update.
Latency	Low Sub-millisecond 4KiB random writes when using 10G ethernet interconnection and HDDs.	High >1ms for 4KiB random write when using 10G ethernet interconnection and HDDs.
Data Redundancy Protection	High Software RAID. Supports both multi-copy and Erasure Code modes. Copies are placed across different servers.	Low Only supports multi-copy, and its disk utilization is lower than Erasure Code mode.
Horizontal Scalability	High	High Several servers to thousands of servers.

	Three to thousands of software-defined controllers share a capacity-unlimited virtual storage pool.	
Aggregate Throughput	High No single bottleneck of throughput.	High No single bottleneck of throughput.

2 Benefits

2.1 Easy Installation

You can directly download the installation package from our **official website** and install HBlock on the Linux operating system with just three commands. It takes no more than three minutes from decompressing the installation package to initializing the cluster. You can then enjoy the read and write speed and performance of local disks and expand your storage space based on demand.

If you only have one server, you can install the standalone model on the condition that the server has a single-core CPU, 2GB of memory, and 10G of available hard disk space, and you can store data after adding a data disk of at least 5GB.

2.1.1 Lightweight Installation Package

The HBlock installation zip package is highly optimized, making HBlock easy to install and deploy.

2.1.2 Independent of Hardware Drivers

HBlock is completely decoupled with hardware drivers, and it can be installed on the Linux operating system running on any physical server, bare metal server, or virtual machine. Users can freely use the latest hardware on the market with less vendor lock-in, and can build a cluster as long as the servers are interconnected.

2.2 High Utilization

2.2.1 Hybrid Deployment

HBlock is a user-mode process-level software-defined storage controller. Compared with other system-level, software-hardware-integrated or cloud service software-defined storage solutions, HBlock does not rely on or modify the operating system environment, does not monopolize the entire hard drive, and does not interfere with the execution of any other processes. Therefore, it can run in the same Linux operating system instance concurrently with other applications, helping users improve the utilization of existing hardware resource, and lower the barriers for potential users to try HBlock. Additionally, HBlock supports setting quotas for data path, and restricting disk space usage. When it is deployed alongside other businesses, this feature helps avoid contention for disk space.

2.2.2 Heterogeneous Hardware Deployment

A cluster can be composed of servers with different architectures, and each Linux operating system instance supports different hardware configurations, such as different numbers of CPUs, different memory sizes, and local hard drives with different capacities, improving the hardware resource usage.

2.2.3 Thin Provisioning

Thin provisioning offers applications with much more virtual storage space than that is available on the actual physical storage devices. HBlock allows the system to provide storage devices without occupying any physical storage space until data is written into logical volumes. HBlock's volumes are thin provisioned by default, which remarkably improves efficient storage space utilization.

2.3 Wide Compatibility

HBlock is compatible with mainstream Linux operating systems on commodity 64-bit x86 servers, ARM servers and Loongson servers. HBlock supports deployment on Linux operating systems of physical servers, bare metal servers, and VMs, and integrates these servers into high-performance virtual disks. HBlock also supports heterogeneous hardware and different operating systems. This means HBlock is much more versatile than other software-defined storage controllers.

2.3.1 Hardware Compatibility

Hardware compatibility includes:

- **CPU architecture:** Commodity x86 server, ARM server and Loongson server.
- **Storage medium:** NVMe SSDs, SAS SSDs, SATA SSDs, SAS HDDs, NL-SAS HDDs and SATA HDDs.

2.3.2 Software Compatibility

Software compatibility includes:

- **Operating system:** HBlock can be deployed on Linux operating system, independent of specific version of Linux kernel or distribution. The supported client operating systems include both Windows and Linux.
- **Virtualization platform:** Supports integration with both KVM and VMware virtualization platforms.
- **Database:** Supports a wide range of database applications, such as Oracle, MySQL, SQL Server, PostgreSQL, MongoDB and DB2.
- **Application:** Supports various enterprise IT applications, industry applications and web applications.
- **Cloud and container platform**: Provides plug-and-play OpenStack Cinder and Kubernetes CSI drivers.

2.4 High Availability

2.4.1 Second-level Failover

In cluster mode, a logical volume corresponds to at least two Targets: Active Target and Standby Target. When the active Target is unavailable due to the failure of the server where the active Target is located, HBlock will automatically switch to Standby Target within a few seconds without interrupting services. Additionally, HBlock supports multiple cold standby targets for a logical volume, ensuring uninterrupted data access even when multiple node failures occur in succession. This improves the availability of storage services.

The traditional Virtual IP (VIP) mode adopts the design method of "nodes correspond to virtual IPs", which requires the authority to change the network configuration of the nodes. The tight coupling between the storage system and the network system is not conducive to system expansion and update. HBlock adopts an advanced multi-controller architecture. It only needs to ensure that the client can connect to the IP of the servers where the active Target, standby Target and cold standby Targets are located, and it can achieve second-level failover through standard Multipath I/O (MPIO) technology without adding additional virtual IPs, proxy IPs, or changes to the network structure. This makes it easy to deploy alongside other business systems.

2.4.2 No Single Point of Failure

HBlock adopts distributed multi-controller architecture, and all services in the cluster are deployed in full redundancy mode. HBlock is designed to guarantee high availability when any single server, link or disk fault occurs in the cluster or in various uncertain environments such as unstable networks, unstable computing power, and unstable disks. Service availability will not be affected by any single point of failure.

2.4.3 Intelligent Speed Regulator

HBlock monitors the usage of disk space, memory, and other resources during reads/writes. When resources are insufficient, the speed regulator will automatically adjust the write speed to ensure that the disk is always writable, and the services are always available. While other storage products continue to write data when resources are insufficient, a sudden service interruption will occur when the disk is full.

2.5 High Reliability

2.5.1 Fault Domain

HBlock supports data path-level, server-level, rack-level and room-level fault domains, where copies or shards of source data are distributed across different fault domains to ensure uninterrupted business and zero data loss in the event of disk or server failures. Disks are added to the management scope of the HBlock cluster in the form of data paths. For environments with a small number of servers but a large number of disks, users can choose data path-level fault domain to store data on different disks. Users can choose the appropriate fault domain level based on their actual resources. The redundancy overlap feature of a LUN allows data replicas/shards to be stored in the same fault domain when there are insufficient fault domains. This improves system resource utilization, prevents data storage issues from insufficient fault domains, reduces user costs, enhances data availability, and boosts the system's overall flexibility.

2.5.2 Snapshot and Clone

HBlock supports snapshot creation for critical business data, capturing the instant state of LUNs to simplify backup, recovery, and version management, while ensuring data security. Snapshots use an incremental backup mechanism to save storage space. When data is accidentally deleted or logical errors occur, snapshots enable recovery within seconds, significantly reducing the risk of data loss. The clone function quickly creates test LUNs identical to the production environment. Without full-scale data copying, cloning time is remarkably shortened. Thus, it is suitable for development, testing, and data analysis.

2.5.3 Backup

Create snapshot-based backups that are independent of the source LUN. Both full and incremental backups are supported; incremental backups store only changed data, saving storage space. Backup files can be flexibly exported to various storage media (e.g., object storage, NAS) and imported back into LUNs—whether within the same cluster or across different clusters—to enable data migration and cross-cluster replication. This capability serves disaster recovery, testing, and development scenarios.

2.5.4 Full Support of Erasure Code

HBlock supports Erasure Code (EC), a data redundancy protection mechanism widely used in distributed storage, to enhance data redundancy and reliability. After the data is written into HBlock, EC mode can divide the original data into N pieces, and then generate M pieces of encoded data from these N pieces to obtain N + M pieces of data and store the data pieces on N + M different fault domains. EC N+M redundancy mode can tolerate M devices' failures in the fault domain at most, and the data of M pieces can be recovered through the data of other N pieces. Compared with the three-copy mode, EC mode delivers the same or even higher reliability, remarkably improves disk utilization, and dramatically saves TCO.

In the scenario of using HDDs and small blocks to read and write data, HBlock also supports Erasure Code mode, and can achieve low latency.

2.5.5 Zero Data Loss

HBlock supports multi-copy and Erasure Code data redundancy protection. Data is stored on different fault domains. In case of a fault domain, link or disk failure, HBlock uses the data pieces located on other fault domains, and starts data rebuilding/rebalancing in the background to redistribute data. None of the data is ever lost or temporarily unavailable.

HBlock may also encounter various uncertain situations when it is running. For example, high load caused by other applications in the servers, high CPU/memory usage, network exceptions such as packet loss or high latency, and other problems. These situations are collectively referred to as sub-healthy state. In sub-healthy state, HBlock can also ensure that data will not be lost, and services will not be stopped.

Besides, HBlock has its unique internal clock checking mechanism, which can determine the running time of each server. Unlike the traditional distributed storage requiring the Network Time Protocol (NTP), HBlock does not require the NTP service on any of the servers or synced server clocks. The clocks of the servers used to deploy HBlock can even be set arbitrarily, and data loss or service unavailability will not occur due to unsynced clocks.

2.5.6 Data Consistency

HBlock supports multiple data verification mechanisms, including client-to-server verification, cluster internal full-process data verification, and data verification among multiple servers to ensure data consistency.

- Data consistency among multiple servers: The data among multiple servers in the cluster will use the version number as the comparison standard, and the latest data is the one with the latest version number. This ensures strict data consistency. After an abnormal copy is found, HBlock will automatically repair the abnormal copy.
- The consistency of memory data and persistent data: HBlock scans the memory and disk data periodically. When the disk data is inaccessible or the verification fails, HBlock will automatically start the data recovery process to reconstruct the data.
- Consistency between cloud data and local data: HBlock utilizes atomic operations of OOS to ensure the consistency of cloud data (i.e., cloud data is always a snapshot of local data), and there will be no situation where the entire business cannot be restored due to inconsistency of cloud data, thus ensuring data security. Other storage products cannot utilize the atomic operation interface of cloud object storage, therefore cannot ensure the consistency of data on the cloud. Moreover, some storage products only store metadata in the local system, which can easily lead to metadata loss. HBlock stores metadata and data in the cloud and ensures consistency.

2.6 High Performance

2.6.1 Low Latency

HBlock can achieve ultra-low latency. For the cluster using normal 10G Ethernet interconnection and HDDs, the latency of random writing of 4KiB blocks can achieve sub-milliseconds, fully utilizing the maximum hardware speed.

2.6.2 Aggregated Throughput

The iSCSI Targets in HBlock can be created on any server in the cluster. When you create a LUN, HBlock will select servers with lower load in the cluster as the Target servers to balance system load. HBlock maximizes network bandwidth and disk throughput with no single bottleneck of throughput. However, for traditional hardware storage arrays, limited bandwidth between controllers and disk enclosures is a bottleneck of throughput.

2.6.3 Cache Storage Pool

HBlock supports multiple cache pools and can create storage pools with different performance based on disk read and write speeds. For example, using SSD disks to create storage pool pool1 and using HDD disks to create storage pool pool2. When creating a LUN, use pool1 as the cache pool and pool2 as the final storage pool. In this way, all I/O operations of the LUNs are first processed by the cache pool, and then the data is written back to the final storage pool. The data migration between the cache pool and the final storage pool is automatically triggered, allowing clients to enjoy high performance on the cache pool while reducing overall costs.

2.6.4 Data Rebuilding to Avoid Performance Bottlenecks

HBlock supports multi-copy and Erasure Code data redundancy protection, and data pieces are stored on different servers. When a disk or server is faulty or a new server is added to or removed from the cluster, HBlock will automatically start data rebuilding/rebalancing in the background to redistribute data. Since data pieces are distributed on multiple different servers, data rebuilding/rebalancing will be executed in parallel on multiple servers, thereby effectively avoiding the performance bottlenecks caused by a large amount of data rebuilding/rebalancing on a single server. The impact on the business is also minimized.

2.6.5 QoS

HBlock uses a token-bucket algorithm to enforce QoS policies in two dimensions—bandwidth and IOPS. Rules can be applied at three granularities: LUN, LUNs within a storage pool, or the entire storage pool, giving fine-grained control over client access to cluster resources. When burst traffic occurs, these rules prevent any client or workload from monopolizing bandwidth or IOPS, guaranteeing that critical applications always receive a predictable, pre-emptible resource lane and maintaining steady performance under mixed loads. In short, QoS eliminates resource contention and keeps business-critical performance stable.

2.7 Elastic Scalability

The HBlock architecture not only supports scale-up (expanding by adding a single server's processor, memory, network and disk), but also supports scale-out (expanding by adding servers). This means that HBlock can be independently expanded based on IOPS, storage space and bandwidth.

HBlock supports flexible expansion methods: adding new disks to increase the capacity of existing server or adding new servers. After the expansion, there is no need to relocate a large amount of data, and the system can automatically achieve load balance.
With HBlock, users are not required to make large upfront investment in the initial stage. You can use the commodity hardware, which is inexpensive and easy to use, and you can add servers and disks on demand at any time, without interrupting the business continuity.

2.8 Secure Authentication

HBlock supports Challenge-Handshake Authentication Protocol (CHAP). CHAP is a peer authentication protocol that allows an iSCSI client and Target to perform the secure authentication based on a password. CHAP includes both one-way and mutual authentication. With one-way CHAP, a Target authenticates a client initiator during connection. With mutual CHAP, the client and the Target authenticate each other based on their respective passwords. HBlock supports one-way CHAP now and will support mutual CHAP in the future.

2.9 Ease of Operation and Maintenance

2.9.1 Multiple Access Methods

Users can access HBlock management functions through the RESTful API, CLI, or the web console according to their operation requirements and operating environments.

2.9.2 Fault Warning

HBlock monitors all resources in the system to help you know resource usage and emergencies in real time. When the components or resources in the system are abnormal, HBlock will automatically send an email to notify the users.
Take the slow disk detection for example: after the disks have been working for a long time, problems such as aging of components might occur, which slows down the I/O response time, and eventually causes service unavailability. HBlock periodically performs disk detection, monitors, analyzes, and diagnoses read and write requests on disks to evaluate whether the disk is a slow disk, and notify users in time when a slow disk is found.

2.9.3 Support NAT Access

Normally, iSCSI initiator accesses the Targets in HBlock through an intranet. If Network Address Translation (NAT) is configured on the router of the intranet, the iSCSI initiator can connect with the server of the Target through an external network IP of NAT so that iSCSI can be delivered remotely as a cloud service through HBlock. Other storage products do not support users to set the portal IP of the Target, so they cannot support NAT access.

2.9.4 Application Scenario Adaptation

HBlock supports three different write policies. Users can set LUN-level write policy according to the specific application scenes. The three ways are as follows:

- WriteBack: After the data is written to the memory, it returns to the client successfully, and then the data is written to the disk asynchronously. It is suitable for scenarios with high performance requirements and low stability requirements.
- WriteThrough: The data is written to both memory and disk at the same time, and then returns to the client after successful writing. It is suitable for scenarios where the stability requirements are high, the write performance requirements are not high, and the recently written data will be read in a short time.

- WriteAround: Write data directly to the disk without writing to memory. It is suitable for scenarios with high stability requirements, low performance requirements, and more writing and less reading.

2.9.5 One-Click Tuning for Specific Business Scenarios

When HBlock is applied in intelligent computing or virtualization scenarios, it supports a one-click tuning function. This automatically adjusts system parameters to enhance the read/write performance of business data. Additionally, this feature allows for the configuration of HBlock's high-availability sensitivity levels. It adapts to different network configurations to ensure the system's high availability.

3 Application Scenarios

3.1 Utilization of Existing Resources

HBlock features broad compatibility and uniformly manages and integrates idle storage space in various servers into storage pools, and provides high-performance and HA virtual disks to other hosts through the iSCSI protocol. To meet the increasing demand for storage capacity raised by rapid service growth and solve the waste of resources due to idle server resources, HBlock provides rapid deployment and expansion solutions that improve the usage of storage resources without extra cost, and support rolling service updates. This way, HBlock meets the ever-changing requirements for capacity and performance of future business operations.

3.2 HA Storage for Small-scale Distributed Systems

HBlock manages the physical disks of application nodes and mounts the HA virtual disks to the application nodes. It makes it easier for applications to achieve HA and re-utilize the storage resources of application nodes. Furthermore, no additional procurement for the storage hardware is required, which reduces the total cost of ownership (TCO) for users.

3.3 Independent Management and Control of New Resources

When using HBlock to manage the storage resource pools that you have created, you will be able to have actual control over storage servers. This means you can use HBlock to manage storage resources and deploy other applications on your hardware to give full play to the hardware. Traditional integrated software and hardware storage products or distributed storage solutions require exclusive access to devices, and users can only perform limited operations on the management console. The storage cluster managed by HBlock enables users to comprehensively manage and control resource pools and enhances operating flexibility. You can select servers with any specifications for resource pool upgrading or expansion without limitations of vendor lock-in. You can also flexibly choose suitable hardware based on your business needs and budgets, improving the cost-effectiveness of investment.

3.4 Hybrid Cloud Storage

HBlock can manage both local storage resources and object storage resources simultaneously, achieving unified management of storage space and meeting customers' needs for hybrid cloud storage. For customers who need to store massive amounts of data, HBlock can seamlessly connect local applications with cloud storage, synchronize data to the cloud, and achieve on-demand use of storage space for elastic expansion. For scenarios where high data security is required and sensitive data is not suitable for cloud storage, HBlock can also help users achieve local data storage and improve data access speed. In addition, HBlock simplifies data management in hybrid cloud storage environments. Providing virtual targets and logical volumes for upper layer applications through the standard SCSI protocol, it can be deployed not only locally but also on private or public clouds.

3.5 Rapid Construction of a Production-like Environment

Based on its distributed block storage-derived multi-LUNs consistent snapshot capability, HBlock can instantly capture the data states of multiple applications in the upper-layer business system. When data is accidentally deleted or logical errors occur, snapshots enable second-level recovery, significantly reducing the risk of data loss. This makes it an ideal choice for data backup, recovery, and version management. Additionally, leveraging the capability to create clone LUNs in seconds, HBlock can rapidly replicate multiple independent test environments that are consistent with the production environment. This greatly improves the efficiency of data replication and environment setup, facilitating the verification of rapid application recovery and disaster recovery capabilities. Meanwhile, it enables efficient business testing, upgrade validation, and data analysis without disrupting the production environment.

3.6 Geo-Diverse Multi-Active High-Reliability Storage

HBlock optimizes the storage protocol stack and innovates in distributed algorithms to ensure cross-AZ strong data consistency (RPO=0) and achieve second-level fault recovery (RTO in seconds). The system can automatically detect faults, triggering cross-AZ data reconstruction. It uses intelligent data redundancy and redundancy overlap to ensure data security while reducing storage costs. In case of an AZ failure, services remain available with seamless business failover, requiring no manual intervention. This guarantees data security and business continuity.

3.7 Permanent Independent Backup of Critical Data

HBlock supports full and incremental backups via snapshot technology, guaranteeing long-term retention and complete isolation from production systems. Backup files can be flexibly exported to various storage media (e.g., object storage, NAS), so data is fully decoupled from the platform. Even if the primary system is totally destroyed or hit by ransomware, these standalone backups remain a trusted source for precise LUN-level recovery, ensuring zero loss of business-critical data. The solution is ideal for finance, healthcare, and other industries with strict long-term archival and compliance-audit requirements, providing rock-solid data protection.

3.8 Zero-Trust Storage Security Framework

HBlock builds an end-to-end zero-trust storage access fabric through a triple lock: iSCSI target allowlist, CHAP authentication, and QoS flow control.

- At LUN granularity, only clients whose IQN is on the allowlist can discover or connect—unauthorized hosts remain completely invisible.
- CHAP handshakes block spoofed identities and man-in-the-middle attempts before a single I/O is issued.
- Coupled with QoS policies that dynamically throttle or block abnormal traffic in real time across both IOPS and bandwidth dimensions.

This delivers a triple guarantee: zero impact on storage performance, airtight isolation of data and traffic, and uninterrupted operation of mission-critical workloads.

3.9 Cloud Data Differential Restoration

HBlock leverages the suspend/resume mechanism of cloud-bound LUNs to achieve cross-cluster disaster recovery switching and data flow management. It supports efficient cross-cluster data synchronization and rapid recovery, building a secure and efficient cloud-edge data collaboration system.

Based on the exclusive state control mechanism of cloud-bound LUNs, the suspend capability pauses interaction between the LUN and the cloud. Subsequently, the cloud data restoration function supports recovering cloud data on other clusters and continuing data read/write operations until the next suspend. When restoring a suspended LUN, intelligent incremental difference comparison and targeted differential data backhaul technologies are employed—only synchronizing incremental cloud data is required to complete LUN restoration.

This ensures that the same business LUN belongs to only a single cluster and remains readable/writable at any given time, fundamentally avoiding data conflict risks caused by multi-site writes. HBlock delivers core value in business continuity assurance, cross-cluster data security, and automated disaster recovery switching processes, providing secure, reliable, elastic, flexible, and visually controllable storage-layer disaster recovery capabilities for hybrid cloud and multi-active architectures.

4 Deployment Methods

- Independent Deployment

HBlock can be deployed on physical servers, bare metal servers, and virtual machines, providing both the low latency and high availability of traditional hardware storage arrays, as well as the high scalability and throughput of traditional distributed storage.

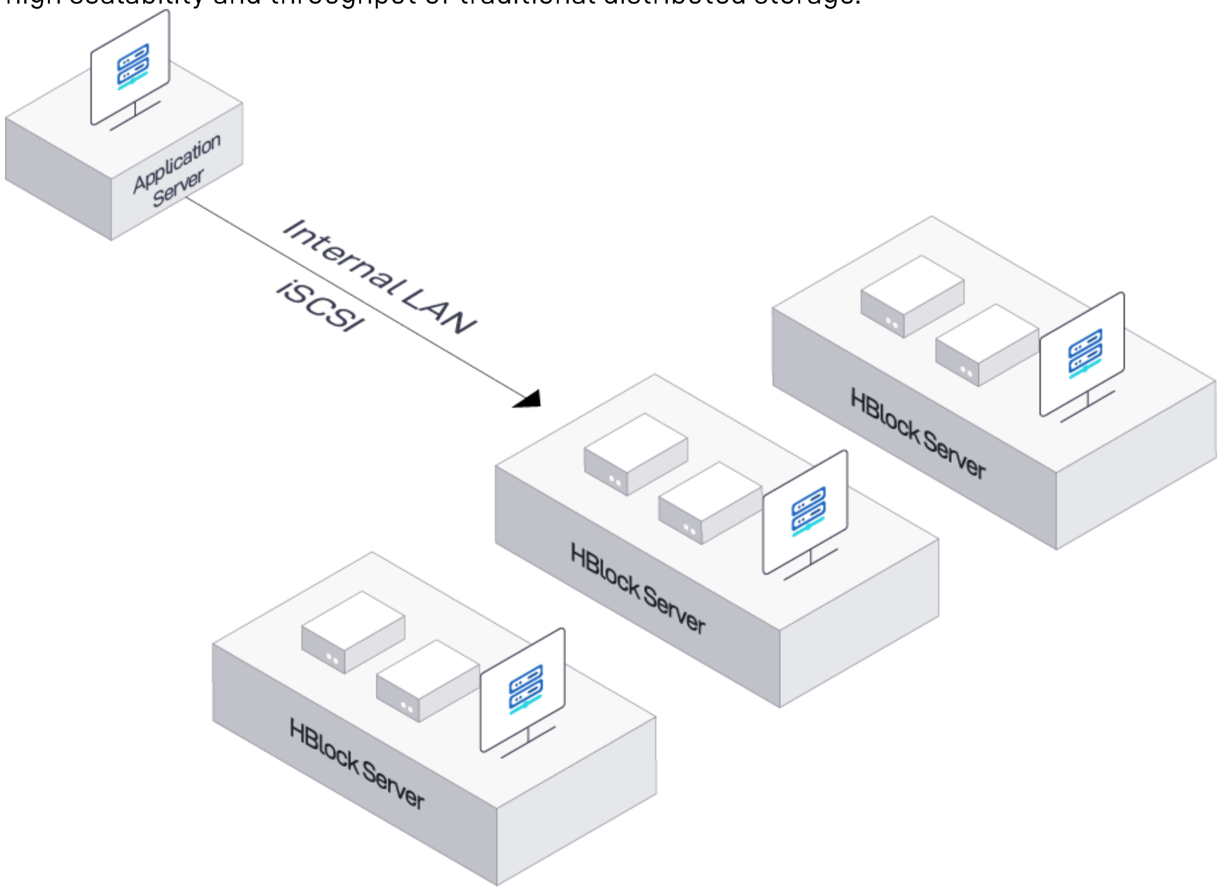

Figure 2 Independent Deployment of HBlock

5 Specifications

Architecture	Distributed multi-controller architecture
Supported Protocol	Standard iSCSI protocol
Number of Servers	Standalone and cluster modes: 3 to thousands of nodes
Number of LUNs	Unlimited
CPU Architecture	x86, ARM, Loongson
OS	CentOS 7/8/9 64-bit, or CTyunOS 2/4 64-bit
Hybrid Deployment	Supports running in the same Linux operating system instance concurrently with other applications.
Heterogeneous Hardware Deployment	Supports each Linux operating-system instance in the cluster having its own distinct hardware configuration.
High Availability	Supports MPIO
High Performance	Low latency and high throughput
Data Reliability	Multi-copy: three-copy Erasure Code: N+M redundancy Supports multi-level fault domains.
Data Protection	Supports data backup to eSurfing Cloud OOS or S3-compatible object storage. Supports volume-level snapshots and snapshot rollback. Also supports cloning and backup based on snapshots.
Storage Media	NVMe SSD, SAS SSD, SATA SSD, SAS HDD, NL-SAS HDD, SATA HDD
Network	TCP/IP