Data Warehouse Service

User Guide

Issue	01
Date	2025-09-24

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud 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.

Contents

1 What Is DWS?

Data Warehouse Service (DWS) is an online data analysis and processing database built on the Huawei Cloud infrastructure and platform. It offers scalable, ready-to-use, and fully managed analytical database services, and is compatible with ANSI/ISO SQL-92, SQL-99, and SQL:2003 syntax. Additionally, DWS is interoperable with other database ecosystems such as PostgreSQL, Oracle, Teradata, and MySQL. This makes it a competitive option for petabyte-scale big data analytics across diverse industries.

DWS offers both storage-compute coupled and decoupled data warehouses and helps you create a cutting-edge data warehouse that excels in enterprise-level kernels, real-time analysis, collaborative computing, convergent analysis, and cloud native capabilities. For details, see **Data Warehouse Types**.

- **Coupled storage and compute**: This type of data warehouse provides enterprise-level data warehouse services with high performance, high scalability, high reliability, high security, and easy O&M. It is capable of data analysis at a scale of 2,048 nodes and 20 petabytes of data and is suitable for converged analysis services that integrate databases, warehouses, marts, and lakes.
- **Decoupled storage and compute**: This type of data warehouse is designed with a cloud native architecture that separates storage and compute. It also features hierarchical auto scaling for computing and storage, as well as multi-logical cluster shared storage technology (Virtual Warehouse or VW). These capabilities allow for computing isolation and concurrent expansion to handle varying loads, making it an ideal choice for OLAP analysis scenarios.

DWS can fit into a wide range of fields, such as finance, Internet of Vehicles (IoV), government and enterprise, e-commerce, energy, and carrier. It has been listed in the Gartner Magic Quadrant for Data Management Solutions for Analytics for two consecutive years, thanks to its large-scale scalability, enterprise-grade reliability, and higher cost-effectiveness over conventional data warehouses.

Logical Cluster Architecture

Figure 1-1 shows the logical architecture of a DWS cluster. For details about the instance, see **Table 1-1**.

Figure 1-1 Logical cluster architecture

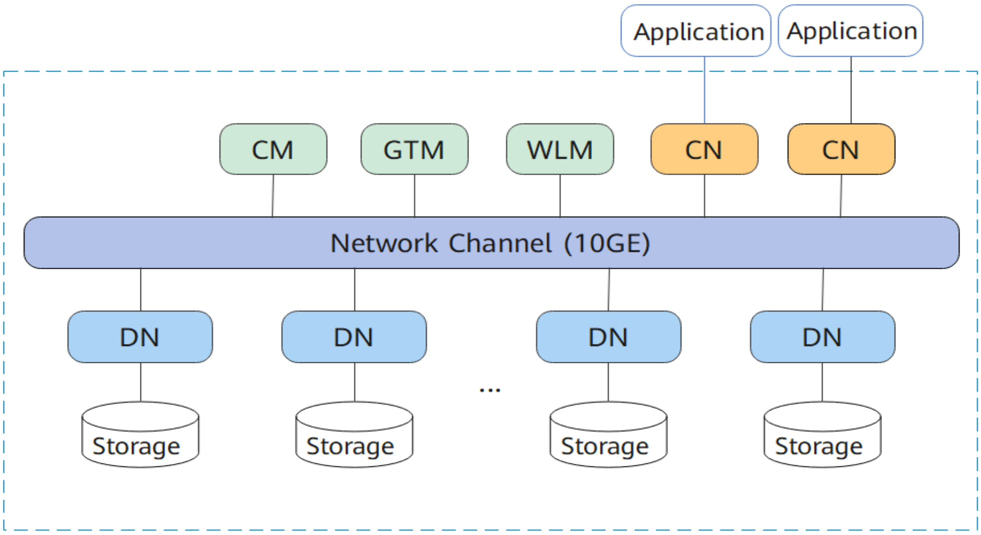


Table 1-1 Cluster architecture description

Name	Function	Description
Cluster Manager (CM)	Cluster Manager. It manages and monitors the running status of functional units and physical resources in the distributed system, ensuring system stability.	The CM consists of CM Agent, OM Monitor, and CM Server. ● CM Agent monitors the running status of primary and standby GTMs, CNs, and primary and standby DNs on the host, and reports the status to CM Server. In addition, it executes the arbitration instruction delivered by CM Server. A CM Agent process runs on each host. ● OM Monitor monitors scheduled tasks of CM Agent and restarts CM Agent when CM Agent stops. If CM Agent cannot be restarted, the host cannot be used. In this case, manually rectify this fault. **NOTE** CM Agent cannot be restarted probably because of insufficient system resources, which is not a common situation. ● CM Server checks whether the current system is normal according to the instance status reported by CM Agent. In the case of exceptions, CM Server delivers recovery commands to CM Agent. CM Servers are deployed in primary/standby pairs to ensure system high availability. CM Agent connects to the primary CM Server. If the primary CM Server is faulty, the standby CM Server is promoted to primary to prevent a single point of failure (SPOF).
Global Transaction Manager (GTM)	Generates and maintains the globally unique information, such as the transaction ID, transaction snapshot, and timestamp.	The cluster includes only one pair of GTMs: one primary GTM and one standby GTM.
Workload Manager (WLM)	Workload Manager. It controls allocation of system resources to prevent service congestion and system crash resulting from excessive workload.	You do not need to specify names of hosts where WLMs are to be deployed, because the installation program automatically installs a WLM on each host.

Name	Function	Description
Coordinator (CN)	A CN receives access requests from applications, and returns execution results to the client; splits tasks and allocates task fragments to different DNs for parallel processing.	CNs in a cluster have equivalent roles and return the same result for the same DML statement. Load balancers can be added between CNs and applications to ensure that CNs are transparent to applications. If a CN is faulty, the load balancer automatically connects the application to the other CN. For details, see **Associating and Disassociating ELB**. CNs need to connect to each other in the distributed transaction architecture. To reduce heavy load caused by excessive threads on GTMs, no more than 10 CNs should be configured in a cluster. DWS handles the global resource load in a cluster using the Central Coordinator (CCN) for adaptive dynamic load management. When the cluster is started for the first time, the CM selects the CN with the smallest ID as the CCN. If the CCN is faulty, CM replaces it with a new one.

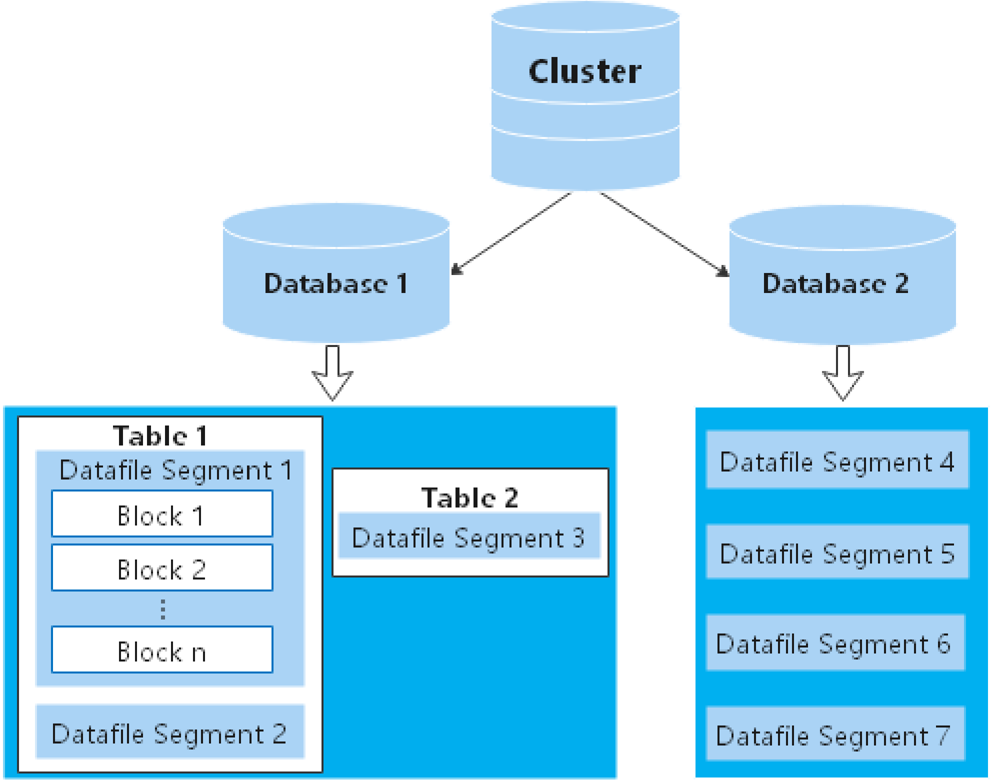
Name	Function	Description
Datanode (DN)	A DN stores data in row-store, column-store, or hybrid mode, executes data query tasks, and returns execution results to CNs.	There are multiple DNs in the cluster. Each DN stores part of data. DWS provides DN high availability: active DN, standby DN, and secondary DN. The working principles of the three are as follows: ● During data synchronization, if the active DN suddenly becomes faulty, the standby DN is switched to the active state. ● Before the faulty active DN recovers, the new active DN synchronizes data logs to the secondary DN. ● After the faulty active DN recovers, it becomes the standby DN and uses data logs stored on the secondary DN to restore data generated during its faulty period. The secondary DN serves exclusively as a backup, never ascending to active or standby status in case of faults. It conserves storage by only holding Xlog data transferred from the new active DN and data replicated during original active DN failures. This efficient approach saves one-third of the storage space compared to conventional tri-backup methods.
Storage	Functions as the server's local storage resources to store data permanently.	-

DNs in a cluster store data on disks. **Figure 1-2** describes the objects on each DN and the relationships among them logically.

- A database manages various data objects and is isolated from other databases.
- A datafile segment stores data in only one table. A table containing more than 1 GB of data is stored in multiple data file segments.
- A table belongs only to one database.
- A block is the basic unit of database management, with a default size of 8 KB.

Data can be distributed in replication, round-robin, or hash mode. You can specify the distribution mode during table creation.

Figure 1-2 Logical database architecture

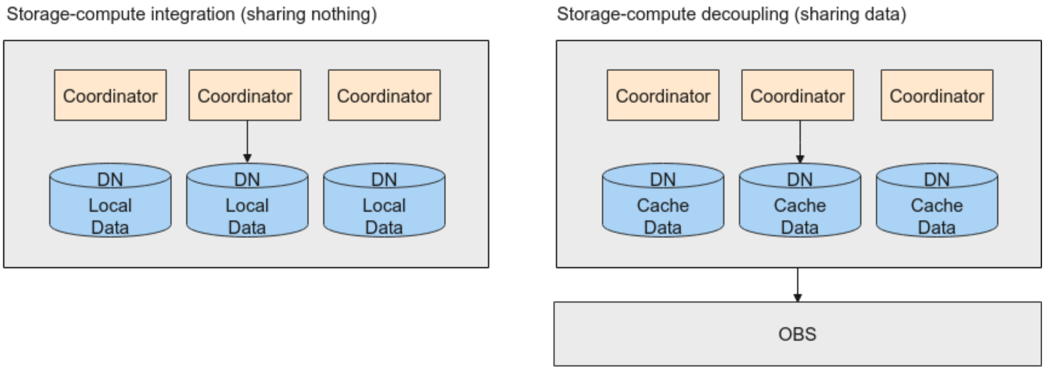


Physical Architecture of a Cluster

DWS supports the storage-compute coupled and decoupled architectures.

In the storage-compute coupled architecture, data is stored on local disks of DNs. In the storage-compute decoupled architecture, local DN disks are used only for data cache and metadata storage, and user data is stored on OBS. You can select an architecture as required.

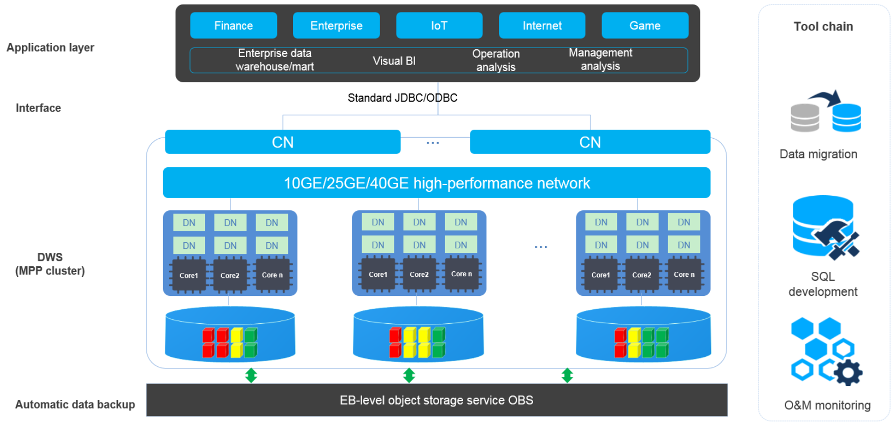
Figure 1-3 Architecture selection



Storage-Compute Coupled Architecture

DWS employs the shared-nothing architecture and the massively parallel processing (MPP) engine, and consists of numerous independent logical nodes that do not share the system resources such as CPUs, memory, and storage. In such a system architecture, service data is separately stored on numerous nodes. Data analysis tasks are executed in parallel on the nodes where data is stored. The massively parallel data processing significantly improves response speed.

Figure 1-4 Architecture



- **Application layer**

  Data loading tools, Extract-Transform-Load (ETL) tools, Business Intelligence (BI) tools, and data mining and analysis tools can be integrated with DWS through standard interfaces. DWS is compatible with the PostgreSQL ecosystem, and the SQL syntax is compatible with Oracle, MySQL, and Teradata. Applications can be smoothly migrated to DWS with only a few changes.

- **API**

  Applications can connect to DWS through standard JDBC and ODBC.

- **DWS**

  A DWS cluster contains nodes of the same flavor in the same subnet. These nodes jointly provide services. Datanodes (DNs) in a cluster store data on disks. CNs, or Coordinators, receive access requests from the clients and return the execution results. They also split and distribute tasks to the Datanodes (DNs) for parallel execution.

- **Automatic data backup**

  Cluster snapshots can be automatically backed up to the EB-level Object Storage Service (OBS), which facilitates periodic backup of the cluster during off-peak hours, ensuring data recovery after a cluster exception occurs.

  A snapshot is a complete backup of DWS at a specified time point. It records all configuration data and service data of the cluster at the specified moment.

- **Tool chain**

  The parallel data loading tool General Data Service (GDS), SQL syntax migration tool Database Schema Convertor (DSC), and SQL development tool Data Studio are provided. The cluster O&M can be monitored on a console.

Storage-Compute Decoupled Architecture

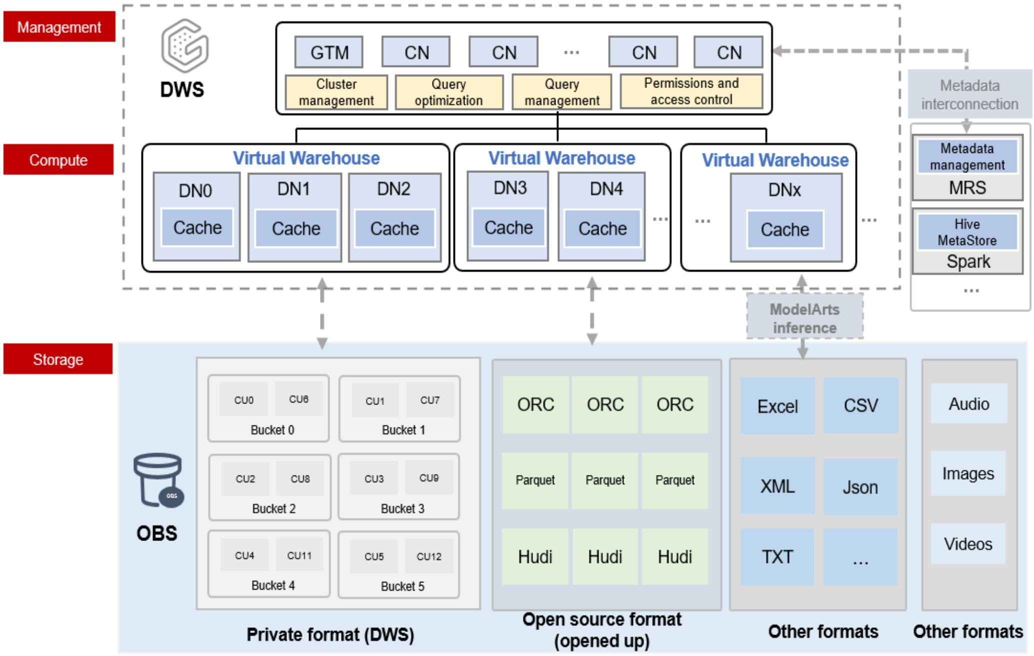
The newly released DWS storage-compute decoupled cloud-native data warehouse provides resource pooling, massive storage, and the MPP architecture with decoupled compute and storage. This enables high elasticity, real-time data import and sharing, and lake warehouse integration.

Cloud-native data warehouse allows for separate scaling of compute and storage resources through decoupled compute and storage. Users can easily adjust their computing capabilities during peak and off-peak hours. Additionally, storage can be expanded limitlessly and paid for on-demand, allowing for quick and flexible responses to service changes while maintaining cost-effectiveness.

The storage-compute decoupled architecture has the following advantages:

- **Lakehouse:** It simplifies the maintenance and operation of an integrated lakehouse. It seamlessly integrates with DLI, supports automatic metadata import, accelerates external table queries, enables joined queries of internal and external tables, and allows for reading and writing of data lake formats, as well as easier data import.
- **Real-time write:** It provides the H-Store storage engine which optimizes real-time data writes and supports high-throughput real-time batch writes and updates.
- **High elasticity:** Scaling compute resources and using on-demand storage can result in significant cost savings. Historical data does not need to be migrated to other storage media, enabling one-stop data analysis for industries such as finance and Internet.
- **Data sharing:** Multiple loads share one copy of data in real time, while the computing resources are isolated. Multiple writes and reads are supported.

Figure 1-5 Storage-compute decoupled architecture



- Superb scalability
  - Logical clusters, known as Virtual Warehouses (VWs), can be expanded concurrently based on service requirements.
  - Data is shared among multiple VWs in real-time, eliminating the need for data duplication.
  - Multiple VWs enhance throughput and concurrency while providing excellent read/write and load isolation.
- Lakehouse
  - Seamless hybrid query across data lakes and data warehouses
  - In data lake analysis, you can enjoy the ultimate performance and precise control of data warehouses.

Comparison Between Storage-Compute Coupled and Decoupled Architectures

Table 1-2 Differences between storage-compute coupled and decoupled architectures

Version	Coupled storage and compute	Decoupled storage and compute
Storage medium	Data is stored on the local disks.of compute nodes. The local disks can be cloud SSDs or local SSDs, depending on the storage type you select. ● Cloud SSDs are recommended as they use EVS disks as the data storage media, and their capacity can be expanded flexibly. ● Local SSDs use the local disks of ECSs as the data storage media. Their capacity cannot be expanded, and their specifications cannot be changed. If the storage capacity is insufficient, you can only add nodes.	Column-store data is stored in Huawei Cloud Object Storage Service (OBS). Local disks of compute nodes are used as the query cache of OBS data. Row-store data is still stored in local disks of compute nodes.

Advantage	Data is stored on local disks of compute nodes, providing high performance.	The architecture separates storage and compute, offering layered elasticity, on-demand storage use, rapid compute scaling, unlimited computing power, and capacity. Data stored on object storage reduces costs and multiple VWs support higher concurrency. Data sharing and lakehouse integration.

2 Data Warehouse Types

Product Type Overview

- **Coupled storage and compute**: The storage-compute coupled data warehouse provides enterprise-level data warehouse services with high performance, high scalability, high reliability, high security, low latency, and easy O&M. It is capable of data analysis at a scale of 2,048 nodes and 20 petabytes of data and is suitable for converged analysis services that integrate databases, warehouses, marts, and lakes.
- **Decoupled storage and compute**: The storage-compute decoupled data warehouse is designed with a cloud native architecture that separates storage and compute. It also features hierarchical auto scaling for computing and storage, as well as multi-logical cluster shared storage technology (Virtual Warehouse or VW). These capabilities allow for computing isolation and concurrent expansion to handle varying loads, making it an ideal choice for OLAP analysis scenarios.

NOTE

- DWS data warehouses cannot access each other. You can create an OBS foreign table to associate two databases in the same data directory for data query.

Features

Table 2-1 Features

Module	Function	Coupled Storage and Compute	Decoupled Storage and Compute
Dashboard	Resources	Yes	Yes
	Alarms	Yes	Yes
	Recent events	Yes	Yes
	Cluster monitoring metrics (DMS)	Yes	Yes
Data	-	Yes	Yes

Module	Function	Coupled Storage and Compute	Decoupled Storage and Compute
Cluster management	SQL editor	Yes	Yes
	Monitoring panel (DMS)	Yes	Yes
	Monitoring metrics (Cloud Eye)	Yes	Yes
	Restart	Yes	Yes
	Start	Yes	Yes
	Stop	Yes	Yes
	Scaling	Yes	Yes
	Scale-in	Yes	Yes
	Redistributing data	Yes	Yes (Note 1)
	Viewing redistribution details	Yes	Yes
	Changing the node flavor	Yes	Yes
	Changing all specifications	Yes	No
	Resetting passwords	Yes	Yes
	Creating snapshots	Yes	Yes
	Canceling read-only status	Yes	Yes
	Deletion	Yes	Yes
	Managing CNs	Yes	Yes
	Storage space scaling	Yes	Yes
Basic Information	Basic information	Yes	Yes
	ELB	Yes	Yes
	Resource Management	Yes	Yes
	Intelligent O&M	Yes	Yes
	Logical cluster	Yes	Yes

Module	Function	Coupled Storage and Compute	Decoupled Storage and Compute
	Snapshot	Yes	Yes
	Parameter modifications	Yes	Yes
	Security settings	Yes	Yes
	MRS data sources	Yes	Yes
	Tags	Yes	Yes
	Node management	Yes	Yes
	Upgrade management	Yes	Yes
	Logging	Yes	Yes
	User management	Yes	Yes
Integration	Data migration	Yes	Yes
DR management	DR management	Yes	No
Snapshot management	Restoration	Yes	Yes
	Deletion	Yes	Yes
	Copy	Yes	Yes
Incident management	Event management (general)	Yes	Yes
Alarm management	Alarm management	Yes	Yes
Client connections	Client connections	Yes	Yes
Others	Inspection	Yes	Yes
	Intelligent O&M	Yes	Yes
	Node restoration	Yes	Yes
	Warm backup on the tenant side	Yes	Yes

NOTE

- Note 1: The storage-compute decoupled table stores data on OBS, eliminating the need for redistribution. However, metadata and indexes are stored locally and must still be redistributed. Redistributing a table with decoupled storage and compute nodes only allows for read operations, and metadata redistribution is fast. However, creating an index on the table can affect redistribution performance, with completion time increasing with index data volume. During this period, the table is read-only and cannot be modified.
- Only storage-compute decoupled clusters of 9.0.2 and later versions support the snapshot function.

3 Data Warehouse Flavors

DWS provides storage-compute coupled and decoupled data warehouses. For details about the differences between them, see **Data Warehouse Types**.

NOTE

You are advised not to use clusters with low specifications, such as clusters with 16 GB memory and 4-core vCPUs, in the production environment. Otherwise, resource overload may occur.

Flavors for Storage-Compute Coupled Clusters

- A storage-compute coupled data warehouse using cloud disks with a vCPU to memory ratio of 1:8 can be elastically scaled, providing unlimited computing and storage capacity. For details, see **Table 3-1**.
- A storage-compute coupled data warehouse using cloud disks with a vCPU to memory ratio of 1:4 provides high-concurrency, high-performance, and low-latency transaction processing capabilities at low costs based on large-scale data query and analysis capabilities. This type of data warehouse is ideal for HTAP hybrid load scenarios. For details about the specifications, see **Table 3-2**.
- A storage-compute coupled data warehouse using local disks cannot be scaled up. You can only increase capacity by adding more nodes. For details, see **Table 3-3**.

NOTE

Step indicates the interval for increasing or decreasing the disk size during cluster configuration change. You need to select a value based on the storage step of the corresponding flavor.

Table 3-1 Cloud disk flavors with a vCPU to memory ratio of 1:8 for storage-compute clusters

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Default Storage	Step (GB)	Recommended Storage	Number of DNs	Scenario
dwsx2.xlarge.m7	x86	4	32	20 GB–2,000 GB	100	10	800	1	Suitable for DWS starters. These flavors can be used for testing, learning environments, or small-scale analytics systems.
dwsk2.xlarge	Arm	4	32	20 GB–2,000 GB	100	10	800	1	
dwsx2.xlarge.m7n	x86	4	32	20 GB–2,000 GB	100	10	800	1	
dwsk2.xlarge.km2	Arm	4	32	20 GB–2,000 GB	100	10	800	1	
dwsx2.2xlarge.m7	x86	8	64	100 GB–4,000 GB	200	100	1,600	1	Suitable for internal data warehousing and report analysis in small- and medium-sized enterprises (SMEs).
dwsk2.2xlarge	Arm	8	64	100 GB–4,000 GB	200	100	1,600	1	
dwsx2.2xlarge.m7n	x86	8	64	100 GB–4,000 GB	200	100	1,600	1	
dwsk2.2xlarge.km2	Arm	8	64	100 GB–4,000 GB	200	100	1,600	1	
dwsx2.4xlarge.m7	x86	16	128	100 GB–8,000 GB	400	100	3,200	1	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Default Storage	Step (GB)	Recommended Storage	Number of DNs	Scenario
dwsk2.4xlarge	Arm	16	128	100 GB–8,000 GB	400	100	3,200	1	
dwsk2.4xlarge.km2	Arm	16	128	100 GB–8,000 GB	400	100	3,200	1	
dwsx2.8xlarge.m7	x86	32	256	100 GB–16,000 GB	800	100	6,400	2	Recommended for the production environment. These flavors are applicable to OLAP systems that have to deal with large data volumes, BI reports, and data visualizations on large screens for most companies.
dwsk2.8xlarge	Arm	32	256	100 GB–16,000 GB	800	100	6,400	2	
dwsx2.8xlarge.m7n	x86	32	256	100 GB–16,000 GB	800	100	6,400	2	
dwsk2.8xlarge.km2	Arm	32	256	100 GB–16,000 GB	800	100	6,400	2	
dwsk2.12xlarge	Arm	48	384	100 GB–24,000 GB	1200	100	9,600	4	These flavors can deliver excellent performance and are applicable to high-throughput data warehouse processing and high-concurrency online query.
dwsx2.16xlarge.m7	x86	64	512	100 GB–32,000 GB	1,600	100	12,800	4	
dwsx2.16xlarge.m7n	x86	64	512	100 GB–32,000 GB	1,600	100	12,800	4	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Default Storage	Step (GB)	Recommended Storage	Number of DNs	Scenario
dwsx2.16xlarge.m7	x86	64	512	100 GB–32,000 GB	1,600	100	12,800	4	
dwsk2.16xlarge	Arm	64	512	100 GB–32,000 GB	1,600	100	12,800	4	
dwsx2.24xlarge.m7	x86	96	768	100 GB–48,000 GB	2,400	100	19,200	4	
dwsk2.24xlarge	Arm	96	768	100 GB–48,000 GB	2,400	100	19,200	4	
dwsx2.32xlarge.m7	x86	128	1,024	100 GB–48,000 GB	3,200	100	25,600	4	

Table 3-2 Cloud disk flavors with a vCPU to memory ratio of 1:4 for storage-compute clusters

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	Number of DNs	Scenario
dwsx2.h.xlarge.4.c7	x86	4	16	20 GB–2,000 GB	20	1	Suitable for DWS starters. These flavors can be used for testing, learning environments, or small-scale analytics systems.
dwsk2.h.xlarge.4.kc1	Arm	4	16	20 GB–2,000 GB	20	1	
dwsk2.h.xlarge.kc2	Arm	4	16	20 GB–2,000 GB	20	1	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	Number of DNs	Scenario
dwsx2.h.xlarge.4.c7n	x86	4	16	20 GB–2,000 GB	20	1	
dwsx2.h.2xlarge.4.c6	x86	8	32	100 GB–4,000 GB	100	1	Suitable for internal data warehousing and report analysis in small- and medium-sized enterprises (SMEs).
dwsx2.h.2xlarge.4.c7	x86	8	32	100 GB–4,000 GB	100	1	
dwsk2.h.2xlarge.4.kc1	Arm	8	32	100 GB–4,000 GB	100	1	
dwsk2.h.2xlarge.kc2	Arm	8	32	100 GB–4,000 GB	100	1	
dwsx2.h.2xlarge.4.c7n	x86	8	32	100 GB–4,000 GB	100	1	
dwsx2.h.4xlarge.4.c7	x86	16	64	100 GB–8,000 GB	100	1	Recommended for the production environment. These flavors are applicable to OLAP systems that have to deal with large data volumes, BI reports, and data visualizations on large screens for most companies.
dwsk2.h.4xlarge.4.kc1	Arm	16	64	100 GB–8,000 GB	100	1	
dwsk2.h.4xlarge.kc2	Arm	16	64	100 GB–8,000 GB	100	1	
dwsx2.h.4xlarge.4.c7n	x86	16	64	100 GB–8,000 GB	100	1	
dwsx2.h.8xlarge.4.c7	x86	32	128	100 GB–16,000 GB	100	2	
dwsk2.h.8xlarge.4.kc1	Arm	32	128	100 GB–16,000 GB	100	2	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	Number of DNs	Scenario
dwsk2.h.8xlarge.kc2	Arm	32	128	100 GB–16,000 GB	100	2	
dwsx2.h.8xlarge.4.c7n	x86	32	128	100 GB–16,000 GB	100	2	
dwsk2.h.12xlarge.4.kc1	Arm	48	192	100 GB–24,000 GB	100	4	These flavors can deliver excellent performance and are applicable to high-throughput data warehouse processing and high-concurrency online query.
dwsk2.h.12xlarge.kc2	Arm	48	192	100 GB–24,000 GB	100	4	
dwsx2.h.16xlarge.4.c7	x86	64	256	100 GB–32,000 GB	100	4	
dwsx2.h.16xlarge.4.c7n	x86	64	256	100 GB–32,000 GB	100	4	
dwsk2.h.16xlarge	Arm	64	256	100 GB–32,000 GB	100	4	
dwsk2.h.24xlarge	Arm	96	384	100 GB–48,000 GB	100	4	
dwsk2.h.32xlarge	Arm	128	512	100 GB–64,000 GB	100	4	

Table 3-3 Local disk flavors for storage-compute coupled clusters

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Number of DNs	Scenario
dws2.olap.4xlarge.i3	x86	16	128	1,490 GB	1	Recommended for the production environment. These flavors are applicable to OLAP systems that have to deal with large data volumes, BI reports, and data visualizations on large screens for most companies. These flavors can deliver excellent performance and are applicable to high-throughput data warehouse processing and high-concurrency online query.
dws2.olap.4xlarge.ki1	Arm	16	64	2,980 GB	1	
dws2.olap.8xlarge.i3	x86	32	256	2,980 GB	2	
dws2.olap.8xlarge.ki1	Arm	32	128	5,960 GB	2	
dws2.olap.16xlarge.i3	x86	64	512	5,960 GB	4	
dws2.olap.16xlarge.ki1	Arm	64	228	11,921 GB	4	

Flavors for Storage-Compute Decoupled Clusters

- A storage-compute decoupled data warehouse using cloud disks can be elastically scaled, providing unlimited computing and storage capacity. For details, see **Table 3-4**.
- A storage-compute decoupled data warehouse using local disks has a fixed storage capacity that cannot be expanded or modified. You can only increase capacity by adding more nodes. For details, see **Table 3-5**.

NOTE

When creating a storage-compute decoupled cluster, only the second half of the flavors (for example, **4U16G.4DPU**) are shown. The prefixes (**dwsx3**/**dwsax3**/**dwsk3**) in the flavor list indicate the storage-compute decoupled CPU architecture.

Table 3-4 Storage-compute decoupled cloud disk flavors

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	Number of DNs	Scenario
dwsx3.4U16G.4DPU	x86	4	16	20 GB–2,000 GB	10	1	Suitable for DWS starters. These flavors can be used for testing, learning environments, or small-scale analytics systems.
dwsk3.4U16G.4DPU	Arm	4	16	20 GB–2,000 GB	10	1	
dwsax3.4U16G.4DPU	x86	4	16	20 GB–2,000 GB	10	1	
dwsax3.4U32G.4DPU	x86	4	32	20 GB–2,000 GB	10	1	
dwsx3.8U32G.8DPU	x86	8	32	100 GB–4,000 GB	100	1	Suitable for internal data warehousing and report analysis in small- and medium-sized enterprises (SMEs).
dwsk3.8U32G.8DPU	Arm	8	32	100 GB–4,000 GB	100	1	
dwsax3.8U32G.8DPU	x86	8	32	100 GB–4,000 GB	100	1	
dwsax3.8U64G.8DPU	x86	8	64	100 GB–4,000 GB	100	1	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	Number of DNs	Scenario
dwsx3.16U64G.16DPU	x86	16	64	100 GB–8,000 GB	100	1	Recommended for the production environment. These flavors are applicable to OLAP systems that have to deal with large data volumes, BI reports, and data visualizations on large screens for most companies.
dwsk3.16U64G.16DPU	Arm	16	64	100 GB–8,000 GB	100	1	
dwsax3.16U64G.16DPU	x86	16	64	100 GB–8,000 GB	100	1	
dwsax3.16U128G.16DPU	x86	16	128	100 GB–8,000 GB	100	1	
dwsx3.32U128G.32DPU	x86	32	128	100 GB–16,000 GB	100	2	
dwsk3.32U128G.32DPU	Arm	32	128	100 GB–16,000 GB	100	2	
dwsax3.32U128G.32DPU	x86	32	128	100 GB–16,000 GB	100	2	
dwsax3.32U256G.32DPU	x86	32	256	100 GB–16,000 GB	100	2	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	Number of DNs	Scenario
dwsk3.48U192G.48DPU	Arm	48	192	200 GB–24,000 GB	100	4	These flavors can deliver excellent performance and are applicable to high-throughput data warehouse processing and high-concurrency online query.
dwsx3.64U256G.64DPU	x86	64	256	200 GB–32,000 GB	100	4	
dwsk3.64U256G.64DPU	Arm	64	256	100 GB–32,000 GB	100	4	
dwsax3.64U256G.64DPU	x86	64	256	100 GB–32,000 GB	100	4	
dwsax3.64U512G.64DPU	x86	64	512	100 GB–32,000 GB	100	4	
dwsx3.96U768G.96DPU	x86	96	768	100 GB–48,000 GB	100	4	
dwsk3.96U384G.96DPU	Arm	96	384	100 GB–48,000 GB	100	4	
dwsax3.96U384G.96DPU	x86	96	384	100 GB–48,000 GB	100	4	
dwsax3.96U768G.96DPU	x86	96	768	100 GB–48,000 GB	100	4	
dwsx3.128U1024G.128DPU	x86	128	1,024	100 GB–64,000 GB	100	4	
dwsk3.128U512G.128DPU	Arm	128	512	100 GB–64,000 GB	100	4	
dwsax3.128U512G.128DPU	x86	128	512	100 GB–64,000 GB	100	4	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	Number of DNs	Scenario
dwsax3.128U1024G.128DPU	x86	128	1,024	100 GB–64,000 GB	100	4	

Table 3-5 Storage-compute decoupled local disk flavors

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Number of DNs	Scenario
dws3.16U128G.i7.16DPU	x86	16	128	2,980 GB	1	Recommended for the production environment. These flavors are applicable to OLAP systems that have to deal with large data volumes, BI reports, and data visualizations on large screens for most companies. These flavors can deliver excellent performance and are applicable to high-throughput data warehouse processing and high-concurrency online query.
dws3.16U64G.ki1.16DPU	Arm	16	64	5,960 GB	1	
dws3.32U256G.i7.32DPU	x86	32	256	5,960 GB	2	
dws3.32U128G.ki1.32DPU	Arm	32	128	11,920 GB	2	
dws3.64U512G.i7.64DPU	x86	64	512	11,920 GB	4	
dws3.64U228G.ki1.64DPU	Arm	64	228	23,840 GB	4	

4 Advantages

DWS supports ANSI/ISO SQL-92, SQL-99, and SQL-2003 syntax, as well as the PostgreSQL, Oracle, Teradata, and MySQL database ecosystems. It offers powerful solutions for analyzing massive amounts of data in different industries, even at the petabyte scale.

DWS outperforms conventional data warehouses in hyper-scale data processing and general platform management due to the following features:

Ease of use

- Visualized one-stop management

  DWS helps you easily complete the entire process, from project concept to production deployment. The DWS console allows you to quickly set up a high-performance and highly available enterprise-level data warehouse cluster in just a few minutes, without requiring any data warehouse software or servers.

  With just a few clicks, you can easily connect applications to the data warehouse, back up data, restore data, and monitor data warehouse resources and performance.

- Seamless integration with big data

  Without the need to migrate data, you can use standard SQL statements to directly query data on HDFS and OBS.

- Heterogeneous database migration tools

  DWS provides various migration tools to migrate SQL scripts of Oracle and Teradata to DWS.

High performance

- Cloud-based distributed architecture

  DWS adopts the MPP architecture so that service data is separately stored on numerous nodes. Data analytics tasks are quickly executed in parallel on the nodes where data is stored.

- Query response to trillions of data records within seconds

  DWS improves data query performance by executing multi-thread operators in parallel, running commands in registers in parallel with the vectorized computing engine, and reducing redundant judgment conditions using LLVM.

DWS provides you with a better data compression ratio (column-store), higher index performance (column-store), and better point update and query (row-store) performance.

Furthermore, DWS has achieved a significant breakthrough in overcoming the performance limitations of traditional column-store execution engines. Unlike the original column-store engine, the Turbo engine enhances both memory and disk storage formats for string and numeric data types. Additionally, it optimizes the performance of key operators, such as sorting, aggregation, join, and scanning, effectively doubling the overall performance of the executor and significantly reducing service computing costs.

- Fast data loading

  DWS provides you with GDS, a high-speed parallel bulk data loading tool.

- Data Compression in Column Storage

  To compress old and inactive data to save space and reduce procurement and O&M costs.

  In DWS, data can be compressed using the Delta Value Encoding, Dictionary, RLE, LZ4, and ZLIB algorithms. The system automatically selects a compression algorithm based on data characteristics. The average compression ratio is 7:1. Compressed data can be directly accessed and is transparent to services, greatly reducing the preparation time before accessing historical data.

High scalability

- On-demand scale-out: With the shared-nothing open architecture, nodes can be added at any time to enhance the data storage, query, and analysis capabilities of the system.
- Enhanced linear performance after scale-out: The capacity and performance increase linearly with the cluster scale. The linear rate is 0.8.
- Service continuity: During scale-out, data can be added, deleted, modified, and queried, and DDL operations (**DROP**/**TRUNCATE**/**ALTER TABLE**) can be performed. Table-level scale-out ensures service continuity.
- Online upgrade: Upgrading major versions online from 8.1.1 and performing online patch upgrades from 8.1.3 and later versions is now possible without interrupting your services. Any interruptions will only last a few seconds.

Robust reliability

- Transaction management
  - Transaction blocks are supported. You can run **start transaction** to explicitly start a transaction block.
  - Single-statement transactions are supported. If you do not explicitly start a transaction, a single statement is processed as a transaction.
  - Distributed transaction management and global transaction information management are supported. This includes gxid, snapshot, timestamp management, distributed transaction status management, and gxid overflow processing.
  - The atomicity, consistency, isolation, and durability (ACID) feature is supported, which ensures strong data consistency for distributed transactions.

  - Deadlocks are prevented in the distributed system. A transaction will be unlocked immediately after a deadlock (if any).

- Comprehensive HA design

  All software processes of DWS are in active/standby mode. Logical components such as the CNs and DNs of each cluster also work in active/standby mode. This ensures data reliability and consistency when any single point of failure (SPOF) occurs.

- High security

  DWS supports transparent data encryption and can interconnect with the Database Security Service (DBSS) to better protect user privacy and data security with network isolation and security group rule setting options. In addition, DWS supports automatic full and incremental backup of data for higher reliability.

Low cost

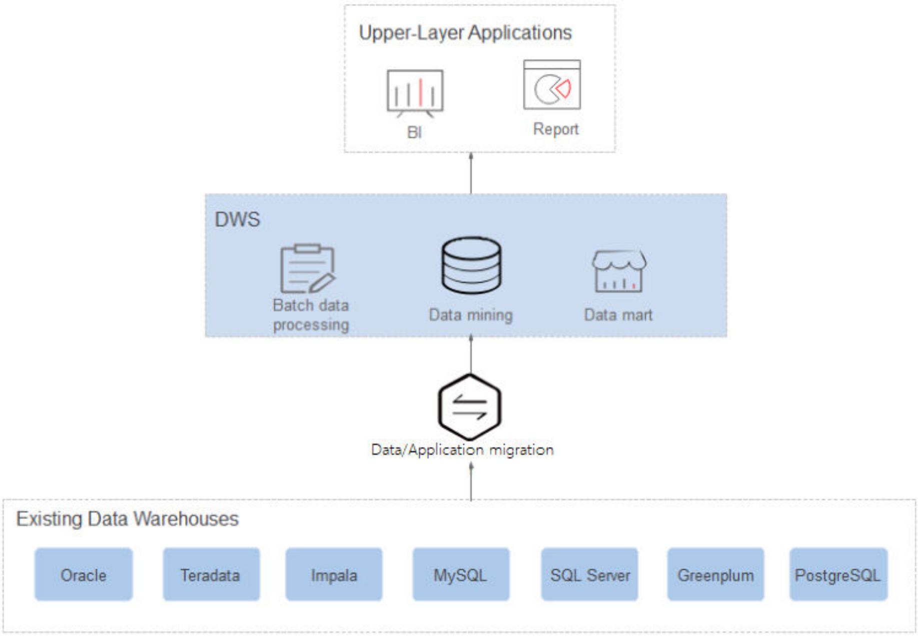
- Pay-per-use: DWS is billed based on the usage and use duration. You only need to pay for the resources you use.
- Flexible investment in infrastructure: You do not need to invest much in infrastructure in the early stage. You can start from a data warehouse instance with low specifications and flexibly scale it up and down at any time.

5 Application Scenarios

Data Warehouse Migration

The data warehouse is an important data analysis system for enterprises. As the service volume grows, performance of their own data warehouses cannot meet the actual service requirements due to scalability limitation and high costs. As an enterprise-class data warehouse on the cloud, DWS features high performance, low cost, and easy scalability, satisfying requirements in the big data era.

Figure 5-1 Data warehouse migration



Advantages

- Seamless migration

  DWS provides tools for easy migration of widely used data analysis systems like Teradata, Oracle, MySQL, SQL Server, PostgreSQL, Greenplum, and Impala.

- Compatible with conventional data warehouses

DWS supports the SQL 2003 standard and stored procedures. It is compatible with some Oracle syntax and data structures, and can be seamlessly interconnected with typical BI tools, saving service migration efforts.

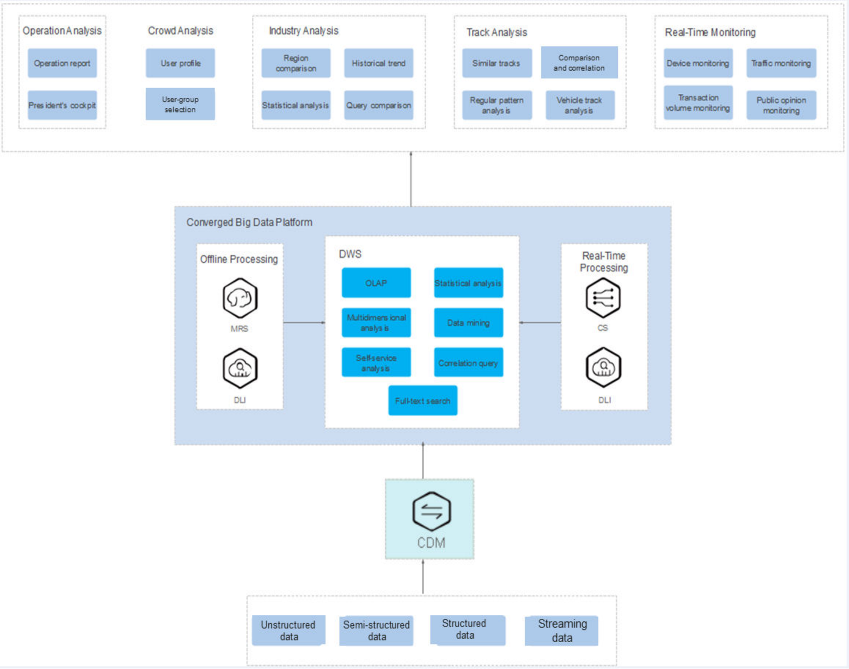
- Secure and reliable

  DWS supports data encryption and connects to DBSS to ensure data security on the cloud. In addition, DWS supports automatic full and incremental backup of data, improving data reliability.

Converged Big Data Analysis

Data has become the most important asset. Enterprises must be able to integrate their data resources and build big data platforms to mine the full value of their data. In predictive analysis use cases, massive volumes of data must be processed. DWS delivers the needed processing power to handle these intense compute scenarios.

Figure 5-2 Converged big data analysis



Advantages

- Unified Analysis Entrance

  DWS SQL acts as a unified entry point for upper-layer applications, enabling developers to access all data using SQL.

- Real-Time Interactive Analysis

  Analysis personnel can obtain immediately actionable information from the big data platform in real time.

- Auto Scaling

  Adding nodes allows you to easily expand into PB-range capacity while enhancing query and analysis performance of the system.

Enhanced ETL + Real-Time BI Analytics

The data warehouse is the pillar of the BI system for collecting, storing, and analyzing massive volumes of data. It powers business decision analysis for the finance, education, mobile Internet, and Online to Offline (O2O) industries.

Advantages

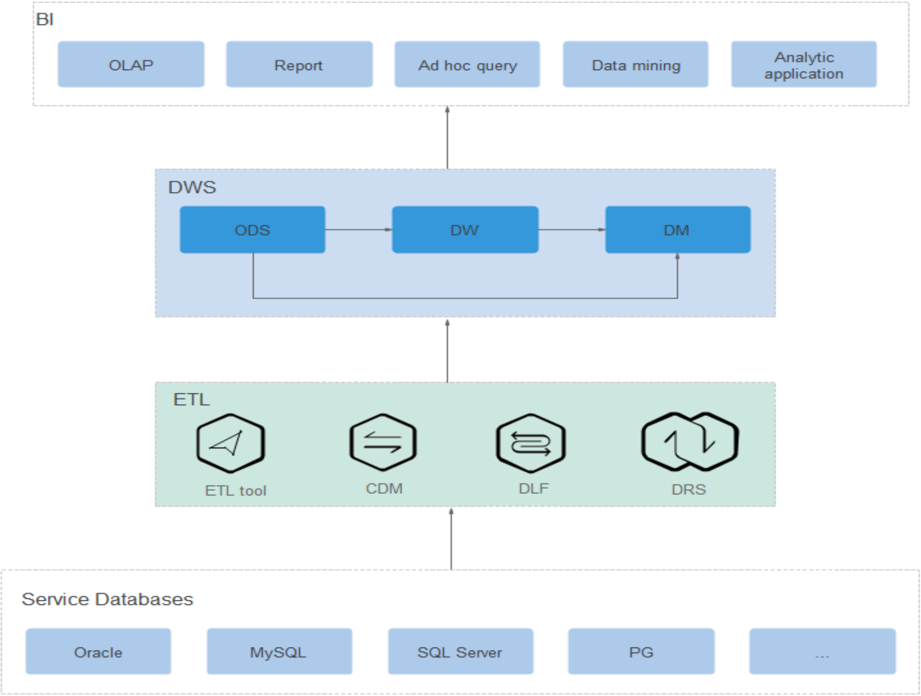
- Data Migration

  Ability to import data in batches in real time from multiple data sources.
- High Performance

  Cost-effective PB-level data storage and response to correlation analysis of trillions of data records within seconds.
- Real-Time

  Real-time consolidation of service data to produce actionable insights in operational decision-making.

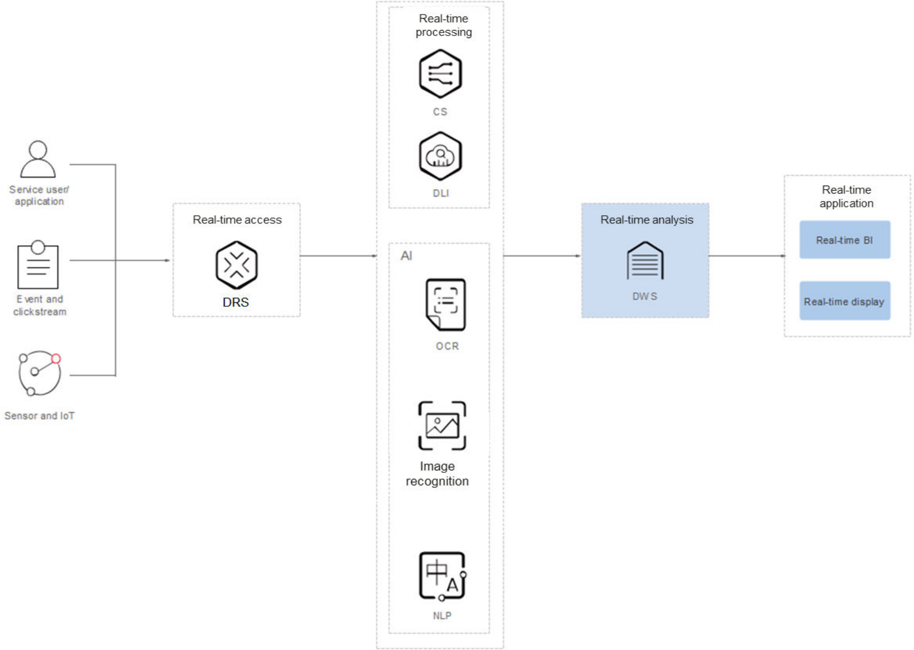
Figure 5-3 Enhanced ETL + real-time BI analysis



Real-Time Data Analytics

In the mobile Internet domain, huge volumes of data must be processed and analyzed in real time to extract the full value from data. The quick data import and query capabilities of DWS accelerate data analysis and enable real-time ingestion, processing, and value generation.

Figure 5-4 Real-time data analysis



Advantages

- Real-Time Import of Streaming Data

  Data from Internet applications can be written into DWS in real time after being processed by the stream computing and AI services.

- Real-Time Monitoring and Prediction

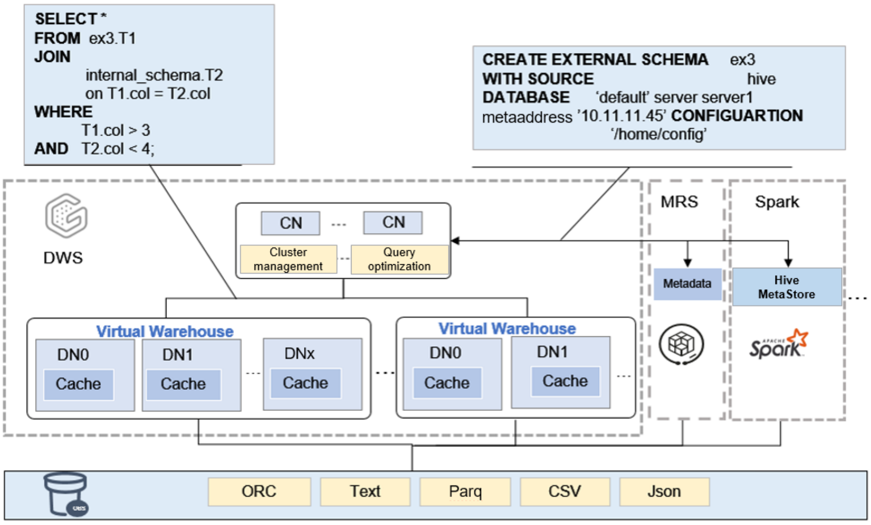
  Device monitoring, control, optimization, supply, self-diagnosis, and self-healing based on data analysis and prediction.

- Converged AI Analysis

  Correlation analysis can be conducted on results of image and text data analysis by AI services and other service data on DWS.

Lakehouse

- **Seamless access to the data lake**
  - With the interconnection with Hive Metastore metadata management, you can directly access the data table definitions in the data lake. You do not need to create a foreign table. You only need to create an external schema.
  - The following data formats are supported: ORC and Parquet.
- **Convergent query**
  - Hybrid query of any data in the data lake and warehouse is supported.
  - The query result is directly sent to the warehouse or data lake. No data needs to be transferred or copied.
- **Excellent query performance**
  - High-quality query plans and efficient execution engines
  - Precise load management methods



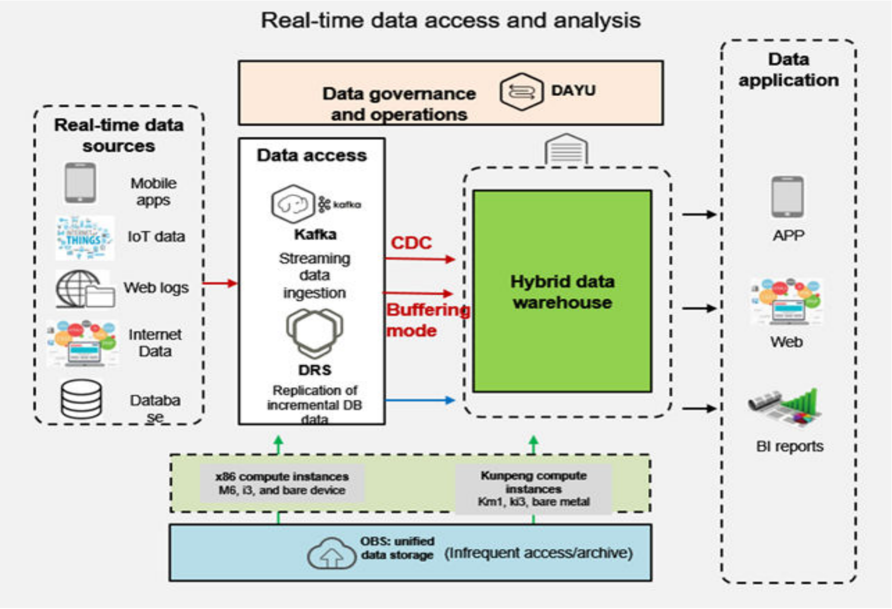
```
SELECT *
FROM  ex3.T1
JOIN
        internal_schema.T2
        on T1.col = T2.col
WHERE
        T1.col > 3
AND   T2.col < 4;
```

```
CREATE EXTERNAL SCHEMA    ex3
WITH SOURCE                  hive
DATABASE    'default' server server1
metaaddress '10.11.11.45' CONFIGUARTION
                  '/home/config'
```

Real-Time Write

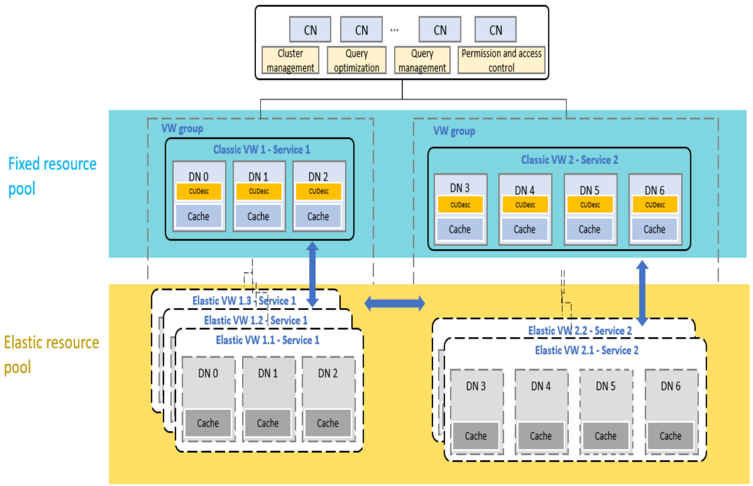
DWS 3.0 utilizes the HStore storage engine to store micro-batch data locally and syncs it to OBS at regular intervals. It enables high-throughput real-time write and update, as well as large-scale data writes.

Real-time data is written and calculated, and can be used for dashboard statistics, analysis, monitoring, risk control, and recommendations.



Service Isolation and Ultimate Elasticity with Multiple Virtual Warehouses (Storage-Compute Decoupling)

- Virtual warehouses (VWs) isolate service loads more effectively than soft isolation methods, using VM-level hard isolation to minimize service impact.
- Multiple classic VWs and multiple elastic VWs are supported.



- Classic VWs are used to isolate services.
  - VWs can be deployed based on service needs, with different services bound to different VWs. Classic VWs allow table creation.
  - Resources are isolated between VWs so that services do not affect each other.
  - Data is shared between VWs in real time.
  - The performance ceiling for a single SQL statement within our MPP architecture is determined by the size of a fixed VW.
  - Fixed VWs are optimized for consistent workloads and low-latency operations, such as real-time data access and processing. The size of fixed VWs can be proactively planned to accommodate anticipated service fluctuations.
- Concurrent expansion through elastic VW
  - In high-concurrency scenarios, elastic VWs are dynamically created to handle queued services. These VWs support read and write operations, but not table creation.
  - Elastic VWs automatically handle queuing queries.
  - Elastic VWs seamlessly absorbs queued queries to enhance service concurrency.
  - As demand subsides, elastic VWs are automatically decommissioned.
  - Elastic VWs offer on-demand resource allocation, with the flexibility for users to define upper limits.
  - Despite their dynamic nature, elastic VWs maintain the same specifications as fixed VWs, ensuring consistent SQL statement performance.
  - Elastic VWs adopt a usage-based billing.
  - Elastic VWs are suitable for handling sporadic and cyclical workloads.

For example, if a customer has multiple service departments, each can be assigned a classic VW to isolate resources. If Service 1 uses a three-node VW and Service 2 uses a four-node VW, and Service 1 has peak hours from 10:00 to 12:00, elastic VWs can be configured to scale during peak hours and be destroyed afterward.

6 Features

DWS provides various methods to access the service, such as the console, client, and REST APIs. This section describes the main functions of DWS.

Enterprise-Level Data Warehouses and Compatibility with Standard SQL

After a DWS cluster is created, you can use the SQL client to connect to the cluster and perform operations such as creating a database, managing the database, importing and exporting data, and querying data.

DWS provides high-performance databases that can handle petabytes of data, with the following features:

- MPP computing framework, hybrid row-column storage, and vectorized execution, enabling response to billion-level data correlation analysis within seconds
- Optimized in-memory computing based on Hash Join of Bloom Filter, improving the performance by 2 to 10 times
- Supports the symmetrically distributed, active-active multi-node cluster architecture, ensuring no SPOFs.
- Optimized communication between large-scale clusters based on telecommunication technologies, improving data transmission efficiency between compute nodes
- Cost-based intelligent optimizers, helping generate the optimal plan based on the cluster scale and data volume to improve execution efficiency

DWS has comprehensive SQL capabilities:

- Supports ANSI/ISO SQL 92, SQL99, and SQL 2003 standards, stored procedures, GBK and UTF-8 character sets, and SQL standard functions and OLAP analysis functions.
- Compatible with the PostgreSQL/Oracle/Teradata/MySQL ecosystem and supports interconnection with mainstream database ETL and BI tools provided by third-party vendors.
- Supports roaring bitmaps and common functions used with them, which are widely used for user feature extraction, user profiling, and more applications in the Internet, retail, education, and gaming industries.

- List partitioning (**PARTITION BY LIST** *(partition_key,[...])*) and range partitioning are supported.
- Read-only HDFS and OBS foreign tables in JSON file format are supported.
- Permissions on system catalogs can be granted to common users. The **VACUUM** permission can be granted separately. Roles with predefined, extensible permissions are supported, including:
  - **ALTER**, **DROP**, and **VACUUM** permissions at the table level.
  - **ALTER** and **DROP** permissions at the schema level.
  - Preset roles **role_signal_backend** and **role_read_all_stats**

For details about the SQL syntax and database operation guidance, see the ***Data Warehouse Service Developer Guide***.

Cluster Management

A DWS cluster contains nodes of the same flavor in the same subnet. These nodes jointly provide services. DWS offers a professional, efficient, and centralized management console that enables you to quickly request clusters, manage data warehouses with ease, and concentrate on data and services.

Main functions of cluster management are described as follows:

- Creating Clusters

  To use data warehouse services on the cloud, create a DWS cluster first. You can select product and node specifications to quickly create a cluster. You can also purchase a yearly/monthly package to create a cluster.

- Managing Snapshots

  A snapshot is a complete backup that records point-in-time configuration data and service data of a DWS cluster. A snapshot can be used to restore a cluster at a certain time. You can manually create snapshots for a cluster or enable automated snapshot creation (periodic). Automated snapshots have a limited retention period. You can copy automatic snapshots for long-term retention.

  When you restore a cluster from a snapshot, the system can restore the snapshot data to a new cluster or the original cluster.

  You can delete snapshots that are no longer needed on the console to release storage space. Automated snapshots cannot be manually deleted.

- Managing nodes

  You can check the nodes in a cluster, including the status, specifications, and usage of each node. To prepare for a large scale-out, you can add nodes in batches. To add 180 nodes, add them in three batches of 60 nodes each. If any nodes fail to be added, retry adding them. Once all 180 nodes are added, use them for scaling out. Adding nodes will not interrupt cluster services.

- Scaling out clusters

  As the service volume increases, the current scale of a cluster may not meet service requirements. In this case, you can scale out the cluster by adding compute nodes to it. Services are not interrupted during the scale-out. You can enable online scale-out and automatic redistribution if necessary.

- Managing redistribution

By default, redistribution is automatically started after cluster scale-out. For enhanced reliability, disable the automatic redistribution function and manually start a redistribution task after the scale-out is successful. Data redistribution can accelerate service response. Currently, DWS supports offline redistribution (default mode) and online redistribution.

- Storage space scaling

  As customer services evolve, disk space often becomes the initial bottleneck. In scenarios where other resources are ample, the conventional scale-out process is not only time-consuming but also resource-inefficient. Disk capacity expansion can quickly increase storage without service interruption. You can expand the disk capacity when no other services are running. If the disk space is insufficient after the expansion, you can continue to expand the disk capacity. If the expansion fails, you can expand the disk capacity again.

- Resource management

  When multiple database users query jobs at the same time, some complex queries may occupy cluster resources for a long time, affecting the performance of other queries. For example, a group of database users continuously submit complex and time-consuming queries, while another group of users frequently submit short queries. In this case, short queries may have to wait in the queue for the time-consuming queries to complete. To improve efficiency, you can use the DWS resource management function to handle such problems. You can create different resource pools for different types of services, and configure different resource ratios for these pools. Then, add database users to the corresponding pools to restrict their resource usages.

- Logical cluster

  A physical cluster can be divided into logical clusters that use the node-group mechanism. Tables in a database can be allocated to different physical nodes by logical cluster. A logical cluster can contain tables from multiple databases.

- Restarting clusters

  Restarting a cluster may cause data loss in running services. If you have to restart a cluster, ensure that there is no running service and all data has been saved.

- Deleting Clusters

  You can delete a cluster when you do not need it. Deleting a cluster is risky and may cause data loss. Therefore, exercise caution when performing this operation.

DWS allows you to manage clusters in either of the following ways:

- Management console

  Use the management console to access DWS clusters. When you have registered an account, log in to the management console and choose **Data Warehouse Service**.

  For more information about cluster management, see **Managing Clusters**.

- REST APIs

  Use REST APIs provided by DWS to manage clusters. In addition, if you need to integrate DWS into a third-party system for secondary development, use APIs to access the service.

For details, see **Data Warehouse Service API Reference**.

Diverse Data Import Modes

DWS supports efficient data import from multiple data sources. The following lists typical data import modes. For details, see **Data Migration to DWS**.

- Importing data from OBS in parallel
- Using GDS to import data from a remote server
- Importing data from MRS to a data warehouse cluster
- Importing data from one DWS cluster to another
- Using the gsql meta-command **\COPY** to import data
- Running the **COPY FROM STDIN** statement to import data
- Using DLI to import data to DWS
- Migrating data to DWS using CDM
- Using Database Schema Convertor (DSC) to migrate SQL scripts
- Using **gs_dump** and **gs_dumpall** to export metadata
- Using **gs_restore** to import data

APIs

You can call standard APIs, such as JDBC and ODBC, to access databases in DWS clusters.

For details, see **Using JDBC to Connect to a Cluster** and **Using ODBC to Connect to a Cluster**.

High Reliability

- Supports instance and data redundancy, ensuring zero single points of failure (SPOF) in the entire system.
- Supports multiple data backups, and all data can be manually backed up to OBS.
- Automatically isolates the faulty node, uses the backup to restore data, and replaces the faulty node when necessary.
- Automatic snapshots work with OBS to implement intra-region disaster recovery (DR). If the production cluster fails to provide read and write services due to natural disasters in the specified region or cluster internal faults, the DR cluster becomes the production cluster to ensure service continuity.
- In the **Unbalanced** state, the number of primary instances on some nodes increases. As a result, the load pressure is high. In this case, you can perform a primary/standby switchback for the cluster during off-peak hours to improve performance.
- If the internal IP address or EIP of a CN is used to connect to a cluster, the failure of this CN will lead to cluster connection failure. To avoid single-CN failures, DWS uses Elastic Load Balance (ELB). An ELB distributes access traffic to multiple ECSs for traffic control based on forwarding policies. It improves the fault tolerance capability of application programs.
- After a cluster is created, the number of required CNs varies with service requirements. DWS allows you to add or delete CNs as needed.

Security Management

- Isolates tenants and controls access permissions to protect the privacy and data security of systems and users based on the network isolation and security group rules, as well as security hardening measures.
- Supports SSL network connections, user permission management, and password management, ensuring data security at the network, management, application, and system layers.

  For details, see **"Establishing Secure TCP/IP Connections in SSL Mode"** and **"Enabling Separation of Duties for DWS Database Users"**.

Monitoring and Auditing

- Monitoring Clusters

  DWS integrates with Cloud Eye, allowing you to monitor compute nodes and databases in the cluster in real time. For details, see **Monitoring Clusters Using Cloud Eye**.
- Database Monitoring

  DMS is provided by DWS to ensure the fast and stable running of databases. It collects, monitors, and analyzes the disk, network, and OS metric data used by the service database, as well as key performance metric data of cluster running. It also diagnoses database hosts, instances, and service SQL statements based on the collected metrics to expose key faults and performance problems in a database in a timely manner, and guides customers to optimize and resolve the problems. For details, see **Database Monitoring (DMS)**.
- Alarms

  You can check and configure alarm rules and subscribe to alarm notifications. Alarm rules display alarm statistics and details of the past week for users to view tenant alarms. This feature monitors common DWS alarms with pre-set rules and allows users to customize the alarm thresholds based on their service needs. For details, see **Alarms**.
- Notifying Events

  DWS interconnects with Simple Message Notification (SMN) so that you can subscribe to events and view events that are triggered. For details, see **Event Notifications**.
- Audit Logs
  - DWS can be integrated with Cloud Trace Service (CTS) to audit management console operations and API calls. For details, see **Viewing Audit Logs of Key Operations on the Management Console**.
  - DWS records all SQL operations, including connection attempts, query attempts, and database changes. For details, see **Viewing Database Audit Logs**.

Multiple Database Tools

DWS provides the following self-developed tools. You can download the tool packages on the DWS console. For how to use the tools, see the ***Data Warehouse Service (DWS) Tool Guide***.

- SQL editor

  The DWS SQL editor provides one-stop data development, ingestion, and processing functions. With the editor, you can connect to a cluster database from the DWS console to edit and execute SQL statements.

- gsql

  gsql is a CLI SQL client tool running on the Linux OS. It helps connect to, operate, and maintain the database in a DWS cluster.

- Data Studio

  Data Studio is a SQL client tool with a Graphical User Interface (GUI) that runs on Windows. It is utilized to connect to databases in a DWS cluster, manage database objects, edit, run, and debug SQL scripts, and view execution plans.

- GDS

  GDS is a data service tool offered by DWS that utilizes the foreign table mechanism to achieve fast data import and export.

  The GDS tool package needs to be installed on the server where the data source file is located. This server is called the data server or the GDS server.

- DSC SQL syntax migration tool

  The DSC is a CLI tool running on the Linux or Windows OS. It is dedicated to providing customers with simple, fast, and reliable application SQL script migration services. It parses the SQL scripts of source database applications using the built-in syntax migration logic, and converts them to SQL scripts applicable to DWS databases.

  The DSC can migrate SQL scripts of Teradata, Oracle, Netezza, MySQL, and DB2 databases.

- **gs_dump** and **gs_dumpall**

  gs_dump exports a single database or its objects. **gs_dumpall** exports all databases or global objects in a cluster.

  To migrate database information, you can use a tool to import the exported metadata to a target database.

- gs_restore

  During database migration, you can export files using **gs_dump tool** and import them to DWS by using **gs_restore**. In this way, metadata, such as table definitions and database object definitions, can be imported.

7 Concepts

DWS Management Concepts

- Cluster

  A cluster is a server group that consists of multiple nodes. DWS is organized using clusters. A data warehouse cluster contains nodes with the same flavor in the same subnet. These nodes work together to provide services.

- Node

  A DWS cluster can contain 3 to 256 nodes. Each node can store and analyze data. For details, see **DWS Technical Specifications**.

- Type

  You need to specify the node flavors when you create a DWS cluster. CPU, memory, and storage resources vary depending on node flavors.

- Snapshot

  You can create snapshots to back up DWS cluster data. A snapshot is retained until you delete it on the management console. Automated snapshots cannot be manually deleted. Snapshots will occupy your OBS quotas.

- Project

  Projects are used to group and isolate OpenStack resources (computing resources, storage resources, and network resources). A project can be a department or a project team. Multiple projects can be created for one account.

DWS Database Concepts

- Database

  A database manages data objects and is isolated from other databases. While creating an object, you can specify a tablespace for it. If you do not specify it, the object will be saved to the **PG_DEFAULT** space by default. Objects managed by a database can be distributed to multiple tablespaces.

- OLAP

  OLAP is a major function of DWS clusters. It supports complex analysis, provides decision-making support tailored to analysis results, and delivers intuitive query results.

- MPP

On each node in the data warehouse cluster, memory computing and disk storage systems are independent from each other. With MPP, DWS distributes service data to different nodes based on the database model and application characteristics. Nodes are connected through the network and collaboratively process computing tasks as a cluster and provide database services that meet service needs.

- Shared-nothing architecture

  The shared-nothing architecture is a distributed computing architecture. Each node is independent so that nodes do not compete for resources, which improves work efficiency.

- Database version

  Each data warehouse cluster has a specific database version. You can check the version when creating a data warehouse cluster.

- Database connections

  You can use a client to connect to the DWS cluster. The client can be used for connection on the Huawei Cloud platform and over the Internet.

- Database users and roles

  DWS uses users and roles to control the access to databases. A role can be a database user or a group of database users based on the role setting. In DWS, the difference between roles and users is that a role does not have the **LOGIN** permission by default. In DWS, one user can have only one role, but you can put a user's role under a parent role to grant multiple permissions to the user.

- Instance

  In DWS, instances are a group of database processes running in the memory. An instance can manage one or more databases that form a cluster. A cluster is an area in the storage disk. This area is initialized during installation and composed of a directory. The directory, called data directory, stores all data and is created by **initdb**. Theoretically, one server can start multiple instances on different ports, but DWS manages only one instance at a time. The start and stop of an instance rely on the specific data directory. For compatibility purposes, the concept of instance name may be introduced.

- Tablespace

  In DWS, a tablespace is a directory storing physical files of the databases the tablespace contains. Multiple tablespaces can coexist. Files are physically isolated using tablespaces and managed by a file system.

- Schema

  DWS schemas logically separate databases. All database objects are created under certain schemas. In DWS, schemas and users are loosely bound. When you create a user, a schema with the same name as the user will be created automatically. You can also create a schema or specify another schema.

- V2 table

  A V2 table refers to a table whose **colversion** defined in the **CREATE TABLE** syntax is 2.0 during table creation, indicating that each column of the column-store table is combined and stored in a file named **relfilenode.C1.0**, and data is stored on the local disk. For storage-compute coupled clusters, if **colversion** is not specified, the created column-store table is a V2 table by default.

- V3 table

  A V3 table refers to a table whose **colversion** defined in the **CREATE TABLE** syntax is 3.0 during table creation, indicating that each column of the column-store table is stored in a file named **C1_field.0**, and data is stored in the OBS file system. For storage-compute coupled clusters, if **colversion** is not specified, the created column-store table is a V3 table by default.

- Transaction management

  In DWS, transactions are managed by multi-version concurrency control (MVCC) and two-phase locking (2PL). It enables smooth data reads and writes. In DWS, MVCC saves historical version data together with the current tuple version. DWS uses the VACUUM process instead of rollback segments to routinely delete historical version data. This does not affect user operations, unless in performance tuning. Transactions are automatically submitted in DWS.

- Node Group

  A node group is a basic concept of the database kernel and contains multiple cluster nodes. A physical cluster contains multiple node groups, and each node group is a logical cluster that can be accessed by users.

- Logical Cluster

  In the storage-compute coupled architecture, each physical cluster is divided into multiple node groups, and each node group is a logical cluster.

- Virtual Warehouse

  A virtual warehouse (VW) is a logical cluster in the storage-compute decoupled architecture.

  In a storage-compute coupled cluster, a node group is called a logical cluster. In a storage-compute decoupled cluster, a node group is called a VW.

8 Related Services

IAM

DWS uses Identity and Access Management (IAM) for authentication and authorization.

Users who have the **DWS Administrator** permissions can fully utilize DWS. To obtain the permissions, contact a user with the **Security Administrator** permissions or directly create a user with the **DWS Administrator** permissions. Users granted the **DWS Database Access** permissions can generate temporary database user credentials based on IAM users to connect to databases in the data warehouse clusters.

ECS

DWS uses an ECS as a cluster node.

BMS

DWS uses a BMS as a cluster node.

VPC

DWS uses the Virtual Private Cloud (VPC) service to provide a network topology for clusters to isolate clusters and control access.

OBS

DWS uses OBS to convert cluster data and external data, satisfying the requirements for secure, reliable, and cost-effective storage.

MRS

Data can be migrated from MRS to DWS clusters for analysis after the data is processed by Hadoop.

CDM

You can use Cloud Data Migration (CDM) to migrate data from multiple sources to DWS.

Cloud Eye

DWS uses Cloud Eye to monitor cluster performance metrics, delivering status information in a concise and efficient manner. Cloud Eye supports alarm customization so that you are notified of the exception instantly.

CTS

DWS uses Cloud Trace Service (CTS) to audit your non-query operations on the management console to ensure that no invalid or unauthorized operations are performed, enhancing service security management.

LTS

DWS users can view collected cluster logs or dump logs on the Log Tank Service (LTS) console.

SMN

DWS uses SMN to actively push notification messages according to your event subscription requirements, so that you can immediately receive a notification when an event occurs (for example, a key cluster operation).

TMS

With Tag Management Service (TMS), DWS can provide centralized tag management and resource classification functions across regions and services. You can customize tags to classify and locate resources.

DNS

DWS uses Domain Name Service (DNS) to provide the cluster IP addresses mapped from domain names.

ELB

With Elastic Load Balance (ELB) health checks, the CN requests of a cluster can be quickly forwarded to normal CNs. If a CN is faulty, the workload can be immediately shifted to a healthy node, minimizing cluster access faults.

9 Security

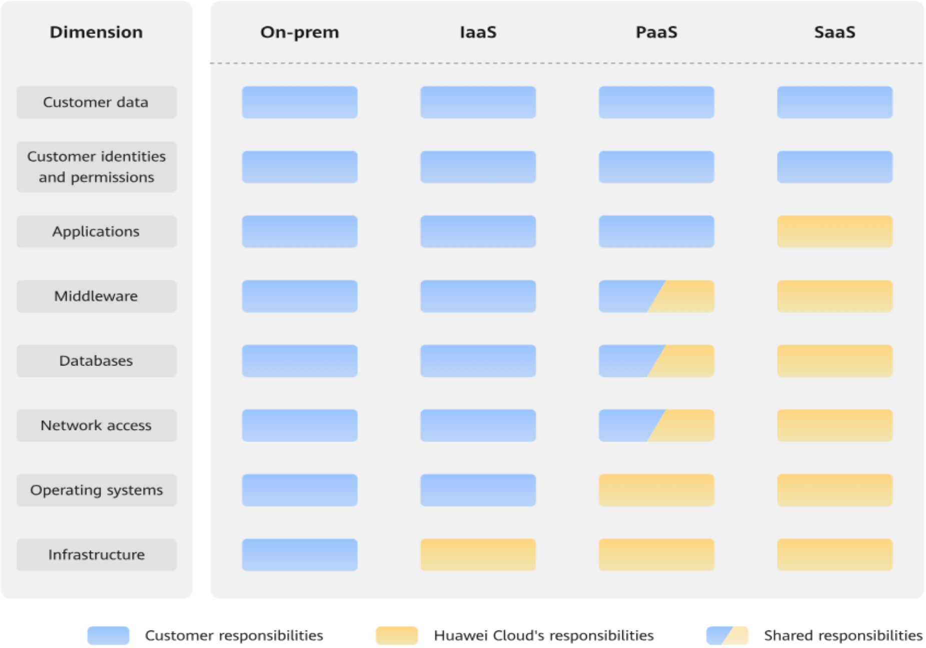
9.1 Shared Responsibilities

Huawei guarantees that its commitment to cyber security will never be outweighed by the consideration of commercial interests. To cope with emerging cloud security challenges and pervasive cloud security threats and attacks, Huawei Cloud builds a comprehensive cloud service security assurance system for different regions and industries based on Huawei's unique software and hardware advantages, laws, regulations, industry standards, and security ecosystem.

Unlike traditional on-premises data centers, cloud computing separates operators from users. This approach not only enhances flexibility and control for users but also greatly reduces their operational workload. For this reason, cloud security cannot be fully ensured by one party. Cloud security requires joint efforts of Huawei Cloud and you, as shown in **Figure 9-1**.

- **Huawei Cloud**: Huawei Cloud is responsible for infrastructure security, including security and compliance, regardless of cloud service categories. The infrastructure consists of physical data centers, which house compute, storage, and network resources, virtualization platforms, and cloud services Huawei Cloud provides for you. In PaaS and SaaS scenarios, Huawei Cloud is responsible for security settings, vulnerability remediation, security controls, and detecting any intrusions into the network where your services or Huawei Cloud components are deployed.
- **Customer**: As our customer, your ownership of and control over your data assets will not be transferred under any cloud service category. Without your explicit authorization, Huawei Cloud will not use or monetize your data, but you are responsible for protecting your data and managing identities and access. This includes ensuring the legal compliance of your data on the cloud, using secure credentials (such as strong passwords and multi-factor authentication), and properly managing those credentials, as well as monitoring and managing content security, looking out for abnormal account behavior, and responding to it, when discovered, in a timely manner.

Figure 9-1 Huawei Cloud shared security responsibility model



Dimension	On-prem	IaaS	PaaS	SaaS
Customer data	🟦	🟦	🟦	🟦
Customer identities and permissions	🟦	🟦	🟦	🟦
Applications	🟦	🟦	🟦	🟨
Middleware	🟦	🟦	◩	🟨
Databases	🟦	🟦	◩	🟨
Network access	🟦	🟦	◩	🟨
Operating systems	🟦	🟦	🟨	🟨
Infrastructure	🟦	🟨	🟨	🟨

🟦 Customer responsibilities 🟨 Huawei Cloud's responsibilities ◩ Shared responsibilities

Cloud security responsibilities are determined by control, visibility, and availability. When you migrate services to the cloud, assets, such as devices, hardware, software, media, VMs, OSs, and data, are controlled by both you and Huawei Cloud. This means that your responsibilities depend on the cloud services you select. As shown in **Figure 9-1**, customers can select different cloud service types (such as IaaS, PaaS, and SaaS services) based on their service requirements. As control over components varies across different cloud service categories, the responsibilities are shared differently.

- In on-premises scenarios, customers have full control over assets such as hardware, software, and data, so tenants are responsible for the security of all components.
- In IaaS scenarios, customers have control over all components except the underlying infrastructure. So, customers are responsible for securing these components. This includes ensuring the legal compliance of the applications, maintaining development and design security, and managing vulnerability remediation, configuration security, and security controls for related components such as middleware, databases, and operating systems.
- In PaaS scenarios, customers are responsible for the applications they deploy, as well as the security settings and policies of the middleware, database, and network access under their control.
- In SaaS scenarios, customers have control over their content, accounts, and permissions. They need to protect their content, and properly configure and protect their accounts and permissions in compliance with laws and regulations.

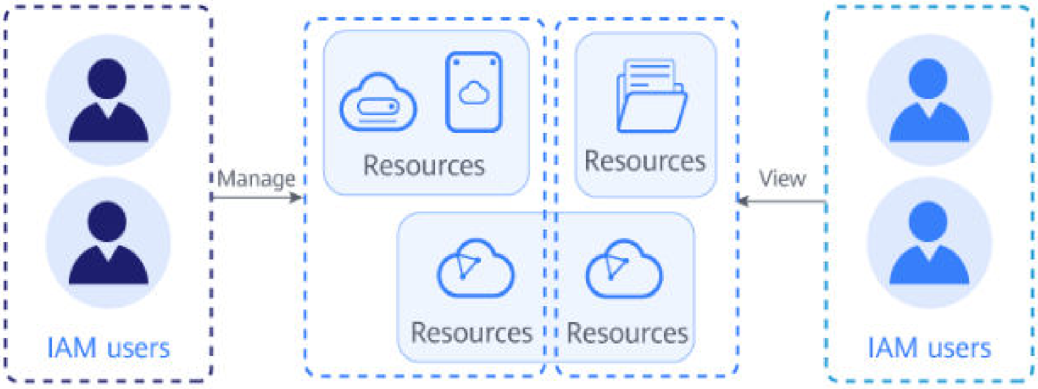
9.2 Authentication and Access Control

9.2.1 Resource Access Control (IAM Permission Control)

If you want to give varying levels of access to your company's DWS resources on Huawei Cloud, using IAM is an effective way to manage permissions in detail. IAM provides identity authentication, permissions management, and access control, helping you securely manage access to your Huawei Cloud resources. With IAM, you can use your Huawei Cloud account to create IAM users, and assign permissions to the users to control their access to specific resources.

- **Scenario 1**: To allow software developers in your company to use DWS resources while restricting high-risk operations and resource deletion, you can create IAM users tailored for these developers and grant them only the essential permissions for DWS usage.
- **Scenario 2**: Allow employees to use only DWS resources, but not the resources of other services. To this end, grant them only the permissions for DWS.

You can use IAM to control cloud resource access and prevents misoperations on cloud resources. For details, see **Creating a User and Granting DWS Permissions**.



9.2.2 Separation of Database Access Permissions

In DWS, you can isolate workloads through database and schema configurations. The differences are:

- Databases cannot communicate with each other and share very few resources. Their connections and permissions can be isolated.
- Schemas share more resources than databases do. User permissions on schemas and subordinate objects can be flexibly configured using the **GRANT** and **REVOKE** syntax.

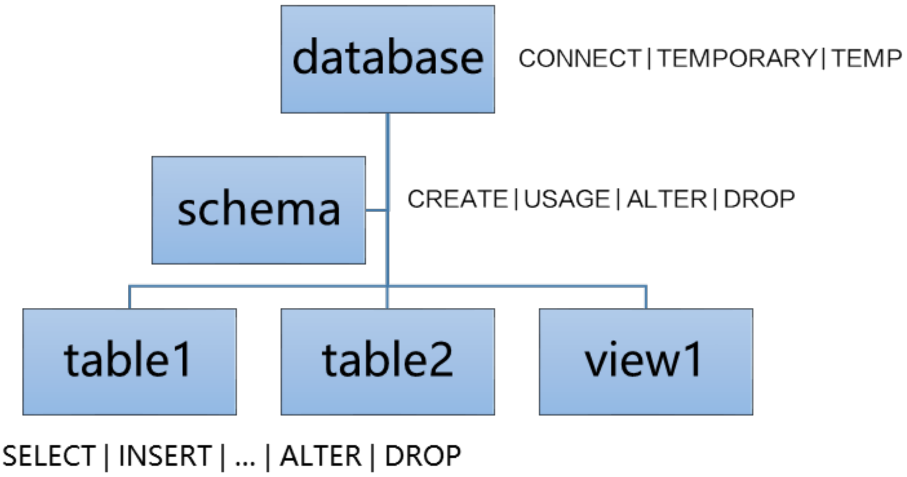
You are advised to use schemas to isolate services for convenience and resource sharing purposes. It is recommended that system administrators create schemas and databases and then assign required permissions to users.

1. Each database has one or more schemas. Each schema contains various types of objects, such as tables, views, and functions.
2. To access an object at the bottom layer, a user must be granted the permission on the object at the upper layer.
3. To create or delete a schema, you must have the **CREATE** permission for its database.

4. To access **table1** in a schema, a user must be granted the **CONNECT** permission for its **database**, the **USAGE** permission of the **schema**, and the **SELECT** permission of **table1**.

For details, see **How Does DWS Implement Workload Isolation?**

Figure 9-2 Permission levels


database
CONNECT | TEMPORARY | TEMP
schema
CREATE | USAGE | ALTER | DROP
table1
table2
view1
SELECT | INSERT | ... | ALTER | DROP

9.2.3 Permissions Management Using GRANT and REVOKE

Granting Permissions

DWS uses the **GRANT** syntax to grant permissions to roles and users. A common user cannot access a table without the permissions granted by the system administrator **dbadmin** or the table owner. This default mechanism controls user access to data and can prevent data leakage.

GRANT is used in the following scenarios:

- Granting **system permissions** to roles or users

  System permissions are also called user attributes, including **SYSADMIN**, **CREATEDB**, **CREATEROLE**, **AUDITADMIN**, and **LOGIN**.

  They can be specified only by the **CREATE ROLE** or **ALTER ROLE** syntax. The **SYSADMIN** permission can be granted and revoked using **GRANT ALL PRIVILEGE** and **REVOKE ALL PRIVILEGE**, respectively. System permissions cannot be inherited by a user from a role, and cannot be granted using **PUBLIC**.

- Granting **database object permissions** to roles or users

  Grant permissions for a database object (table, view, column, database, function, or schema) to a role or user.

  GRANT grants specified database object permissions to one or more roles. These permissions are appended to those already granted, if any.

  DWS grants the permissions on certain types of objects to **PUBLIC**. By default, permissions on tables, columns, sequences, foreign data sources, foreign servers, schemas, and tablespaces are not granted to **PUBLIC**, but the following permissions are granted to **PUBLIC**: **CONNECT** and **CREATE TEMP**

TABLE permissions on databases, **EXECUTE** permission on functions, and **USAGE** permission on languages and data types (including domains). An object owner can revoke the default permissions granted to **public** and grant permissions to other users. For security purposes, create an object and set its permissions in the same transaction, so that the object will not be accessible to any other users until you configure its permissions and end the transaction. In addition, you can run the **ALTER DEFAULT PRIVILEGES** statement to modify the default permissions.

- Granting **a role's or user's permissions** to other roles or users

  Grant a role's or user's permissions to one or more roles or users. In this case, every role or user can be regarded as a set of one or more database permissions.

  If **WITH ADMIN OPTION** is specified, the member can in turn grant permissions in the role to others, and revoke permissions in the role as well. If a role or user granted with certain permissions is changed or revoked, the permissions inherited from the role or user also change.

  A database administrator can grant permissions to and revoke them from any role or user. Roles having **CREATEROLE** permission can grant or revoke membership in any role that is not an administrator.

For more information, see **GRANT**.

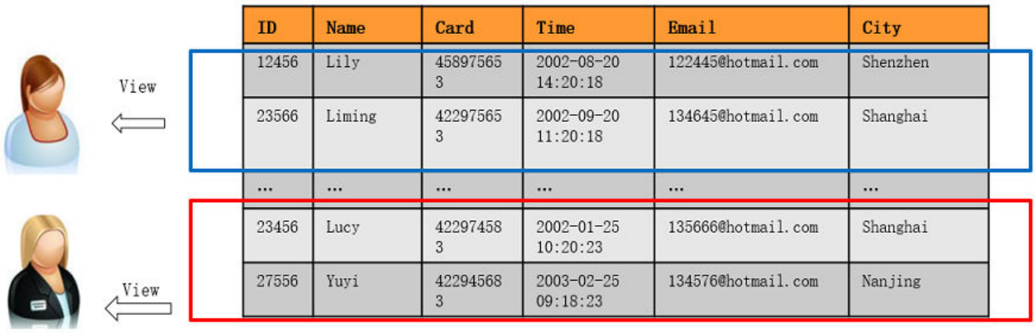
Revoking Permissions

After a user is granted with a database object permission, you can use the **REVOKE** syntax to revoke a permission from a user if the user no longer needs it, or if you need to control the user's permissions.

For more information, see **REVOKE**.

9.2.4 Row-Level Access Control

Multiple users may need to access and perform operations on the same table at the same time. In this case, you need to grant users the permissions for specific rows in the table. DWS can implement row-level access control. For example, a table administrator can see an entire table, but user A is allowed to view only specific rows in the table when they run **SELECT * FROM** *table_name*. This feature enables database access control to be accurate to each row of data tables. In this way, the same SQL query may return different results for different users.



ID	Name	Card	Time	Email	City
12456	Lily	458975653	2002-08-20 14:20:18	122445@hotmail.com	Shenzhen
23566	Liming	422975653	2002-09-20 11:20:18	134645@hotmail.com	Shanghai
...	...	...	...	...	...
23456	Lucy	422974583	2002-01-25 10:20:23	135666@hotmail.com	Shanghai
27556	Yuyi	422945683	2003-02-25 09:18:23	134576@hotmail.com	Nanjing

You can create a row-level access control policy for a data table. The policy defines an expression that takes effect only for specific database users and SQL operations. When a database user accesses the data table, if a SQL statement meets the specified row-level access control policies of the data table, the expressions that meet the specified condition will be combined by using **AND** or **OR** based on the attribute type (**PERMISSIVE** | **RESTRICTIVE**) and applied to the execution plan in the query optimization phase.

Row-level access control is used to control the visibility of row-level data in tables. By predefining filters for data tables, the expressions that meet the specified condition can be applied to execution plans in the query optimization phase, which will affect the final execution result. Currently, the SQL statements that can be affected include **SELECT**, **UPDATE**, and **DELETE**.

For details, see **Row-Level Access Control**.

9.3 Cyber Security

Configuring Database Account Security

Account description

When setting up a DWS cluster, the system will create the following accounts to provide comprehensive background operations and maintenance services for databases:

- **dbadmin**: system administrator account used for the initial login to the DWS database. It is responsible for creating service databases, managing common users, and assigning permissions.
- If separation of duties is enabled, additional accounts such as security administrator and audit administrator will be generated. The names of these accounts can be customized by the user. For details, see .
- **Ruby**: default O&M account, which cannot be used by O&M personnel of non-cloud service providers.
- **om_user**_*First eight digits of the cluster ID*: Other O&M accounts with preset permissions such as **gs_role_analyze_any**, **gs_role_vacuum_any**, **gs_role_read_all_stats**, and **gs_role_signal_backend**. These accounts are used for fault locating and cannot be used by O&M personnel of non-cloud service providers. For details, see **Authorizing a DWS Cluster O&M Account**.

Password complexity requirements

- The password of DWS system administrator **dbadmin** is set when the cluster is created. The DWS console verifies the password complexity. If the verification fails, the password cannot be set.

  You can learn more password complexity requirements in **Creating a DWS Storage-Compute Coupled Cluster**.
- Other common database users are also required to maintain a certain level of complexity in their passwords.
  - The password should contain 8 to 32 characters.
  - The password should contain at least three types of the following characters: uppercase letters, lowercase letters, digits, and special characters.

  - The password cannot be the same as the user name or the user name in reverse order, case insensitive.
  - The password cannot be the current password or the current password in reverse order.

NOTICE

To prevent your database password from being cracked, set a strong password and periodically change it.

User-defined password policy

You can modify default password policies. For details, see **Creating a Custom Password Policy for DWS**.

Configuring policies for account locking and password expiration

- Database user password validity: **password_effect_time** determines how long a database user password remains valid. When the password expires, the system prompts the user to change it. To modify this parameter, contact technical support.
- Maximum number of incorrect password attempts: The number of incorrect password attempts is controlled by the **failed_login_attempts** parameter. If the number of incorrect password attempts exceeds the specified value, the account is automatically locked. In this case, the system administrator needs to unlock the account. This parameter can be set on the DWS console. For details, see **Modifying GUC Parameters of the DWS Cluster**.
- Automatic account unlocking time: **password_lock_time** specifies the duration after which an account is automatically unlocked if it has been locked. To modify this parameter, contact technical support.

Resetting a password

- If the password of system administrator **dbadmin** is locked or forgotten, you can reset it using the DWS console.
- The password of a common user can be reset by system administrator **dbadmin** by running SQL commands in the background. The following command is an example. For details, see **Creating a Custom Password Policy for DWS**.

```
ALTER USER joe IDENTIFIED BY 'password';
```

Configuring Security Groups

A security group is a collection of access control rules for instances, such as cloud servers, containers, and databases, that have the same security requirements and that are mutually trusted within a VPC. You can define different access control rules for a security group, and these rules are then applied to all the instances added to this security group.

Each security group can have inbound and outbound rules to control the traffic to and from instances. For inbound rules, you need to specify the source, port, and protocol. For outbound rules, you need to specify the destination, port, and protocol.

Enabling the EIP function for your database may expose your EIP DNS and database port to potential hacking risks. To protect information such as your EIP, DNS, database port, database account, and password, you are advised to set the range of source IP addresses in the DWS security group to ensure that only trusted source IP addresses can access your DB instances.

When setting up a DWS cluster, you can choose to configure a security group or use the default one. The default security group allows only port **8000** by default. After creating the cluster, you can modify the security group rules or switch to a different one.

Unbinding an EIP from an External Link

EIP offers public IP addresses and bandwidth for Internet access, making external access easier but also increasing the risk of network-wide attacks. EIPs are susceptible to external DoS/DDoS attacks.

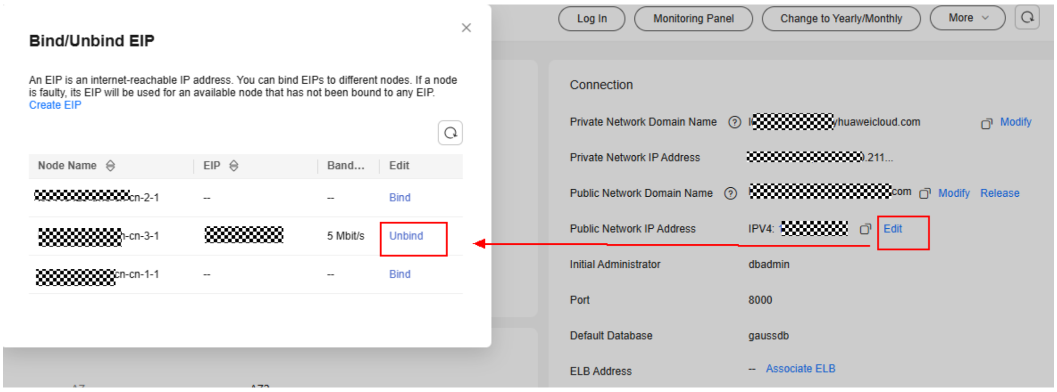
If you do not require public network access, it is best to consider the database as an internal component and use an internal IP address for accessing it. In this case, it is advised to detach the EIP from the database.

Unbinding procedure

Step 1 Log in to the DWS console and choose **Dedicated Clusters** > **Clusters**.

Step 2 In the cluster list, click the name of the target cluster. The cluster details page is displayed.

Step 3 Click **Edit** in the **Connection** area. On the displayed page, click **Unbind**.



Bind/Unbind EIP

An EIP is an internet-reachable IP address. You can bind EIPs to different nodes. If a node is faulty, its EIP will be used for an available node that has not been bound to any EIP.
Create EIP

Node Name	EIP	Band...	Edit
[redacted]cn-2-1	--	--	Bind
[redacted]-cn-3-1	[redacted]	5 Mbit/s	Unbind
[redacted]cn-cn-1-1	--	--	Bind

Connection

Private Network Domain Name [redacted]huaweicloud.com Modify
Private Network IP Address [redacted].211...
Public Network Domain Name [redacted]com Modify Release
Public Network IP Address IPV4: [redacted] Edit
Initial Administrator dbadmin
Port 8000
Default Database gaussdb
ELB Address -- Associate ELB

NOTE

To meet high availability requirements, if a cluster is bound to an ELB, you only need to unbind the EIP from the ELB. For details, see the ELB user guide.

----End

SSL-encrypted Data Transmission

DWS supports the standard SSL. As a highly secure protocol, SSL authenticates bidirectional identification between the server and client using digital signatures and digital certificates to ensure secure data transmission. To support SSL connection, DWS has obtained the formal certificates and keys for the server and client from the CA certification center. It is assumed that the key and certificate for the server are **server.key** and **server.crt** respectively; the key and certificate for the client are **client.key** and **client.crt** respectively, and the name of the CA root certificate is **cacert.pem**.

The SSL mode delivers higher security than the common mode. By default, the SSL function is enabled in a cluster to allow SSL or non-SSL connections from the client. For security purposes, you are advised to enable SSL connection. The server certificate, private key, and root certificate have been configured in DWS by default.

For details, see **Establishing Secure TCP/IP Connections in SSL Mode**.

Using DBSS (Recommended)

Database Security Service (DBSS) is an intelligent database security service. Based on the machine learning mechanism and big data analytics technologies, it can audit your databases, detect SQL injection attacks, and identify high-risk operations.

It is advised to use DBSS for enhanced data security capabilities. For details, see **Database Security Service**.

Advantages

- DBSS can help you meet security compliance requirements.
  - DBSS can help you comply with DJCP (graded protection) standards for database audit.
  - DBSS can help you comply with security laws and regulations, and provide compliance reports that meet data security standards (such as Sarbanes-Oxley).
- DBSS can back up and restore database audit logs and meet the audit data retention requirements.
- DBSS can monitor risks, sessions, session distribution, and SQL distribution in real time.
- DBSS can report alarms for risky behavior and attacks and respond to database attacks in real time.
- DBSS can locate internal violations and improper operations and keep data assets secure.

Deployed in bypass pattern, database audit can perform flexible audits on the database without affecting user services.

- Database audit monitors database logins, operation types (data definition, operation, and control), and operation objects based on risky operations to effectively audit the database.
- Database audit analyzes risks and sessions, and detects SQL injection attempts so you can stay apprised of your database status.
- Database audit provides a report template library to generate daily, weekly, or monthly audit reports according to your configurations. It sends real-time alarm notifications to help you obtain audit reports in a timely manner.

9.4 Data Protection Technologies

9.4.1 Data Backup

Data in DWS can be backed up and restored using snapshots. A snapshot is a full and incremental backup of a DWS cluster at a specific point in time. It records the current database data and cluster information, including the number of nodes, node specifications, and database administrator name. DWS allows you to manually create snapshots on the management console. It can also automatically create snapshots as scheduled to prevent data loss.

Currently, DWS supports backup and restoration based on OBS. For details, see **Snapshot Overview**.

9.4.2 Transparent Data Encryption

DWS supports transparent data encryption (TDE) to encrypt and decrypt data files in real time, protecting user data privacy.

Feature Description

Transparent Data Encryption (TDE) encrypts DWS data files. Generally, data security can be enhanced by threat mitigation measures, for example, design a secure system, encrypt confidential assets, or build a firewall around database servers. However, in a scenario where the physical media (for example, disks) are stolen by attackers or internal personnel, the malicious party can just restore or attach the database and browse the data. To avoid such problems, you can encrypt the sensitive data in the database and protect the keys that are used to encrypt the data. This prevents anyone without the keys from using the data, but this kind of protection must be planned in advance. DWS provides a comprehensive solution – TDE.

TDE performs real-time I/O encryption and decryption of the data. Users are unaware of the encryption. The encryption uses a database encryption key (DEK), which is not stored in the cluster. The DEK is a symmetric key secured by using the cluster encryption key (CEK) stored in a Key Management Service (KMS) server. Database servers store only DEK ciphertext. During database startup, the database connects to the KMS server, decrypts the DEK ciphertext to obtain the key plaintext, and caches the key plaintext in the memory. Once the server is powered off or the cluster is shut down, keys are deleted. Ensure you properly store the key files in the cluster, because they are irrecoverable.

Scenario

In a traditional database cluster, user data is stored in plaintext in column-store or row-store files. Malicious cluster maintenance personnel or attackers can bypass the database permission control mechanism in the OS or steal disks to access user data. DWS interconnects with the Key Management Service (KMS) of Data Encryption Workshop (DEW) on Huawei Cloud to implement transparent data encryption and ensure user data security.

In DWS database-level TDE, each DWS cluster has a CEK and is configured with a DEK. DEKs are encrypted using the CEKs and their ciphertext is stored in DWS. Keys are applied for, encrypted, and decrypted through the KMS service. The cryptographic algorithm is configured using configuration items. Currently, AES and SM4 algorithms are supported.

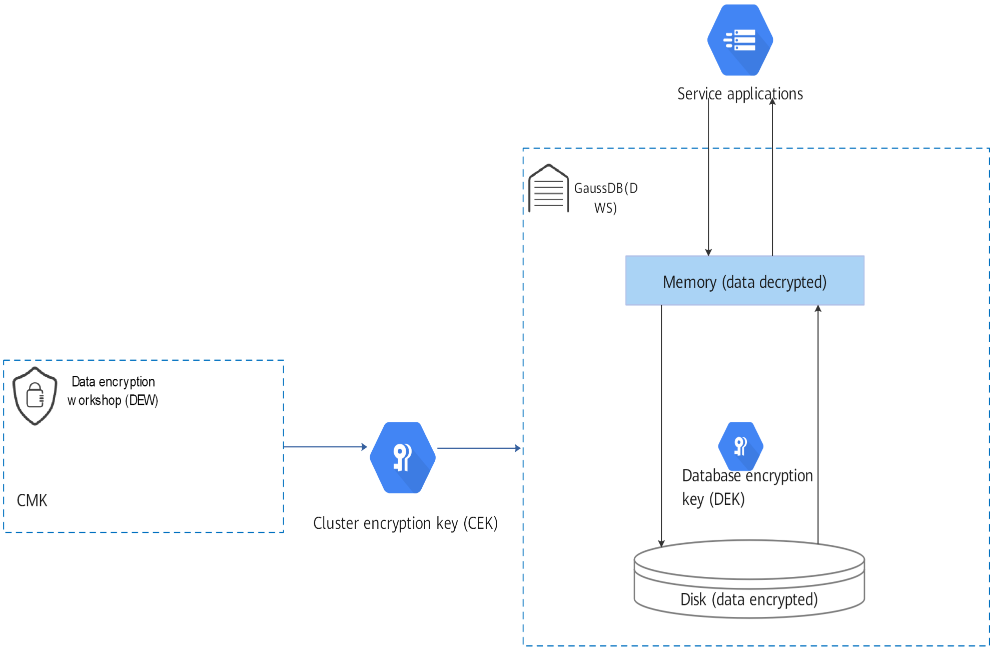
Currently, database-level transparent encryption is supported. You need to configure encryption when creating a cluster.

For details, see **Encryption Overview**.

TDE Encryption and Decryption Principles

With transparent encryption, all encryption and decryption operations are performed in the memory. Data in the memory is plaintext, and data in the disk is ciphertext. The database usage remains unchanged. Transparent encryption does not affect data processing and SQL execution.

Figure 9-3 TDE encryption and decryption principles


Service applications
GaussDB(DWS)
Memory (data decrypted)
Data encryption workshop (DEW)
CMK
Cluster encryption key (CEK)
Database encryption key (DEK)
Disk (data encrypted)

9.4.3 SSL-encrypted Data Transmission

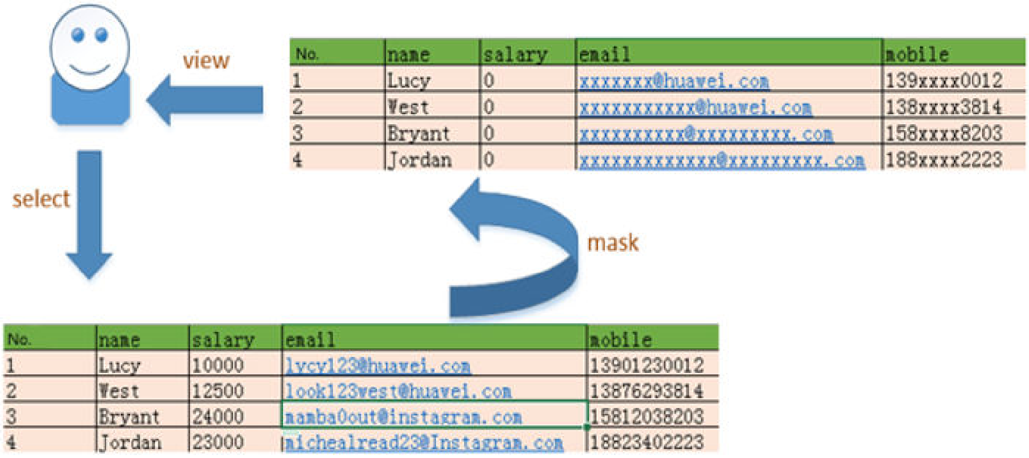
DWS supports the standard SSL. As a highly secure protocol, SSL authenticates bidirectional identification between the server and client using digital signatures and digital certificates to ensure secure data transmission. To support SSL connection, DWS has obtained the formal certificates and keys for the server and client from the CA certification center. It is assumed that the key and certificate for the server are **server.key** and **server.crt** respectively; the key and certificate for the client are **client.key** and **client.crt** respectively, and the name of the CA root certificate is **cacert.pem**.

The SSL mode delivers higher security than the common mode. By default, the SSL function is enabled in a cluster to allow SSL or non-SSL connections from the client. For security purposes, you are advised to enable SSL connection. The server certificate, private key, and root certificate have been configured in DWS by default.

For details, see **Establishing Secure TCP/IP Connections in SSL Mode**.

9.4.4 Data Redaction

DWS provides the column-level dynamic data masking function. For sensitive data, such as the ID card number, mobile number, and bank card number, the DDM function is used to redact the original data to protect data security and user privacy.



No.	name	salary	email	mobile
1	Lucy	0	xxxxxxx@huawei.com	139xxxx0012
2	West	0	xxxxxxxxxxx@huawei.com	138xxxx3814
3	Bryant	0	xxxxxxxxxx@xxxxxxxxx.com	158xxxx8203
4	Jordan	0	xxxxxxxxxxxxx@xxxxxxxxx.com	188xxxx2223

No.	name	salary	email	mobile
1	Lucy	10000	lvcy123@huawei.com	13901230012
2	West	12500	look123west@huawei.com	13876293814
3	Bryant	24000	mamba0out@instagram.com	15812038203
4	Jordan	23000	michealread23@Instagram.com	18823402223

For details, see **Data Redaction**.

9.4.5 Function-based Encryption

Data encryption is widely used in information systems to prevent unauthorized access and data leakage. As the core of an information system, the DWS data warehouse also provides transparent encryption and encryption using SQL functions.

DWS provides hash functions and symmetric cryptographic algorithms to encrypt and decrypt columns. Hash functions include sha256, sha384, sha512, and SM3. Symmetric cryptographic algorithms include AES128, AES192, AES256, and SM4.

- Hash functions
  - md5(string)

    Use MD5 to encrypt string and return a hexadecimal value. MD5 is insecure and is not recommended.
  - gs_hash(hashstr, hashmethod)

    Obtains the digest string of a **hashstr** string based on the algorithm specified by **hashmethod**. **hashmethod** can be **sha256**, **sha384**, **sha512**, or **sm3**.
- Symmetric cryptographic algorithms
  - gs_encrypt(encryptstr, keystr, cryptotype, cryptomode, hashmethod)

    Encrypts an **encryptstr** string using the **keystr** key based on the cryptographic algorithm specified by **cryptotype** and **cryptomode** and the HMAC algorithm specified by **hashmethod**, and returns the encrypted string.

- gs_decrypt(decryptstr, keystr, cryptotype, cryptomode, hashmethod)

  Decrypts a **decryptstr** string using the **keystr** key based on the cryptographic algorithm specified by **cryptotype** and **cryptomode** and the HMAC algorithm specified by **hashmethod**, and returns the decrypted string. The **keystr** used for decryption must be the same as that used for encryption.

- gs_encrypt_aes128(encryptstr,keystr)

  Encrypts **encryptstr** strings using **keystr** as the key and returns encrypted strings. The length of **keystr** ranges from 1 to 16 bytes.

- gs_decrypt_aes128(decryptstr,keystr)

  Decrypts a **decryptstr** string using the **keystr** key and returns the decrypted string. The **keystr** used for decryption must be the same as that used for encryption. **keystr** cannot be empty.

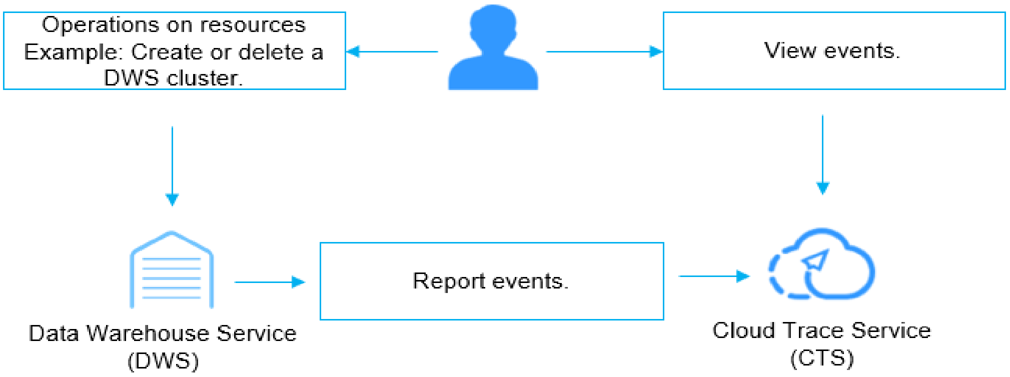
For details, see **Encrypting and Decrypting Data Columns**.

9.5 Audit and Logging

Operation Logs on the Management Console

Cloud Trace Service (CTS) records DWS operation logs on the management console, such as creating or deleting DWS clusters. CTS is a log audit service intended for cloud security. It records operations on the cloud resources in your account. You can use the logs generated by CTS to perform security analysis, track resource changes, audit compliance, and locate faults.

After you enable CTS and configure a tracker, CTS can record management and data traces of DWS for auditing.


Operations on resources
Example: Create or delete a DWS cluster.
View events.
Data Warehouse Service (DWS)
Report events.
Cloud Trace Service (CTS)

For details, see **Viewing Audit Logs of Key Operations on the Management Console**.

Database Audit Logs

In DWS, database logs include O&M logs (CN, DN, and OS logs) and DDL/DML database audit logs, which are audited by Log Tank Service (LTS). LTS collects log data from servers and cloud services. By processing massive amounts of logs efficiently, securely, and in real time, LTS provides useful insights for you to optimize the availability and performance of cloud services and applications. It also helps you efficiently perform real-time decision-making, device O&M management, and service trend analysis.

You can enable LTS to view the CN, DN, OS message, and DML/DDL audit logs of DWS.

In addition, database audit logs can be dumped to OBS to ensure the log retention period. For details, see **Viewing Database Audit Logs**.

9.6 Service Resilience

Security Hardening on the Management Plane

- Tomcat hardening: In the container images on the DWS management plane, the security of open source software like Tomcat is enhanced.
- JRE hardening:
  - Upgrade the HuaweiJre8 kernel version to 1.8.0_262 or later. Use the actual version number.
  - Configure the JRE path after the original **PATH** to avoid local unauthorized operations (**PATH=$PATH:$JAVA_HOME/bin**).
- System resource hardening: DWS has preset security parameters on underlying VMs to enhance the OS security of ECS and BMS.

Isolation Between the Database and External Networks

DWS is deployed in an independent VPC, which is isolated from other VPCs. Regarding firewall security zones, DWS resides in the internal user interface zone (trusted zone). Data transmission (using the CLI, GUI tool, and applications developed based on the client library) between clients and coordinator nodes is encrypted using SSL. Cluster nodes run in the secure internal network.

Database Cluster HA

Cluster high availability (HA) is a practice of write ahead logging (WAL), using mechanisms such as primary/standby data synchronization, switchover, and reconstruction for database instance recovery and self-healing. By doing this, data reliability and integrity, and more importantly, service continuity, can be maintained when a crash occurs in the database.

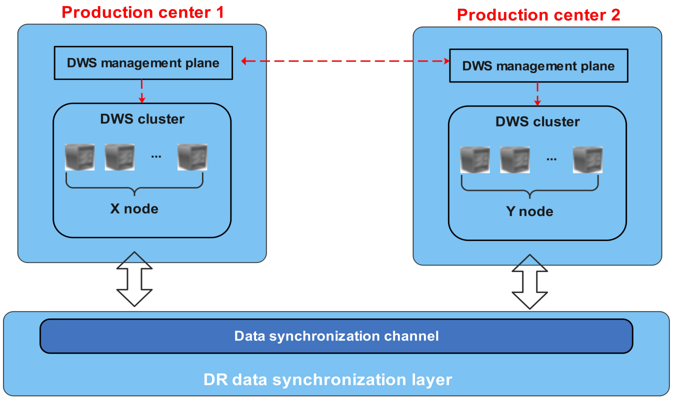
Intra-Region DR Deployment

DWS provides dual-cluster intra-region DR capabilities. A DWS production cluster and its homogeneous DR cluster can be deployed in different AZs within the same region. If the production cluster cannot provide read or write services due to a natural disaster or a fault, the DR cluster can serve as the production cluster to ensure service continuity.

The dual-cluster DR framework is based on Roach. It periodically synchronizes data between two clusters. This framework is flexible, enabling the two clusters to work either independently or together without affecting each other. RTO and RPO are within hours. In the non-recovery period, the standby cluster is in hot standby mode, able to provide read-only services.

For details, see **DR Overview**.

Figure 9-4 DR architecture



9.7 Risk Monitoring

DWS uses Cloud Eye to help you monitor your DWS clusters and receive alarms and notifications in real time. You can learn the metrics and health status of a DWS cluster in real time.

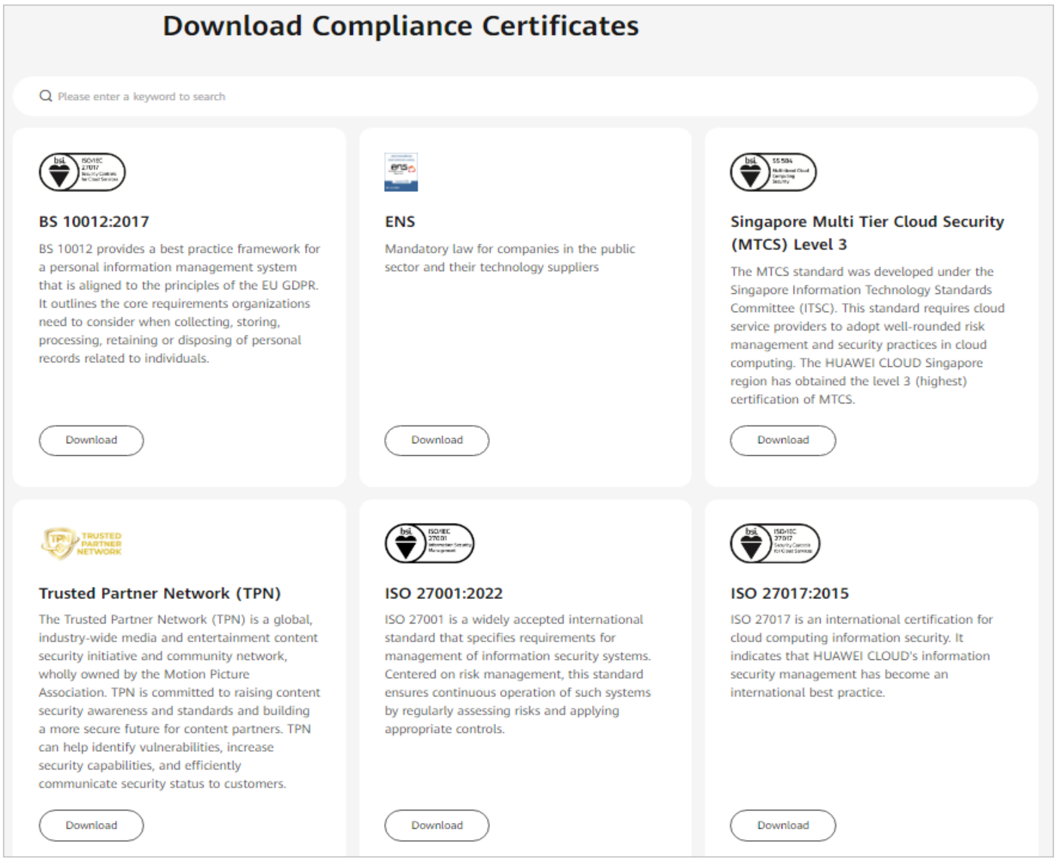
For details, see **Cluster Monitoring**.

9.8 Certificates

Compliance Certificates

Huawei Cloud services and platforms have obtained various security and compliance certifications from authoritative organizations, such as International Organization for Standardization (ISO). You can **download** them from the console.

Figure 9-5 Downloading compliance certificates



Download Compliance Certificates

Please enter a keyword to search

BS 10012:2017

BS 10012 provides a best practice framework for a personal information management system that is aligned to the principles of the EU GDPR. It outlines the core requirements organizations need to consider when collecting, storing, processing, retaining or disposing of personal records related to individuals.

Download

ENS

Mandatory law for companies in the public sector and their technology suppliers

Download

Singapore Multi Tier Cloud Security (MTCS) Level 3

The MTCS standard was developed under the Singapore Information Technology Standards Committee (ITSC). This standard requires cloud service providers to adopt well-rounded risk management and security practices in cloud computing. The HUAWEI CLOUD Singapore region has obtained the level 3 (highest) certification of MTCS.

Download

Trusted Partner Network (TPN)

The Trusted Partner Network (TPN) is a global, industry-wide media and entertainment content security initiative and community network, wholly owned by the Motion Picture Association. TPN is committed to raising content security awareness and standards and building a more secure future for content partners. TPN can help identify vulnerabilities, increase security capabilities, and efficiently communicate security status to customers.

Download

ISO 27001:2022

ISO 27001 is a widely accepted international standard that specifies requirements for management of information security systems. Centered on risk management, this standard ensures continuous operation of such systems by regularly assessing risks and applying appropriate controls.

Download

ISO 27017:2015

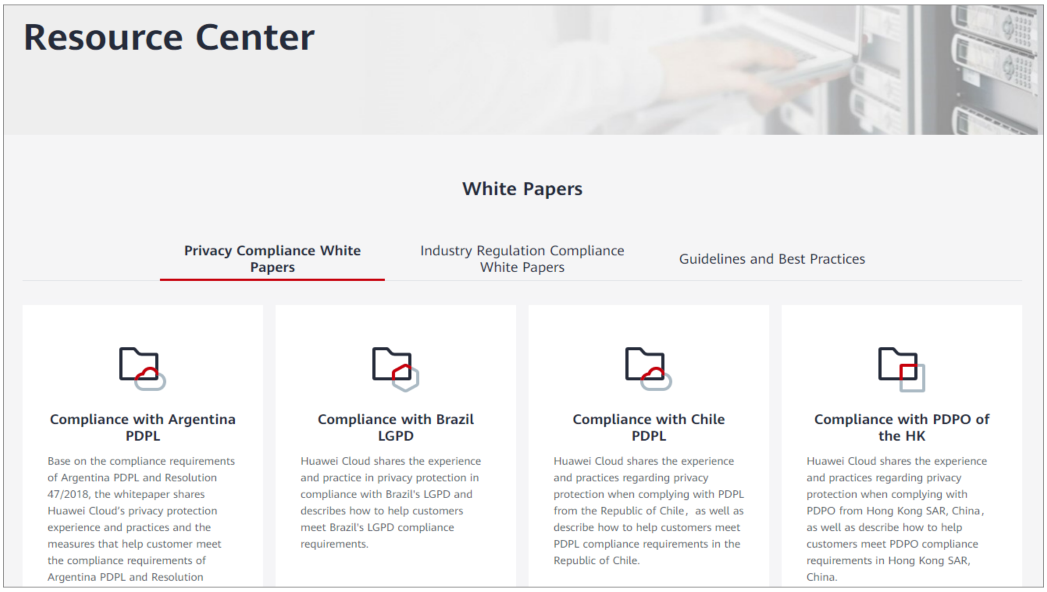
ISO 27017 is an international certification for cloud computing information security. It indicates that HUAWEI CLOUD's information security management has become an international best practice.

Download

Resource Center

Huawei Cloud also provides the following resources to help users meet compliance requirements. For details, see **Resource Center**.

Figure 9-6 Resource center



Resource Center

White Papers

Privacy Compliance White Papers | Industry Regulation Compliance White Papers | Guidelines and Best Practices

Compliance with Argentina PDPL

Base on the compliance requirements of Argentina PDPL and Resolution 47/2018, the whitepaper shares Huawei Cloud's privacy protection experience and practices and the measures that help customer meet the compliance requirements of Argentina PDPL and Resolution

Compliance with Brazil LGPD

Huawei Cloud shares the experience and practice in privacy protection in compliance with Brazil's LGPD and describes how to help customers meet Brazil's LGPD compliance requirements.

Compliance with Chile PDPL

Huawei Cloud shares the experience and practices regarding privacy protection when complying with PDPL from the Republic of Chile, as well as describe how to help customers meet PDPL compliance requirements in the Republic of Chile.

Compliance with PDPO of the HK

Huawei Cloud shares the experience and practices regarding privacy protection when complying with PDPO from Hong Kong SAR, China, as well as describe how to help customers meet PDPO compliance requirements in Hong Kong SAR, China.

9.9 Security Notices

9.9.1 Vulnerability Fixing Description

Table 9-1 Fixed open-source and third-party software vulnerabilities

Software	Version	CVE ID	CSS Score	Description	Affected Version	Fixed Version
log4j	2.13.2	CVE-2021-44228	9.8	Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages, and parameters do not protect against attacker controlled LDAP and other JNDI related endpoints. An attacker who can control log messages or log message parameters can execute arbitrary code loaded from LDAP servers when message lookup substitution is enabled. From log4j 2.15.0, this behavior has been disabled by default. From version 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely removed. Note that this vulnerability is specific to log4j-core and does not affect log4net, log4cxx, or other Apache Logging Services projects.	8.0.0~8.1.2	8.1.3

10 DWS Permissions Management

If you need to assign different permissions to employees in your enterprise to access your DWS resources on Huawei Cloud, IAM is a good choice for fine-grained permissions management. IAM provides identity authentication, permissions management, and access control, helping you secure access to your Huawei Cloud resources.

With IAM, you can use your Huawei Cloud account to create IAM users for your employees, and assign permissions to the users to control their access to specific resource types. For example, to allow software developers in your company to use DWS resources while restricting high-risk operations and resource deletion, you can create IAM users tailored for these developers and grant them only the essential permissions for DWS usage.

If your Huawei Cloud account does not require individual IAM users for permissions management, you can skip this section.

IAM can be used free of charge. You pay only for the resources in your account. For more information about IAM, see **Service Overview**.

DWS Permissions

By default, new IAM users do not have permissions assigned. You need to add a user to one or more groups, and attach permissions policies or roles to these groups. Users inherit permissions from the groups to which they are added and can perform specified operations on cloud services.

DWS is a project-level service deployed and accessed in specific physical regions. To assign DWS permissions to a user group, specify the scope as region-specific projects and select projects for the permissions to take effect. If **All projects** is selected, the permissions will take effect for the user group in all region-specific projects. When accessing DWS, the users need to switch to a region where they have been authorized to use DWS.

- **Role**: IAM initially provides a coarse-grained authorization mechanism to define permissions based on users' job responsibilities. This mechanism provides only a limited number of service-level roles for authorization. When using roles to grant permissions, you must also assign other roles on which the permissions depend to take effect. However, roles are not an ideal choice for fine-grained authorization and secure access control.

- **Policies**: A type of fine-grained authorization mechanism that defines permissions required to perform operations on specific cloud resources under certain conditions. This mechanism allows for more flexible policy-based authorization, meeting requirements for secure access control. For example, you can grant DWS users only the permissions for managing a certain type of DWS resources.

  Most policies define permissions based on APIs. For the API actions supported by DWS, see **Permissions Policies and Supported Actions**.

  For how to create a fine-grained permissions policy, see **Creating a DWS Custom Policy**.

Table 10-1 lists all the system-defined roles and policies supported by DWS.

Table 10-1 DWS system permissions

Role/Policy Name	Description	Category	Dependencies
DWS ReadOnlyAccess	Read-only permissions for DWS. Users granted these permissions can only view DWS data.	System-defined policy	N/A
DWS FullAccess	Database administrator permissions for DWS. Users granted these permissions can perform all operations on DWS.	System-defined policy	N/A
DWS Administrator	Database administrator permissions for DWS. Users granted these permissions can perform operations on all DWS resources. ● Users also granted permissions of the **VPC Administrator** policy can create VPCs or subnets. ● Users also granted permissions of the **CES Administrator** policy can view monitoring information of data warehouse clusters. ● If you need to create an agency, you also need to configure the **Security Administrator** permission.	System-defined role	Dependent on the **Tenant Guest** and **Server Administrator** policies, which must be assigned in the same project as the **DWS Administrator** policy.

Role/Policy Name	Description	Category	Dependencies
DWS Database Access	Database access permissions for DWS. Users granted these permissions can generate temporary database user credentials based on IAM users to connect to databases in the data warehouse clusters.	System-defined role	Dependent on the **DWS Administrator** policy, which must be assigned in the same project as the **DWS Database Access** policy.

Table 10-2 lists the common operations supported by each system-defined policy or role of DWS. Choose appropriate policies or roles as required.

NOTE

- If you use the EIP binding function for the first time in each project of each region, the system prompts you to create the **DWSAccessVPC** agency to authorize DWS to access VPC. After the authorization is successful, DWS can switch to a healthy VM when the VM bound with the EIP becomes faulty.
- In addition to policy permissions, you may need to grant different operation permissions on resources to users of different roles. For details about operations, such as creating snapshots and restarting clusters, see **Syntax of Fine-Grained Permissions Policies**.
- By default, only Huawei Cloud accounts or users with **Security Administrator** permissions can query and create agencies. By default, IAM users in the account do not have the permission to query and create agencies. When an EIP is bound, the binding button is shielded. In this case, you need to contact a user with the DWS Administrator permission to authorize the DWS agency on the current page. For details, see **Allowing DWS to Manage Resources**.

Table 10-2 Common operations supported by system-defined permissions for DWS

Operation	DWS FullAccess	DWS ReadOnlyAccess	DWS Administrator	DWS Database Access
Creating/Restoring clusters	√	x	√	x

Operation	DWS FullAccess	DWS ReadOnlyAccess	DWS Administrator	DWS Database Access
Obtaining the cluster list	√	√	√	x
Obtaining the details of a cluster	√	√	√	x
Setting automated snapshot policy	√	x	√	x
Setting security parameters/ parameter groups	√	x	√	x
Restarting clusters	√	x	√	x
Scaling out clusters	√	x	√	x
Resetting passwords	√	x	√	x
Deleting clusters	√	x	√	x
Configuring maintenance windows	√	x	√	x
Binding EIPs	x	x	√	x
Unbinding EIPs	x	x	√	x
Creating DNS domain names	√	x	√	x
Releasing DNS domain names	√	x	√	x

Operation	DWS FullAccess	DWS ReadOnlyAccess	DWS Administrator	DWS Database Access
Modifying DNS domain names	√	x	√	x
Creating MRS connections	√	x	√	x
Updating MRS connections	√	x	√	x
Deleting MRS connections	√	x	√	x
Adding/ Deleting tags	√	x	√	x
Editing tags	√	x	√	x
Creating snapshots	√	x	√	x
Obtaining the snapshot list	√	√	√	√
Deleting snapshots	√	x	√	x
Copying snapshots	√	x	√	x

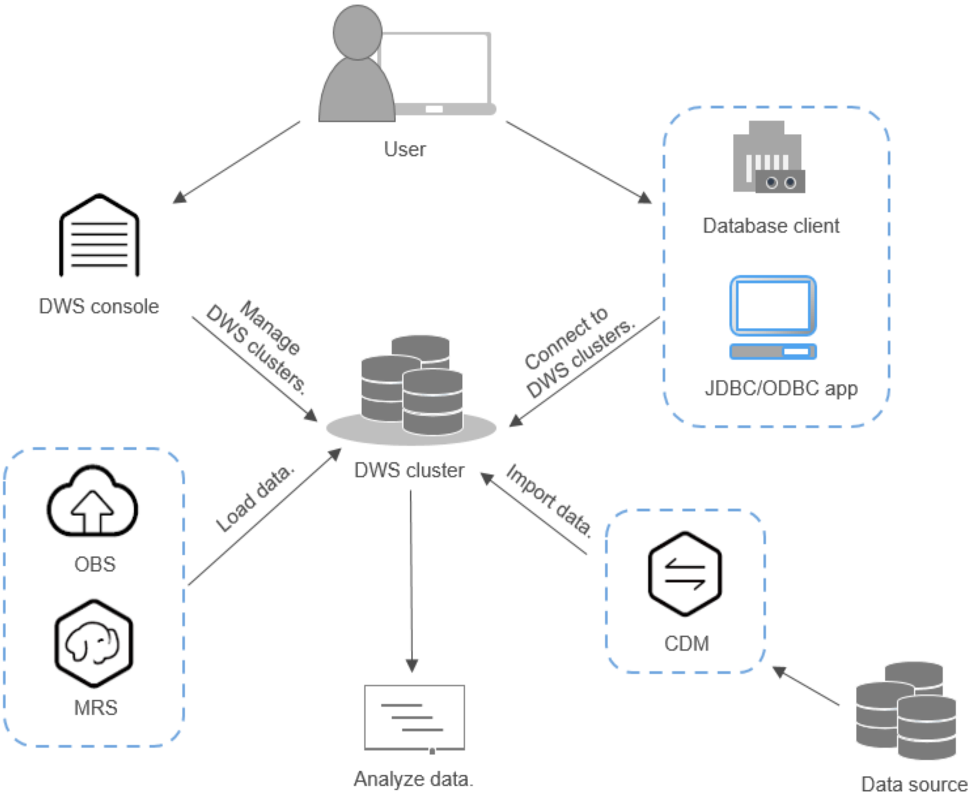
Helpful Links

- **IAM Service Overview**
- **Creating a User and Granting DWS Permissions**
- **Permissions Policies and Supported Actions**

11 Accessing DWS

The following figure shows how to use DWS.

Figure 11-1 Process for using DWS



Accessing a Cluster

DWS provides a web-based management console and HTTPS-compliant APIs for you to manage DWS clusters.

NOTE

In cluster deployment, if a single node is faulty, the abnormal node is automatically skipped when DWS is accessed. However, the cluster performance will be affected.

Accessing the Database in a Cluster

DWS supports database access using the following methods:

- DWS client

  Use the DWS client to access the database in the cluster. For details, see **Using the Linux gsql Client to Connect to a Cluster**, **Using the Data Studio GUI Client to Connect to a Cluster** and **Using the SQL Editor to Connect to a Cluster**.

- JDBC and ODBC API calling

  You can call standard APIs, such as JDBC and ODBC, to access databases in DWS clusters.

  For details, see **Using a JDBC Driver to Connect to a Cluster** and **Using an ODBC Driver to Connect to a Cluster**.

- psycopg2 and PyGreSQL drivers

  After creating a data warehouse cluster, you can use the third-party function library psycopg2 or PyGreSQL to connect to the cluster, and use Python to access DWS and perform various operations on data tables. For details, see **Using the Third-Party Function Library psycopg2 of Python to Connect to a Cluster** and **Using the Third-Party Function Library PyGreSQL of Python to Connect to a Cluster**.

  NOTE

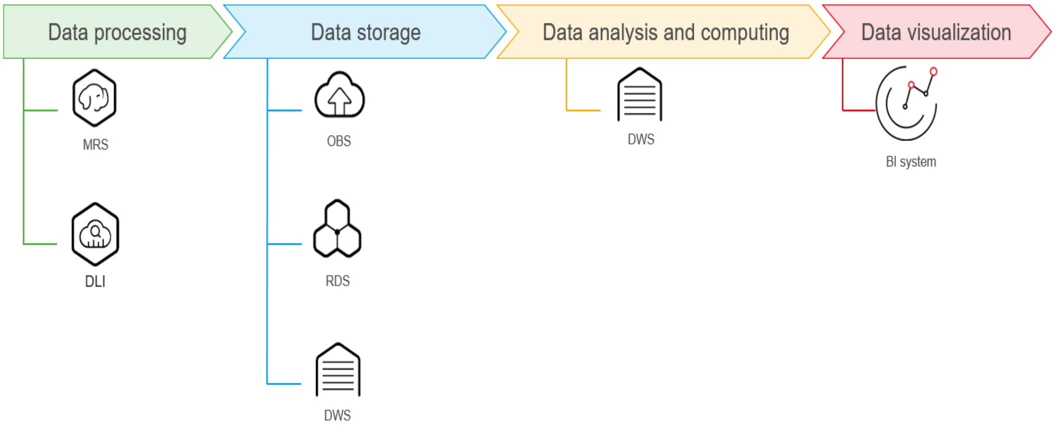
  Currently, DWS does not support cross-database access. Schemas can be used to isolate resources. For details, see **CREATE SCHEMA**.

End-to-End Data Analysis Process

DWS has been seamlessly integrated with other services on Huawei Cloud, helping you rapidly deploy end-to-end data analysis solutions.

The following figure shows the end-to-end data analysis process. Services in use during each process are also displayed.

Figure 11-2 End-to-end data analysis process



12 Restrictions

This document describes the constraints and precautions of using the key functions of DWS.

After creating a DWS cluster, you do not need to perform basic database O&M operations, such as HA and security patch installation. However, you need to pay attention to the following:

Specification and Performance Limits

Table 12-1 Specification and performance limits

Item	Limit	Description
Multi-AZ	3	● The **Multi-AZ** option is displayed only if the number of AZs in the selected region is greater than or equal to 3. If this condition is not met, only a single-AZ cluster can be created. ● For a multi-AZ cluster, only three AZs can be selected at a time so far. Server nodes are evenly distributed among the three AZs. ● The numbers of nodes in a multi-AZ cluster must be a multiple of 3. ● In a multi-AZ cluster, the number of DNs must be less than or equal to 2.

Item	Limit	Description
Storage	- If you want to increase the upper limit of the storage space of a cloud SSD disk, contact customer service. - The amount of storage space available on the local SSD disk depends on the type of data warehouse flavor you choose.	- Cloud SSD disks offer a cost-effective solution for enterprise systems that require moderate performance. - Local SSD disks do not support disk scale-out.
Number of deployed CNs	The value ranges from 3 to the number of cluster nodes. The maximum value is **20** and the default value is **3**.	In a large-scale cluster, you are advised to deploy multiple CNs.
Maximum number of connections	- Number of CN connections: 100 to 262,143 - Number of DN connections: 100 to 262,143	For details about CNs and DNs, see **Logical Cluster Architecture**.

Quota Limits

Table 12-2 Quota limits

Item	Limit	Description
Number of nodes	256	The number of nodes in a new cluster cannot exceed the quota that can be used by a user or 256. If the node quota is insufficient, click **Increase quota** to submit a service ticket and apply for higher node quota.
Tags	Each cluster can have a maximum of 20 tags.	For more information, see **Overview**.

Item	Limit	Description
Snapshot storage space	DWS provides some free-of-charge storage space for you to store the snapshot data.	However, if you use more space than the free-of-charge storage space, the exceeded part is charged according to OBS billing rules. For details, see the **OBS pricing details**. The amount of free space you have is equal to the total storage capacity of your cluster. This can be calculated by multiplying the storage capacity of a single node (standby node) by the number of nodes in the cluster.
Retention period of automated backups	The value ranges from 1 to 31 days. The default value is 7 days.	The system deletes expired snapshots when the retention period ends.

Naming Rules

Table 12-3 Naming rules

Item	Description
Cluster name	Enter 4 to 64 characters. Only letters (case-insensitive), digits, hyphens (-), and underscores (_) are allowed. The name must start with a letter.
Administrator account	● The username can contain only lowercase letters, digits, and underscores (_). ● The username must start with a lowercase letter or an underscore (_). ● The username should contain 6 to 64 characters. ● Cannot be a keyword of the DWS database. For details about the keywords of the DWS database, see **Keyword** in the *Data Warehouse Service Database Developer Guide*.
Role name	The value must start with a letter and can contain a maximum of 63 characters, including letters, digits, and underscores (_).
Username	The value must start with a letter and can contain a maximum of 63 characters, including letters, digits, and underscores (_).

Item	Description
Snapshot name	The snapshot name must be 4 to 64 characters in length and start with a letter. It is case-insensitive and contains only letters, digits, hyphens (-), and underscores (_).
Snapshot policy name	The policy name must be unique, consist of 4 to 92 characters, and start with a letter. It is case-insensitive and can contain only letters, digits, hyphens (-), and underscores (_).
Alarm rule name	The rule name contains 6 to 64 characters and can contain only letters, digits, and slashes (/).
DR name	The DR name contains 4 to 64 case-insensitive characters and must start with a letter. It can contain only letters, digits, hyphens (-), and underscores (_).

Restrictions on Basic Cluster Operations

Table 12-4 Restrictions on basic cluster operations

Item	Description
Binding a load balancer	● To bind a load balancer to a DWS cluster, ensure that the load balancer is in the same region, VPC, and enterprise project as the cluster. ● Only dedicated load balancers can be bound to DWS. ● When you unbind a load balancer from a cluster, related cluster information is cleared on DWS but the load balancer is not deleted. Delete the load balancer in time to prevent unnecessary costs.
Adding or deleting a CN node	● The default number of CNs is 3. You can adjust the number of CNs based on the number of provisioned nodes. The number of CNs ranges from 2 to 20. ● If one of your CNs is abnormal, you can only delete this abnormal CN. If two or more CNs are abnormal, you can delete CNs only after the CNs are recovered from faults.
Managing resource load	● Resources cannot be managed during offline scale-out. If a resource management plan is enabled, stop it before performing offline scale-out.

Item	Description
Managing logical clusters	● You are advised to allocate tables in a database to the same logical cluster. ● If the original physical cluster contains data, it is not possible to switch the logical cluster of a cluster. Ensure the original physical cluster is empty during the switchover.
Restarting a cluster	A cluster cannot provide services during the restart. Therefore, before the restart, ensure that no task is running and all data is saved. If a cluster is processing transactional data, for example, importing data, querying data, creating snapshots, or restoring snapshots, files may be damaged or the cluster may fail to be restarted once the cluster is restarted. You are advised to stop all cluster tasks before restarting a cluster.
Starting or stopping a cluster	After the cluster is stopped, ECS basic resources (vCPUs and memory) are no longer reserved. When you start the service again, it may fail to be started due to insufficient resources. In this case, wait for a while and try again later.

Restrictions on Cluster O&M Operations

Table 12-5 Cluster O&M operation restrictions

Item	Description
Scaling out a cluster	● The cluster will be intermittently disconnected during scale-out. Exercise caution when performing this operation. ● Certain cluster functions, including restarting, stopping, and starting, modifying specifications, adding or removing CNs, creating snapshots, and resetting the database administrator's password, cannot be performed while scaling out the cluster. ● If a cluster is billed in yearly/monthly mode, new nodes in the cluster will also be billed in this mode. ● This function can be manually enabled only when the cluster task information displays **To be redistributed** after scale-out.

Item	Description
Scaling in a cluster	● When scaling in a cluster, several functions are disabled, including cluster restart, cluster scale-out, snapshot creation, node management, intelligent O&M, resource management, parameter modification, security configurations, log service, database administrator password resetting, and cluster deletion.
Performing elastic specification changes	● Elastic specification change is only supported by storage-compute coupled and decoupled clusters that use ECSs and EVS disks. Clusters with local ECSs do not have this capability. ● Stop the VM before changing the flavor. The flavor change can only be done offline and it takes 5 to 10 minutes.
Performing classic specification changes	● A cluster billed in yearly/monthly mode does not support this feature. ● A cluster can have up to 240 nodes. The old and new clusters can have up to 480 nodes in total. ● Logical clusters do not support the **Change all specifications** option.
Backing up and restoring a cluster	● **Backing up the cluster is essential for maintaining data reliability, especially when the service provider cannot restore data through upstream re-import. This helps prevent data loss caused by human or other factors.** ● If a snapshot is being created for a cluster, the cluster cannot be restarted, scaled, its password cannot be reset, and its configurations cannot be modified.
Upgrading a cluster	● If you are using a version of 8.1.3 or earlier, you will not be able to roll back or submit upgrade tasks until the cluster upgrade is finished. ● DR cannot be established after a hot patch is installed in a cluster.

Item	Description
Managing DR tasks	● Storage-compute decoupled clusters and multi-AZ clusters do not support DR. ● If the DR task is stopped or encounters an abnormal situation, but the DR cluster remains normal, it can still provide read services. Once the DR switchover is completed, the DR cluster can provide both read and write services. ● After creating the DR cluster, the snapshot function of the production cluster remains normal, but the snapshot function of the DR cluster is disabled, along with the restoration function for both the production and DR clusters. ● Logical clusters and resource pools are not supported.
Managing logs	● This function cannot be used if OBS is not available. ● When CNs are changed, such as modifying specifications or adding/deleting CNs, there is a risk of data loss. To mitigate this risk, disable audit log dump while performing the change.

13 Technical Support

DWS is a cloud-based data warehousing solution powered by Huawei Cloud. It offers scalable, ready-to-use, and fully managed analytical database services. Adhering to ANSI/ISO SQL92, SQL99, and SQL 2003 standards, it ensures compatibility with major database ecosystems like PostgreSQL, Oracle, Teradata, and MySQL. This makes it a competitive option for petabyte-scale big data analytics across diverse sectors.

Maintenance Policy Statement

As a data warehouse service, DWS offers cloud service capabilities that are fully managed using these resources. Users have complete control over their clusters. Cloud services provide monitoring and alarms for customer clusters, but cannot perform operations on them. Users are responsible for routine cluster maintenance. If you experience technical issues, contact the technical support team for help. Note that they only assist with DWS cloud services, including those used in application systems.

Technical Support Scope

- **Supported services**

  The DWS console provides the following functions:

  - Cluster creation, deletion, scaling, specification adjustment, upgrade, patching, and backup and restoration
  - Cluster monitoring and alarm management
  - IAM user agency management
  - External API management
- **Unsupported services**
  - DWS is not responsible for handling inquiries regarding customer service application development on its clusters. This covers service design, code development, job performance optimization, and service migration. If you need support, contact and purchase the corresponding expert service.
  - DWS does not troubleshoot or analyze job running exceptions if there are no visible problems or defects with its cluster component service.

14 Service Quotas

Quotas are enforced for service resources on the platform to prevent unforeseen spikes in resource usage. Quotas limit the number or amount of resources available to users.

Table 14-1 shows the default user quotas of GaussDB(DWS). For how to view and increase quotas, see **Quotas**.

Table 14-1 Service quotas

Resource Type	Total Quota
Nodes	256

15 DWS Technical Specifications

This section describes the technical specifications of DWS in different versions.

Table 15-1 Technical specifications of DWS 8.1.3 – 9.1.0

Technical Specifications	Maximum Value of 8.1.3	Maximum Value of 8.2.0	Maximum Value of 8.2.1	Maximum Value of 8.3.0	Maximum Value of 9.1.0
Number of cluster nodes	2048	2048	2048	2048	● Storage-compute integration: 2048 ● Decoupled storage and compute: The multi-VW technology is used to support up to 256 VWs, each with up to 1,024 DNs. It is recommended that you limit VWs to 32 or fewer and DNs to 128 or fewer per VW. The total number of DNs of all VWs cannot exceed 2,048.

Technical Specifications	Maximum Value of 8.1.3	Maximum Value of 8.2.0	Maximum Value of 8.2.1	Maximum Value of 8.3.0	Maximum Value of 9.1.0
Number of concurrent connections	Number of concurrent complex queries in minutes: 80 Number of short queries in seconds: 500 Number of concurrent short transactions in milliseconds: 5000	Number of concurrent complex queries in minutes: 80 Number of short queries in seconds: 500 Number of concurrent short transactions in milliseconds: 5000	Number of concurrent complex queries in minutes: 80 Number of short queries in seconds: 500 Number of concurrent short transactions in milliseconds: 5000	Number of concurrent complex queries in minutes: 80 Number of short queries in seconds: 500 Number of concurrent short transactions in milliseconds: 5000	● Storage-compute integration: Number of concurrent complex queries in minutes: 80 Number of short queries in seconds: 500 Number of concurrent short transactions in milliseconds: 5000 ● Decoupled storage and compute: The multi-VW technology can increase the number of concurrent requests. As the number of VWs increases, the number of concurrent requests can be increased accordingly. The total number of concurrent requests in a cluster is affected by

Technical Specifications	Maximum Value of 8.1.3	Maximum Value of 8.2.0	Maximum Value of 8.2.1	Maximum Value of 8.3.0	Maximum Value of 9.1.0
					the GTM/CCN queuing. It is recommended that the number of concurrent requests be no more than 8192.
Cluster data capacity	20 PB	20 PB	20 PB	20 PB	● Storage-compute integration: 20 PB ● Decoupled storage and compute: Data is stored on OBS. Theoretically, the capacity can be expanded infinitely.
Size of a single table	1 PB	1 PB	1 PB	1 PB	1 PB
Size of data in each row	1 GB	1 GB	1 GB	1 GB	1 GB
Number of columns in a single table: (excluding Hudi tables)	1600	1600	1600	1600	● Row storage: 1600 ● Column storage: 1600 ● HStore: 1600

Technical Specifications	Maximum Value of 8.1.3	Maximum Value of 8.2.0	Maximum Value of 8.2.1	Maximum Value of 8.3.0	Maximum Value of 9.1.0
Number of columns in a Hudi table	N/A	N/A	5000	5000	5000
Number of partitions of the partitioned table	32,768	32,768	32,768	32,768	The maximum value is 32768. It is recommended that the value be no more than 1000.
RTO after a SPOF	60s	60s	60s	60s	60s
RPO after a SPOF	0	0	0	0	0
RTO after cluster DR switchover	60min	60min	60min	60min	60min
RPO after cluster DR switchover	60min	60min	60min	60min	60min

NOTE

Virtual Warehouse (VW): also called compute group. DWS storage-compute decoupling splits a physical cluster into multiple VWs. Different services can be bound to different VWs to isolate service loads and increase the number of concurrent services.

Table 15-2 Technical specifications of DWS 8.0.x-8.1.1

Technical Specifications	Maximum Value of 8.0.*x*	Maximum Value of 8.1.0	Maximum Value of 8.1.1
Data capacity	10 PB	10 PB	20 PB
Number of cluster nodes	256	256	2048

Technical Specifications	**Maximum Value of 8.0.*x***	**Maximum Value of 8.1.0**	**Maximum Value of 8.1.1**
Size of a single table	1 PB	1 PB	1 PB
Size of data in each row	1 GB	1 GB	1 GB
Size of a single column in each record	1 GB	1 GB	1 GB
Number of records in each table	2^{55}	2^{55}	2^{55}
Number of columns in each table	1600	1600	1600
Number of indexes in each table	Unlimited	Unlimited	Unlimited
Number of columns in the index of each table	32	32	32
Number of constraints in each table	Unlimited	Unlimited	Unlimited
Number of concurrent connections	Number of concurrent complex queries in minutes: 60 Number of concurrent short transactions in milliseconds: 5000	Number of concurrent complex queries in minutes: 60 Number of concurrent short transactions in milliseconds: 5000	Number of concurrent complex queries in minutes: 80 Number of concurrent short transactions in milliseconds: 5000
Number of partitions in a partitioned table	32,768	32,768	32,768
Size of each partition in a partitioned table	1 PB	1 PB	1 PB
Number of records in each partition in a partitioned table	2^{55}	2^{55}	2^{55}

NOTE

The maximum number of concurrent connections is based on the data warehouse with the cloud disk flavor of 48 vCPUs or 64 vCPUs. For example, you can choose **dwsk.12xlarge (48 vCPU | 384GB | 24000GB SSD)** or **dwsx2.16xlarge.m7 (64 vCPU | 512GB | 32000GB SSD)** for a storage-compute coupled data warehouse.