Data Warehouse Service

Service Overview

Issue 01

Date 2024-12-27

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

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

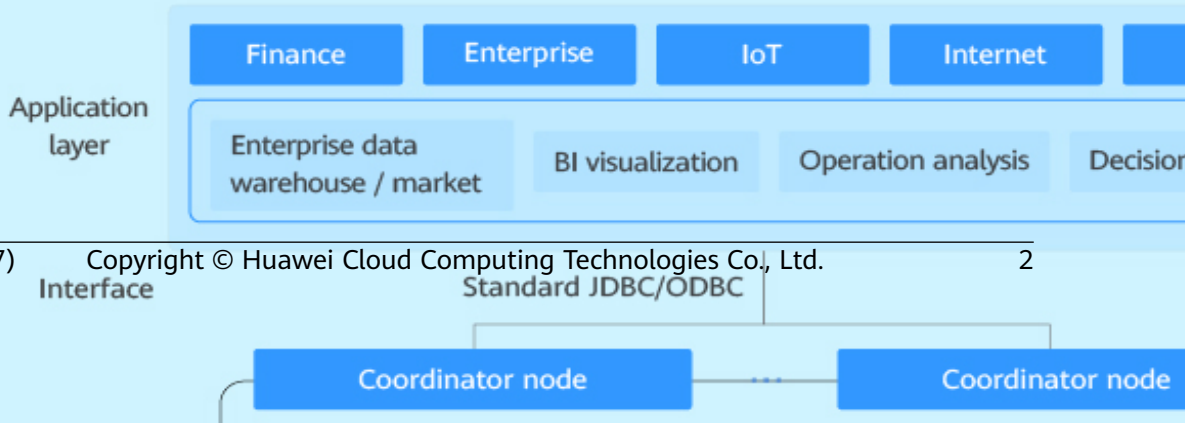
The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 GaussDB(DWS) Infographics

01/ What Is Data Warehouse Servic

GaussDB(DWS) is an online data processing database that ru
infrastructure to provide scalable, fully-managed, and out-of-t
database service, freeing you from complex database manage
monitoring. It is a native cloud service based on the converge
GaussDB, and is compatible with ANSI/ISO SQL92, SQL99, an
Additionally, GaussDB(DWS) is interoperable with other datab
such as PostgreSQL, Oracle, Teradata, and MySQL. GaussDB(D
competitive solutions for PB-level big data analysis in various



2 What Is GaussDB(DWS)?

GaussDB(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 SQL92, SQL99, and SQL 2003 syntax. Additionally, GaussDB(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.

GaussDB(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**.

- **Computing In-Memory(CIM)**: The storage-compute coupled 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**: 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.

GaussDB(DWS) is widely used in domains such as finance, Internet of Vehicles (IoV), government and enterprise, e-commerce, energy, and telecom. It has been listed in the Gartner Magic Quadrant for Data Management Solutions for Analytics for two consecutive years. Unlike conventional data warehouses, GaussDB(DWS) is more cost-effective and has large-scale scalability and enterprise-level reliability.

In addition, GaussDB(DWS) can be deployed on physical machines. For details, see **Physical Machine Deployment**.

Logical Cluster Architecture

Figure 2-1 shows the logical architecture of a GaussDB(DWS) cluster. For details about the instance, see Table 2-1.

Figure 2-1 Logical cluster architecture

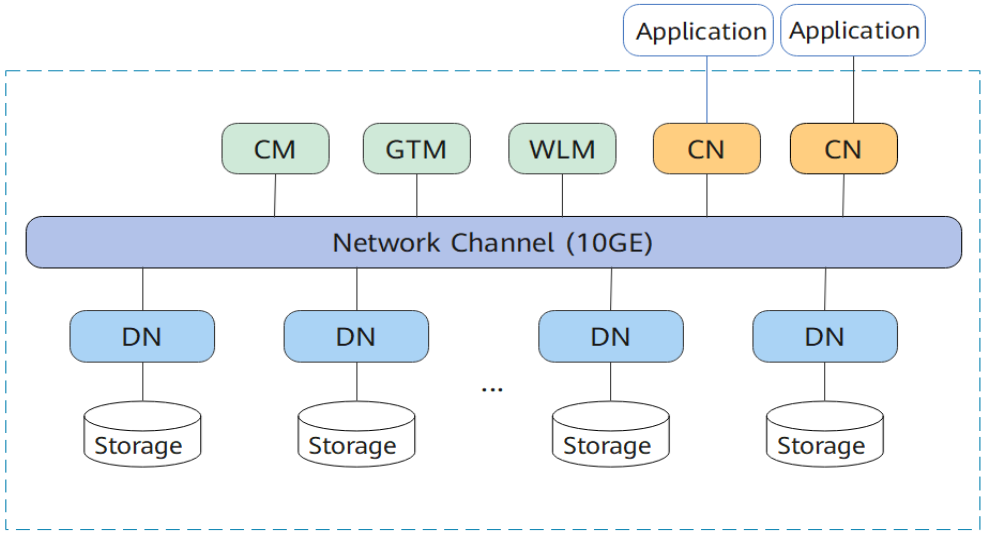


Table 2-1 Cluster architecture description

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. GaussDB(DWS) provides the primary/standby CM Server solution to ensure system HA. 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. GaussDB(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.

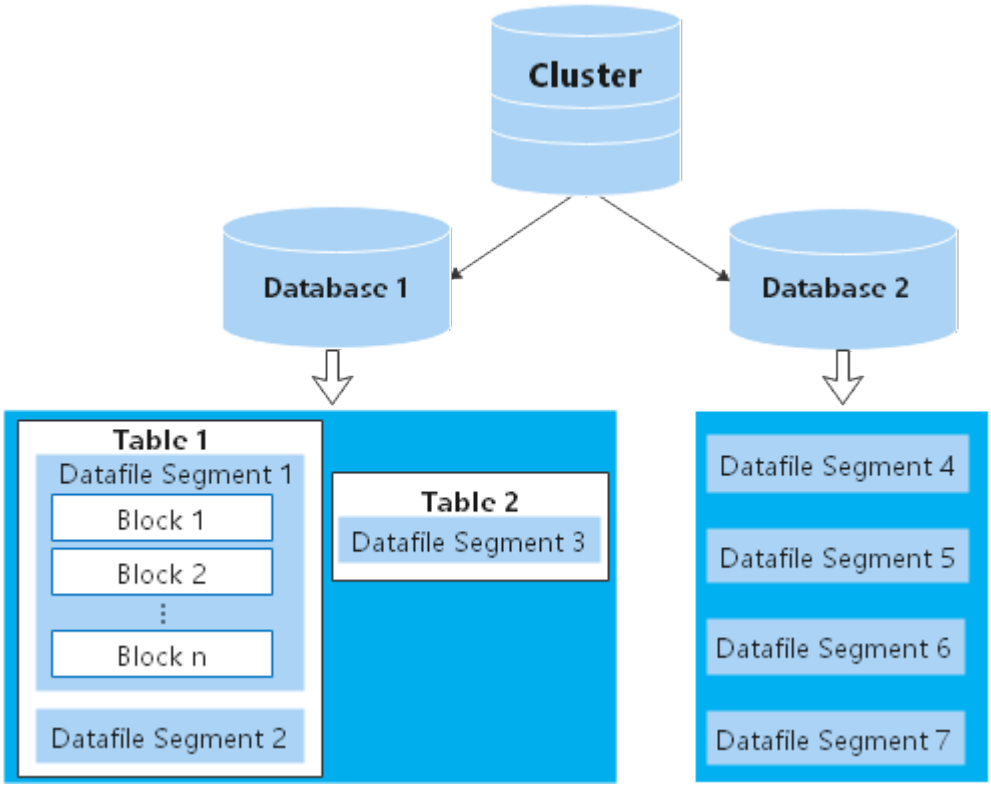
Name	Function	Description
Datanode (DN)	A DN stores data in row-store, column-store, or hybrid mode, executes data query tasks, and returns execution results to CNs.	There are multiple DNs in the cluster. Each DN stores part of data. GaussDB(DWS) provides DN high availability: active DN, standby DN, 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 2-2** 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 2-2 Logical database architecture

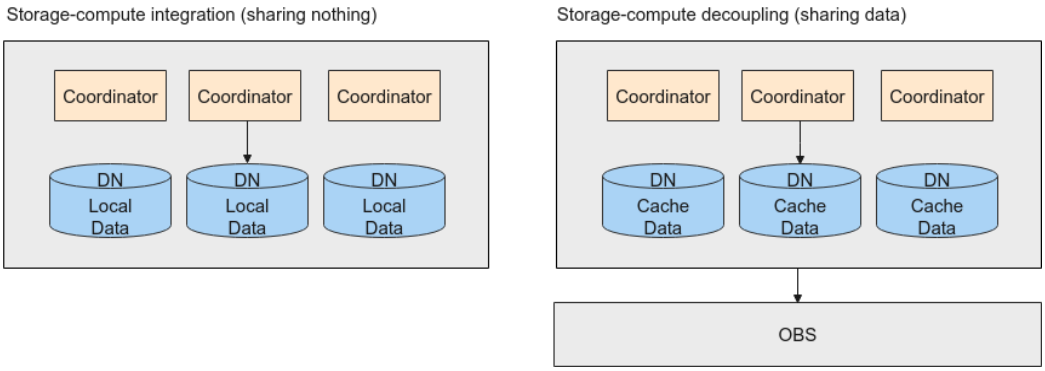


Physical Architecture of a Cluster

GaussDB(DWS) supports the storage-compute coupled and decoupled architectures.

In the storage and 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 2-3 Architecture selection

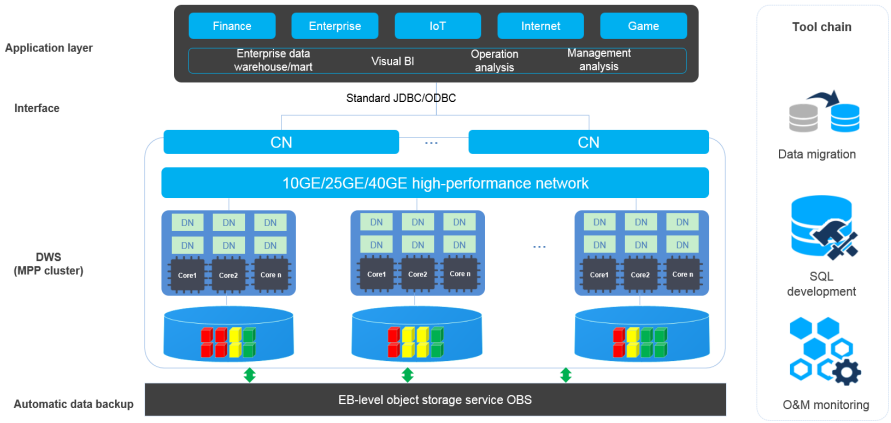


Storage-Compute Coupled Architecture

GaussDB(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 2-4 Architecture



- **Application layer**

  Data loading tools, extract, transform, and load (ETL) tools, business intelligence (BI) tools, as well as data mining and analysis tools, can be integrated with GaussDB(DWS) through standard APIs. GaussDB(DWS) is compatible with the PostgreSQL ecosystem, and the SQL syntax is compatible with Oracle, MySQL, and Teradata. Applications can be smoothly migrated to GaussDB(DWS) with few changes.

- **API**

  Applications can connect to GaussDB(DWS) through standard JDBC and ODBC.

- **GaussDB(DWS)**

  A GaussDB(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 GaussDB(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

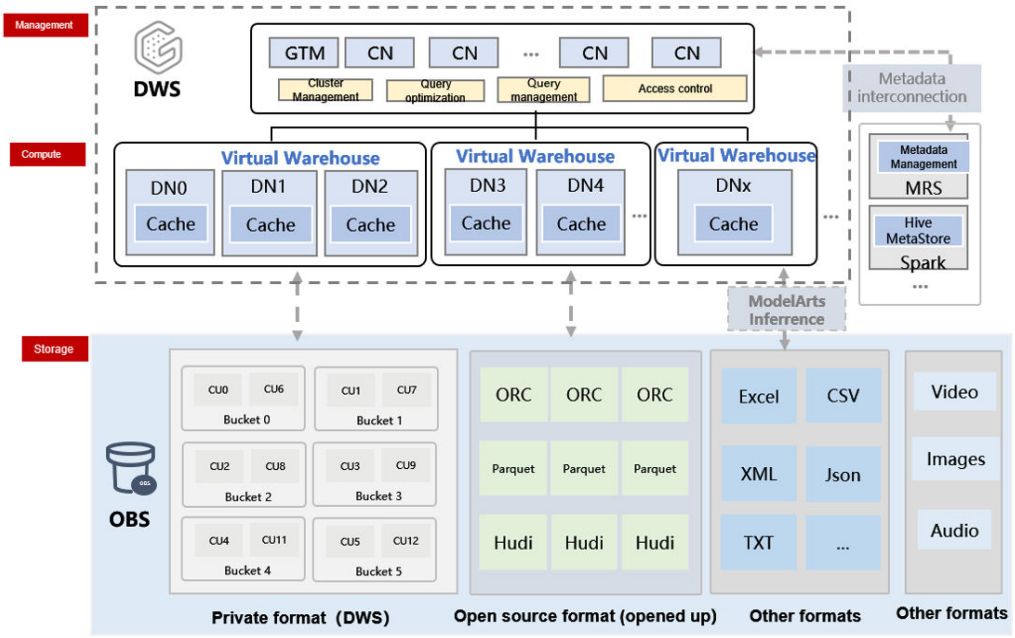
The newly released GaussDB(DWS) storage-compute decoupled cluster provides resource pooling, massive storage, and the MPP architecture with decoupled compute and storage. This enables high elasticity, real-time data import and sharing, and lake warehouse integration.

The GaussDB(DWS) storage-compute decoupled cluster enables independent scaling of compute and storage resources by separating compute and storage functionalities. 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 GaussDB(DWS) storage-compute decoupled cluster 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 2-5 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 2-2 Differences between storage-compute coupled and decoupled architectures

Version	Coupled storage and compute	Decoupled storage and compute
Storage medium	Data is stored on local disks of compute nodes.	Column-store data is stored in Huawei Cloud OBS. Local disks 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 locally on 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.

3 Data Warehouse Types

Product Type Overview

- **Standard Data Warehouse (DWS 2.0):** Oriented to data analysis scenarios, DWS 2.0 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.
- **Standard data warehouse (DWS 3.0):** DWS 3.0 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.
- **Hybrid data warehouse**: a cost-effective solution that goes beyond large-scale data query and analysis. It also offers high-concurrency, high-performance, and low-latency transaction processing capabilities. The data warehouse can be used to process HTAP hybrid loads, and can be deployed in standalone or cluster mode.

NOTE

- GaussDB(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 3-1 Features

Module	Function	Standard Data Warehouse (DWS 2.0)	Standard Data Warehouse (DWS 3.0)	Hybrid (Cluster)	Hybrid (Standalone)
Dashboard	Resources	Yes	Yes	Yes	Yes
	Alarms	Yes	Yes	Yes	Yes

Module	Function	Standard Data Warehouse (DWS 2.0)	Standard Data Warehouse (DWS 3.0)	Hybrid (Cluster)	Hybrid (Standalone)
	Recent events	Yes	Yes	Yes	Yes
	Cluster monitoring metrics (DMS)	Yes	Yes	Yes	No
Data	-	Yes	Yes	Yes	Yes
Cluster management	SQL editor	Yes	Yes	Yes	Yes
	Monitoring panel (DMS)	Yes	Yes	Yes	No
	Monitoring metrics (Cloud Eye)	Yes	Yes	Yes	No
	Restart	Yes	Yes	Yes	Yes
	Start	No	No	No	No
	Stop	No	No	No	No
	Scaling	Yes	Yes	Yes	No
	Scale-in	Yes	Yes	Yes	No
	Redistributing data	Yes	Yes (Note 1)	Yes	No
	Viewing redistribution details	Yes	Yes	Yes	No
	Changing the node flavor	Yes	Yes	Yes	Yes
	Changing all specifications	Yes	No	Yes	No
	Resetting passwords	Yes	Yes	Yes	Yes
	Creating snapshots	Yes	Yes	Yes	No

Module	Function	Standard Data Warehouse (DWS 2.0)	Standard Data Warehouse (DWS 3.0)	Hybrid (Cluster)	Hybrid (Standalone)
	Canceling read-only status	Yes	Yes	Yes	Yes
	Deletion	Yes	Yes	Yes	Yes
	Managing CNs	Yes	Yes	Yes	No
	Storage space scaling	Yes	Yes	Yes	Yes
Basic Information	Basic information	Yes	Yes	Yes	Yes
	ELB	Yes	Yes	Yes	N
	Resource Management	Yes	Yes	Yes	N
	Intelligent O&M	Yes	Yes	Yes	No
	Logical cluster	Yes	Yes	Yes	N
	Snapshot	Yes	Yes	Yes	No
	Parameter modifications	Yes	Yes	Yes	Yes
	Security settings	Yes	Yes	Yes	Yes
	MRS data sources	Yes	Yes	Yes	No
	Tags	Yes	Yes	Yes	Yes
	Node management	Yes	Yes	Yes	No
	Upgrade management	Yes	Yes	Yes	No
	Logging	Yes	Yes	Yes	No

Module	Function	Standard Data Warehouse (DWS 2.0)	Standard Data Warehouse (DWS 3.0)	Hybrid (Cluster)	Hybrid (Standalone)
	User management	Yes	Yes	Yes	No
Integration	Data migration	Yes	Yes	Yes	No
DR management	DR management	Yes	No	Yes	No
Snapshot management	Restoration	Yes	Yes	Yes	No
	Deletion	Yes	Yes	Yes	No
	Copy	Yes	Yes	Yes	No
Incident management	Event management (general)	Yes	Yes	Yes	Yes
Alarm management	Alarm management	Yes	Yes	Yes	Yes
Client connections	Client connections	Yes	Yes	Yes	Yes
Others	Inspection	Yes	Yes	Yes	No
	Intelligent O&M	Yes	Yes	Yes	No
	Node restoration	Yes	Yes	Yes	No
	Warm backup on the tenant side	Yes	Yes	Yes	No

NOTE

- Note 1: The storage-compute separated 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 separated 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.
- A hybrid data warehouse deployed in standalone mode cannot be changed to the distributed mode. Scaling, logical cluster operations, and resource management are not supported.
- A hybrid data warehouse deployed in standalone mode does not support snapshots, DMS monitoring, or MRS data sources.
- In a hybrid data warehouse deployed in standalone mode, the original syntax for specifying distribution columns will be ignored. Scheduled jobs, the sequence type, HDFS/OBS foreign table import and export, multi-temperature storage, and auto-increment partitioned tables are not supported. OBS foreign table import and export are supported in 8.2.0 and later versions.
- Only DWS 3.0 clusters of 9.0.2 and later versions support the snapshot function.

4 Data Warehouse Flavors

GaussDB(DWS) provides standard and hybrid data warehouses. The hybrid data warehouse supports the standalone deployment. For details about the differences between them, see **Data Warehouse Types**.

Standard Data Warehouse (DWS 2.0) Flavors

- A standard data warehouse (DWS 2.0) using cloud disks can be elastically scaled, providing unlimited computing and storage capacity. For details, see **Table 4-1**.
- A standard data warehouse (DWS 2.0) using local disks cannot be scaled up. You can only increase capacity by adding nodes. For details, see **Table 4-2**.

  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 4-1 Cloud disk flavors of a standard data warehouse (DWS 2.0)

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Default Storage	Step (GB)	Recommended Storage	DN number	Scenario
dwsx2.xlarge.m7	x86	4	32	20GB ~ 2000GB	100	10	800	1	Suitable for GaussDB(DWS) starters. These flavors can be used for testing, learning environments, or small-scale analytics systems.
dwsk2.xlarge	ARM	4	32	20GB ~ 2000GB	100	10	800	1	
dwsx2.xlarge.m7n	x86	4	32	20GB ~ 2000GB	100	10	800	1	
dwsx2.2xlarge.m7	x86	8	64	100 GB – 4000 GB	200	100	1600	1	Suitable for internal data warehousing and report analysis in small- and medium-sized enterprises (SMEs).
dwsk2.2xlarge	ARM	8	64	100 GB – 4000 GB	200	100	1600	1	
dwsx2.2xlarge.m7n	x86	8	64	100 GB – 4000 GB	200	100	1600	1	
dwsx2.4xlarge.m7	x86	16	128	100GB ~ 8000GB	400	100	3200	1	
dwsk2.4xlarge	ARM	16	128	100GB ~ 8000GB	400	100	3200	1	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Default Storage	Step (GB)	Recommended Storage	DN number	Scenario
dwsx2.8xlarge.m7	x86	32	256	100GB ~ 16000 GB	800	100	6400	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	100GB ~ 16000 GB	800	100	6400	2	
dwsx2.8xlarge.m7n	x86	32	256	100GB ~ 16000 GB	800	100	6400	2	
dwsk2.12xlarge	ARM	48	384	100 GB – 24,000 GB	1200	100	9600	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	1600	100	12800	4	
dwsx2.16xlarge.m7n	x86	64	512	100 GB – 32,000 GB	1600	100	12800	4	

Table 4-2 Local disk flavors of a standard data warehouse (DWS 2.0)

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	DN number	Scenario
dws2.olap.4xlarge.i3	x86	16	128	1490GB	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	2980GB	1	
dws2.olap.8xlarge.i3	x86	32	256	2980GB	2	
dws2.olap.8xlarge.ki1	Arm	32	128	5960GB	2	
dws2.olap.16xlarge.i3	x86	64	512	5960GB	4	
dws2.olap.16xlarge.ki1	Arm	64	228	11921GB	4	

Standard Data Warehouse (DWS 3.0) Flavors

- A standard data warehouse (DWS 3.0) using cloud disks can be elastically scaled, providing unlimited computing and storage capacity. For details, see **Table 4-3**.
- A standard data warehouse (DWS 3.0) using cloud disks have fixed flavors. You can only expand it by adding nodes. For details about the flavors, see **Table 4-4**.

Table 4-3 Standard data warehouse (DWS 3.0) flavors

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	DN number	Scenario
dwsx3.4U16G.4DPU	x86	4	16	20GB~2000GB	10	1	Suitable for GaussDB(DWS) starters. These flavors can be used for testing, learning environments, or small-scale analytics systems.
dwsk3.4U16G.4DPU	Arm	4	16	20GB~2000GB	10	1	
dwsx3.8U32G.8DPU	x86	8	32	100GB~4000GB	100	1	Suitable for internal data warehousing and report analysis in small- and medium-sized enterprises (SMEs).
dwsk3.8U32G.8DPU	Arm	8	32	100GB~4000GB	100	1	
dwsx3.16U64G.16DPU	x86	16	64	100GB~8000GB	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	100GB~8000GB	100	1	
dwsx3.32U128G.32DPU	x86	32	128	100GB~16000GB	100	2	
dwsk3.32U128G.32DPU	Arm	32	128	100GB~16000GB	100	2	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	DN number	Scenario
dwsk3.48U192G.48DPU	Arm	48	192	200GB~24000GB	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	200GB~32000GB	100	4	

Table 4-4 Local disk flavors of a standard data warehouse (DWS 3.0)

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	DN number	Scenario
dws3.16U128G.i7.16DPU	x86	16	128	2980GB	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	5960GB	1	
dws3.32U256G.i7.32DPU	x86	32	256	5960GB	2	
dws3.32U128G.ki1.32DPU	ARM	32	128	11920GB	2	
dws3.64U512G.i7.64DPU	x86	64	512	11920GB	4	
dws3.64U228G.ki1.64DPU	ARM	64	228	23840GB	4	

Hybrid Data Warehouse Flavors

- A hybrid data warehouse can be deployed in cluster or standalone mode.
  - Cluster deployment: If the name of the selected node flavor contains **h**, the hybrid data warehouse can be deployed in cluster mode. You can deploy multiple nodes, scale nodes, and manage resource pools. For more information, see **Table 4-5**.
  - Standalone deployment: If the name of the selected node flavor contains **h1**, the hybrid data warehouse only supports standalone deployment, which does not provide HA capabilities. The storage cost can be reduced by half. A standalone data warehouse can be restored by the automatic reconstruction of ECS, and its data reliability is ensured by the EVS multi-copy mechanism. For more information, see **Table 4-6**. It is less expensive than other flavors and is a good choice for lightweight services.

Table 4-5 Hybrid data warehouse (cluster) flavors

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	DN number	Scenario
dwsx2.h.xlarge.4.c7	x86	4	16	20GB ~ 2000GB	20	1	Suitable for GaussDB(DWS) starters. These flavors can be used for testing, learning environments, or small-scale analytics systems.
dwsk2.h.xlarge.4.kc1	Arm	4	16	20GB ~ 2000GB	20	1	
dwsx2.h.xlarge.4.c7n	x86	4	16	20GB ~ 2000GB	20	1	
dwsx2.h.2xlarge.4.c6	x86	8	32	100 GB – 4000 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 – 4000 GB	100	1	
dwsk2.h.2xlarge.4.kc1	Arm	8	32	100 GB – 4000 GB	100	1	
dwsx2.h.2xlarge.4.c7n	x86	8	32	100 GB – 4000 GB	100	1	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	DN number	Scenario
dwsx2.h.4xlarge.4.c7	x86	16	64	100GB ~ 8000GB	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	100GB ~ 8000GB	100	1	
dwsx2.h.4xlarge.4.c7	x86	16	64	100GB ~ 8000GB	100	1	
dwsx2.h.8xlarge.4.c7	x86	32	128	100GB ~ 16000GB	100	2	
dwsk2.h.8xlarge.4.kc1	Arm	32	128	100GB ~ 16000GB	100	2	
dwsx2.h.8xlarge.4.c7n	x86	32	128	100GB ~ 16000GB	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.
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	

Table 4-6 Hybrid data warehouse (standalone) flavors

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	DN number	Scenario
dwsx2.h1.xlarge.2.c7	x86	4	8	20GB ~ 2000GB	20	1	Suitable for GaussDB(DWS) starters. These flavors can be used for testing, learning environments, or small-scale analytics systems.
dwsk2.h1.xlarge.2.kc1	Arm	4	8	20GB ~ 2000GB	20	1	
dwsx2.h1.xlarge.2.c7n	x86	4	8	20GB ~ 2000GB	20	1	
dwsx2.h1.2xlarge.4.c7	x86	8	32	100 GB – 4000 GB	100	1	Suitable for internal data warehousing and report analysis in small- and medium-sized enterprises (SMEs).
dwsk2.h1.2xlarge.4.kc1	Arm	8	32	100 GB – 4000 GB	100	1	
dwsx2.h1.2xlarge.4.c7n	x86	8	32	100 GB – 4000 GB	100	1	
dwsx2.h1.4xlarge.4.c7	x86	16	64	100GB ~ 8000GB	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.h1.4xlarge.4.kc1	Arm	16	64	100GB ~ 8000GB	100	1	
dwsx2.h1.4xlarge.4.c7n	x86	16	64	100GB ~ 8000GB	100	1	
dwsx2.h1.8xlarge.4.c7	x86	32	128	100GB ~ 16000GB	100	2	
dwsk2.h1.8xlarge.4.kc1	Arm	32	128	100GB ~ 16000GB	100	2	
dwsx2.h1.8xlarge.4.c7n	x86	32	128	100GB ~ 16000GB	100	2	

Flavor	CPU Architecture	vCPU	Memory (GB)	Storage Capacity Per Node	Step (GB)	DN number	Scenario
dwsk2.h1.12xlarge.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.
dwsx2.h1.16xlarge.4.c7	x86	64	256	100 GB – 32,000 GB	100	4	
dwsx2.h1.16xlarge.4.c7n	x86	64	256	100GB~32000GB	100	4	

5 Advantages

GaussDB(DWS) uses the GaussDB database kernel and is compatible with PostgreSQL. It transforms from a single OLTP database to an enterprise-level distributed OLAP database oriented to massive data analysis based on the massively parallel processing (MPP) architecture.

Unlike conventional data warehouses, GaussDB(DWS) excels in mass data processing and general platform management with the following features:

Ease of use

- Visualized one-stop management

  GaussDB(DWS) simplifies the entire process from project conception to production deployment. The GaussDB(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

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

High performance

- Cloud-based distributed architecture

  GaussDB(DWS) adopts the MPP-based database so that 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.
- Query response to trillions of data records within seconds

  GaussDB(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.

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

- Fast data loading

  GDS is a tool that helps you with high-speed massively parallel data loading.

- Data Compression in Column Storage

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

  In GaussDB(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 GaussDB(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

  GaussDB(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, GaussDB(DWS) supports automatic full and incremental backup of data, improving data reliability.

Low cost

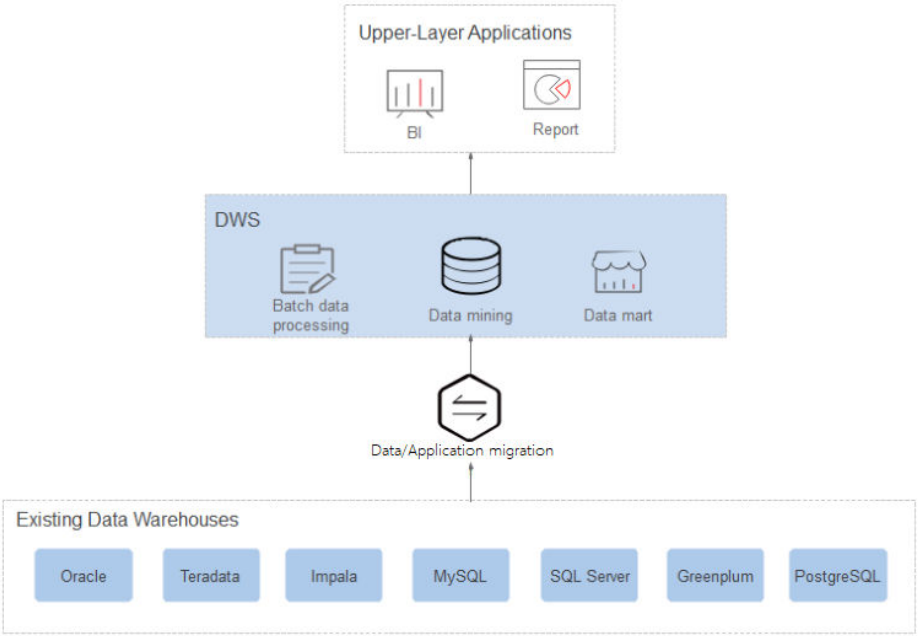
- Pay per use: GaussDB(DWS) is billed based on the usage and use duration. You only need to pay for the resources you use.
- Flexible investment in infrastructure: You do not need to invest much in infrastructure in the early stage. You can start from a data warehouse instance with low specifications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

6 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, GaussDB(DWS) features high performance, low cost, and easy scalability, satisfying requirements in the big data era.

Figure 6-1 Data warehouse migration



Advantages

- Seamless migration

  GaussDB(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

  GaussDB(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.

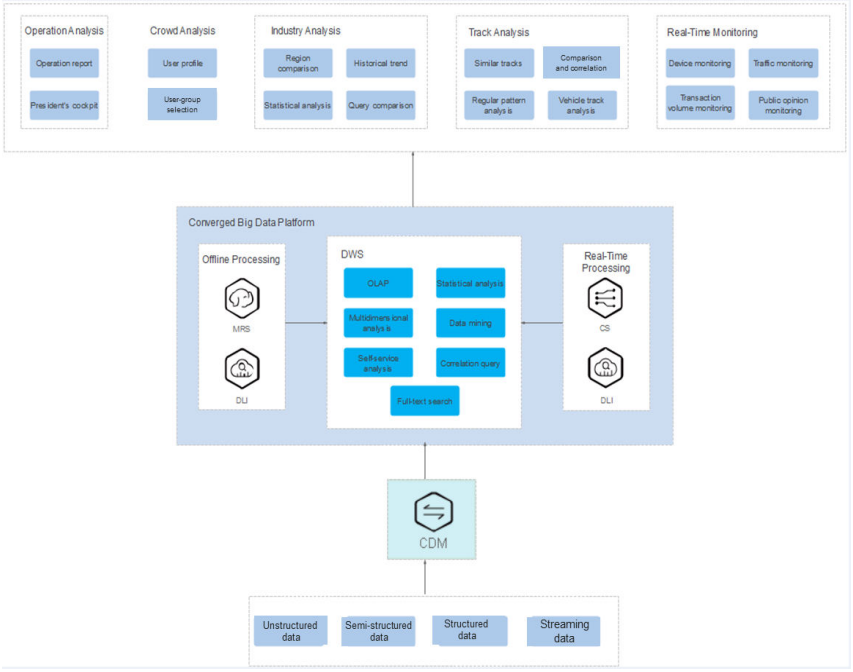
- Secure and reliable

  GaussDB(DWS) supports data encryption and connects to DBSS to ensure data security on the cloud. In addition, GaussDB(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. GaussDB(DWS) delivers the needed processing power to handle these intense compute scenarios.

Figure 6-2 Converged big data analysis



Advantages

- Unified Analysis Entrance

  GaussDB(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

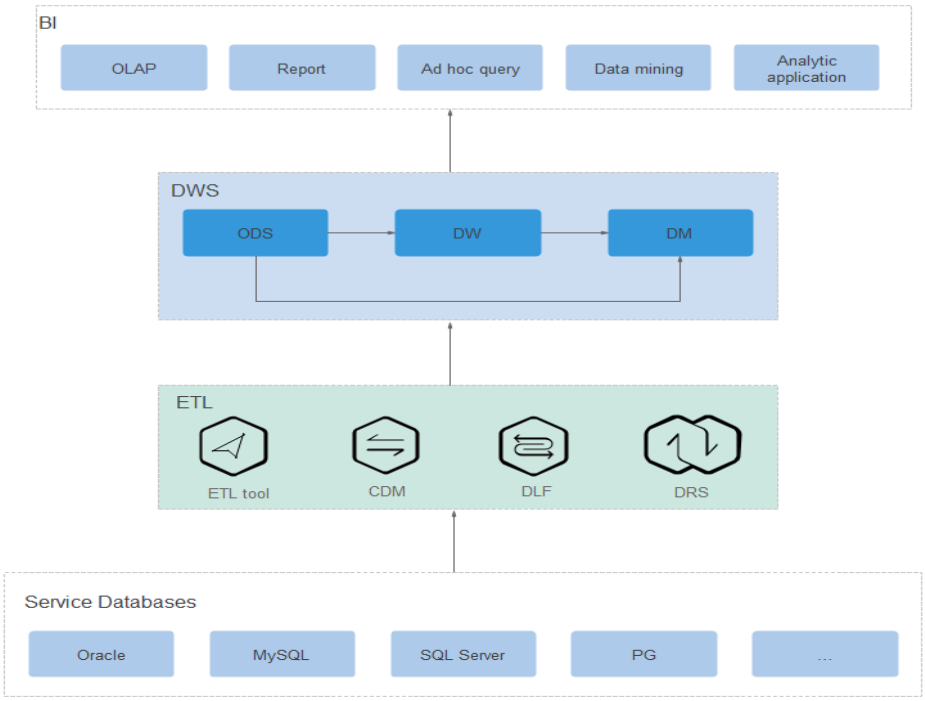
- Data Migration

  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.

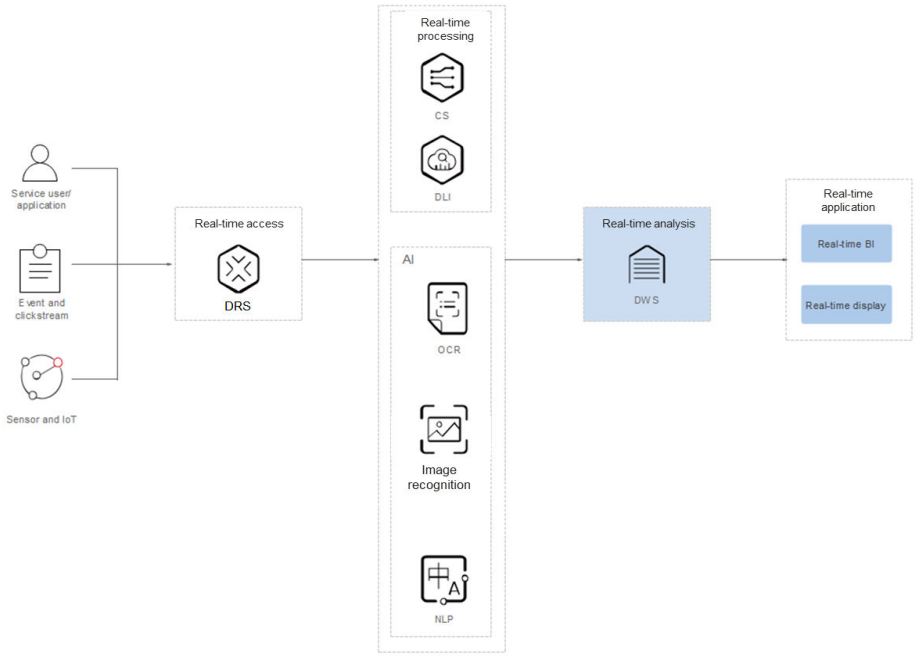
Figure 6-3 Enhanced ETL + real-time BI analysis



Real-Time Data Analytics

In the mobile internet field, processing and analyzing massive amounts of data in real-time is crucial to extract its full potential. GaussDB(DWS) offers fast data import and query capabilities that speed up data analysis, allowing for real-time ingestion, processing, and value generation.

Figure 6-4 Real-time data analysis



Advantages

- Real-Time Import of Streaming Data

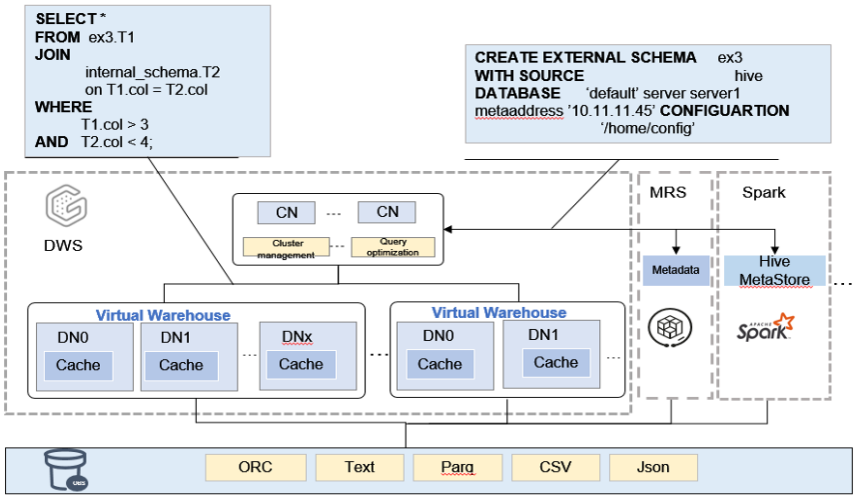
  Data from Internet applications can be written into GaussDB(DWS) in real time after being processed by the stream computing and AI services.
- Real-Time Monitoring and Prediction

  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 GaussDB(DWS).

Lakehouse

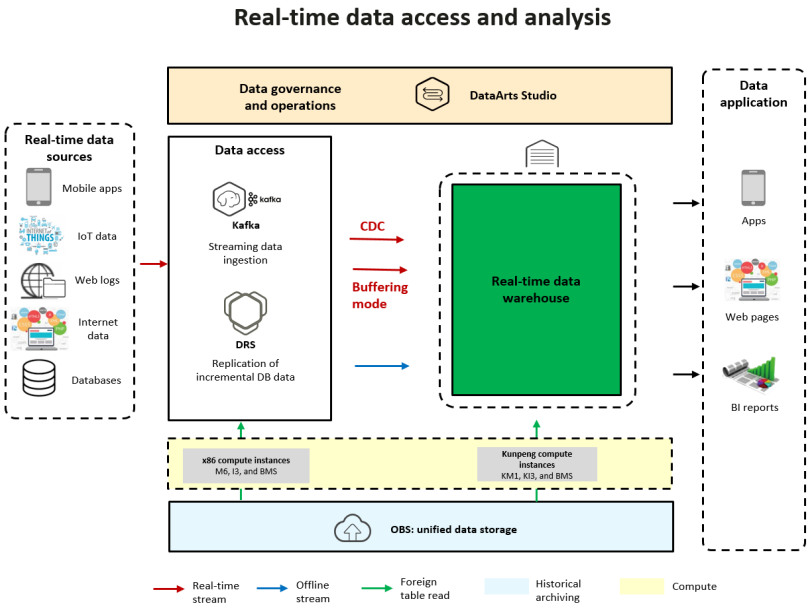
- **Seamless access to the data lake**
  - With the interconnection with Hive Metastore metadata management, you can directly access the data table definitions in the data lake. You do not need to create a foreign table. You only need to create an external schema.
  - The following data formats are supported: 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



Real-Time Write

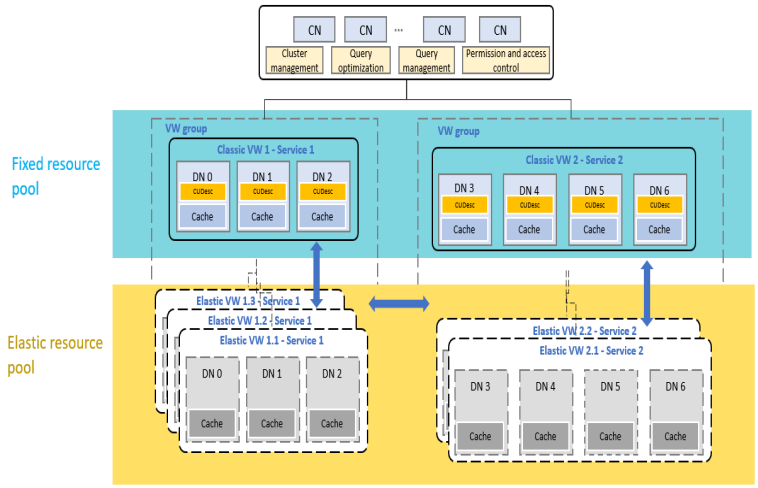
DWS 3.0 utilizes the H-Store storage engine to store micro-batch data locally and syncs it to OBS at regular intervals. It enables high-throughput real-time write and update, as well as large-scale data writes.

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 VWs (Storage-Compute Decoupling)

- 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.

7 Functions

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

Enterprise-Level Data Warehouses and Compatibility with Standard SQL

After a data warehouse 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.

GaussDB(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

GaussDB(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 (DWS) Developer Guide***.

Cluster Management

A GaussDB(DWS) cluster contains nodes of the same flavor in the same subnet. These nodes jointly provide services. GaussDB(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 GaussDB(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 GaussDB(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, GaussDB(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.

- Managing resources

  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 GaussDB(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.

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

- Management console

  Use the management console to access GaussDB(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 GaussDB(DWS) to manage clusters. In addition, if you need to integrate GaussDB(DWS) into a third-party system for secondary development, use APIs to access the service.

For details, see **GaussDB(DWS) API Reference**.

Diverse Data Import Modes

GaussDB(DWS) supports efficient data import from multiple data sources. The following lists typical data import modes. For details, see **Data Migration to GaussDB(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 GaussDB(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 GaussDB(DWS)
- Migrating data to GaussDB(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 GaussDB(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 private 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, GaussDB(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. GaussDB(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 **Configuring Separation of Duties for the GaussDB(DWS) Cluster**.

Monitoring and Auditing

- Monitoring Clusters

  GaussDB(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 GaussDB(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 GaussDB(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

  GaussDB(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
  - GaussDB(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**.
  - GaussDB(DWS) records all SQL operations, including connection attempts, query attempts, and database changes. For details, see **Configuring the Database Audit Logs**.

Multiple Database Tools

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

- gsql

  gsql is a CLI SQL client tool running on the Linux OS. It helps connect to, operate, and maintain the database in a GaussDB(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 GaussDB(DWS) cluster, manage database objects, edit, run, and debug SQL scripts, and view execution plans.

- GDS

  GDS is a data service tool offered by GaussDB(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 GaussDB(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 GaussDB(DWS) by using **gs_restore**. In this way, metadata, such as table definitions and database object definitions, can be imported.

8 Concepts

GaussDB(DWS) Management Concepts

- Cluster

  A cluster is a server group that consists of multiple nodes. GaussDB(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 GaussDB(DWS) cluster can have 3 to 256 nodes. A storage-compute coupled data warehouse (standalone) can only have one node. Each node can store and analyze data. For details, see **GaussDB(DWS) Technical Specifications**.

- Type

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

- Snapshot

  You can create snapshots to back up GaussDB(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.

GaussDB(DWS) Database Concepts

- Databases

  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 data warehouse 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, GaussDB(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 GaussDB(DWS) cluster. The client can be used for connection on the Huawei Cloud platform and over the Internet.

- Database users and roles

  GaussDB(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 GaussDB(DWS), the difference between roles and users is that a role does not have the **LOGIN** permission by default. In GaussDB(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 GaussDB(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 GaussDB(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.

- Tablespaces

  In GaussDB(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

  GaussDB(DWS) schemas logically separate databases. All database objects are created under certain schemas. In GaussDB(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.

- Transaction management

  In GaussDB(DWS), transactions are managed by multi-version concurrency control (MVCC) and two-phase locking (2PL). It enables smooth data reads and writes. In GaussDB(DWS), MVCC saves historical version data together with the current tuple version. GaussDB(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 GaussDB(DWS).

9 Related Services

IAM

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

Users who have the **DWS Administrator** permissions can fully utilize GaussDB(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 GaussDB(DWS) clusters.

ECS

GaussDB(DWS) uses an ECS as a cluster node.

BMS

GaussDB(DWS) uses a BMS as a cluster node.

VPC

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

OBS

GaussDB(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 GaussDB(DWS) clusters for analysis after the data is processed by Hadoop.

CDM

GaussDB(DWS) uses CDM to migrate data from multiple sources to GaussDB(DWS).

Cloud Eye

GaussDB(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

GaussDB(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

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

SMN

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

DNS

GaussDB(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.

10 Security

10.1 Shared Responsibilities

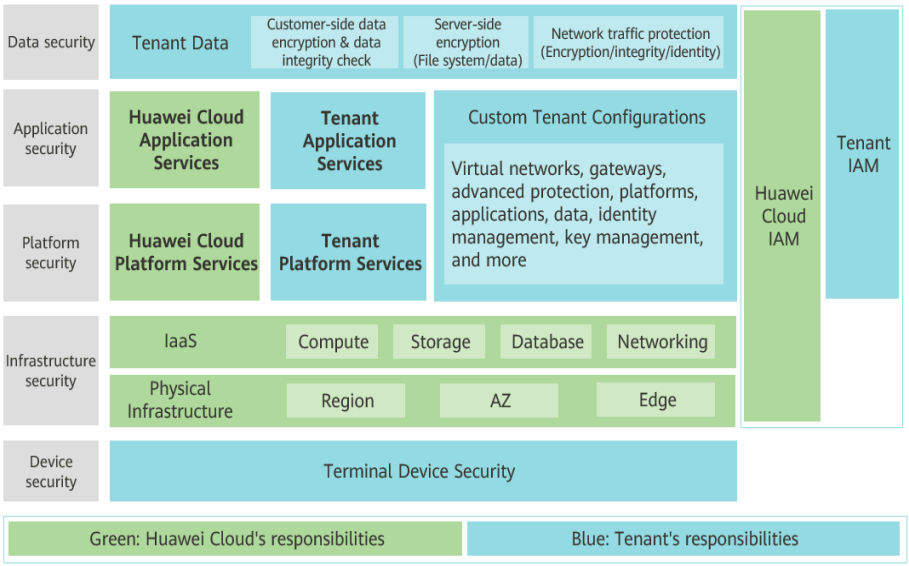
Huawei guarantees that its commitment to cyber security will never be outweighed by the consideration of commercial interests. To cope with emerging cloud security challenges and pervasive cloud security threats and attacks, Huawei Cloud builds a comprehensive cloud service security assurance system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

Figure 10-1 illustrates the responsibilities shared by Huawei Cloud and users.

- **Huawei Cloud**: Ensure the security of cloud services and provide secure clouds. Huawei Cloud's security responsibilities include ensuring the security of our IaaS, PaaS, and SaaS services, as well as the physical environments of the Huawei Cloud data centers where our IaaS, PaaS, and SaaS services operate. Huawei Cloud is responsible for not only the security functions and performance of our infrastructure, cloud services, and technologies, but also for the overall cloud O&M security and, in the broader sense, the security and compliance of our infrastructure and services.
- **Tenant**: Use the cloud securely. Tenants of Huawei Cloud are responsible for the secure and effective management of the tenant-customized configurations of cloud services including IaaS, PaaS, and SaaS. This includes but is not limited to virtual networks, the OS of virtual machine hosts and guests, virtual firewalls, API Gateway, advanced security services, all types of cloud services, tenant data, identity accounts, and key management.

Huawei Cloud Security White Paper elaborates on the ideas and measures for building Huawei Cloud security, including cloud security strategies, the shared responsibility model, compliance and privacy, security organizations and personnel, infrastructure security, tenant service and security, engineering security, O&M security, and ecosystem security.

Figure 10-1 Huawei Cloud shared security responsibility model



10.2 Authentication and Access Control

10.2.1 Resource Access Control (IAM Permission Control)

If you want to give varying levels of access to your company's GaussDB(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 GaussDB(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 GaussDB(DWS) usage.
- **Scenario 2**: Allow employees to use only GaussDB(DWS) resources, but not the resources of other services. To this end, grant them only the permissions for GaussDB(DWS).

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

10.2.2 Separation of Database Access Permissions

In GaussDB(DWS), you can isolate workloads through database and schema configurations. Their differences are as follows:

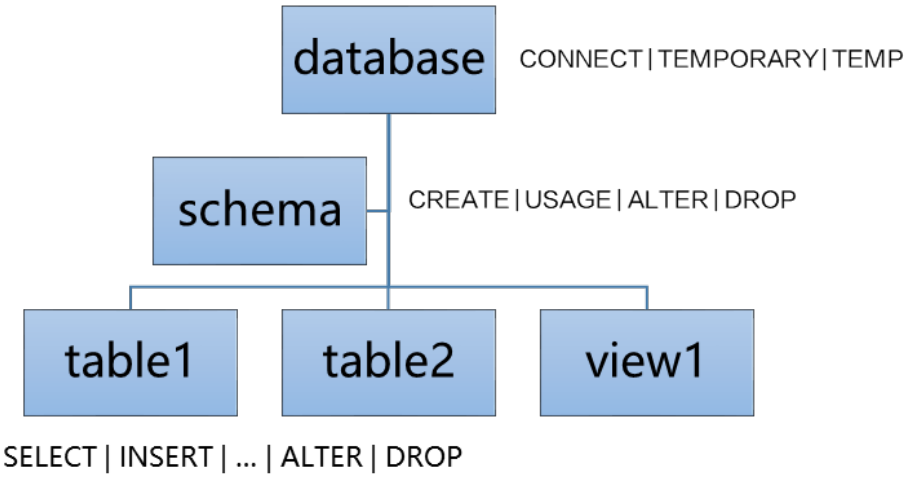
- Databases cannot communicate with each other and share very few resources. Their connections and permissions can be isolated.
- Schemas share more resources than databases do. User permissions on schemas and subordinate objects can be flexibly configured using the **GRANT** and **REVOKE** syntax.

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 GaussDB(DWS) Implement Workload Isolation?**

Figure 10-2 Permission levels



10.2.3 Permissions Management Using GRANT and REVOKE

Granting Permissions

GaussDB(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.

  GaussDB(DWS) grants the permissions for objects of certain types 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**.

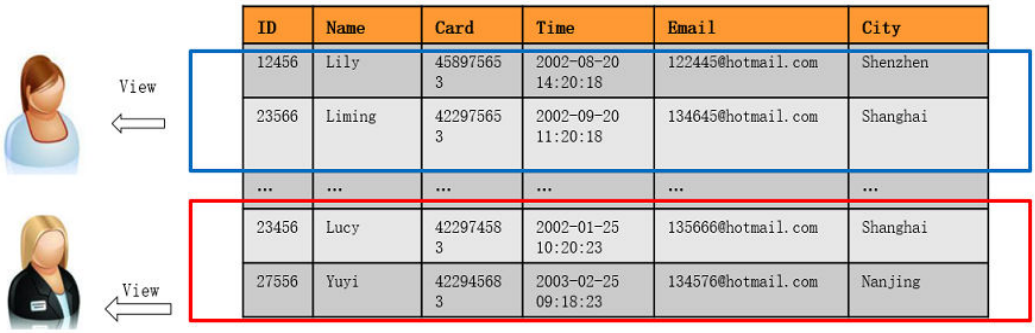
Revoking Permissions

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**.

10.2.4 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. GaussDB(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.



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**.

10.3 Cyber Security

Configuring Database Account Security

Account description

When setting up a GaussDB(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 GaussDB(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 GaussDB(DWS) Cluster O&M Account**.

Password complexity requirements

- The password of GaussDB(DWS) system administrator **dbadmin** is set when the cluster is created. The GaussDB(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 GaussDB(DWS) 2.0 Cluster with Coupled Storage and Compute**.
- 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 GaussDB(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.
- Maximum incorrect password attempts: **failed_login_attempts** sets the limit for the maximum number of incorrect password attempts. If this limit is exceeded, the account is automatically locked and can only be unlocked by the system administrator.
- Automatic account unlocking time: **password_lock_time** specifies the duration after which an account is automatically unlocked if it has been locked.

You can set these parameters on the GaussDB(DWS) console. For details, see **Modifying GUC Parameters of the GaussDB(DWS) Cluster**.

Resetting a password

- If the password of system administrator **dbadmin** is locked or forgotten, you can reset it using the GaussDB(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 GaussDB(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 GaussDB(DWS) security group to ensure that only trusted source IP addresses can access your DB instances.

When setting up a GaussDB(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.

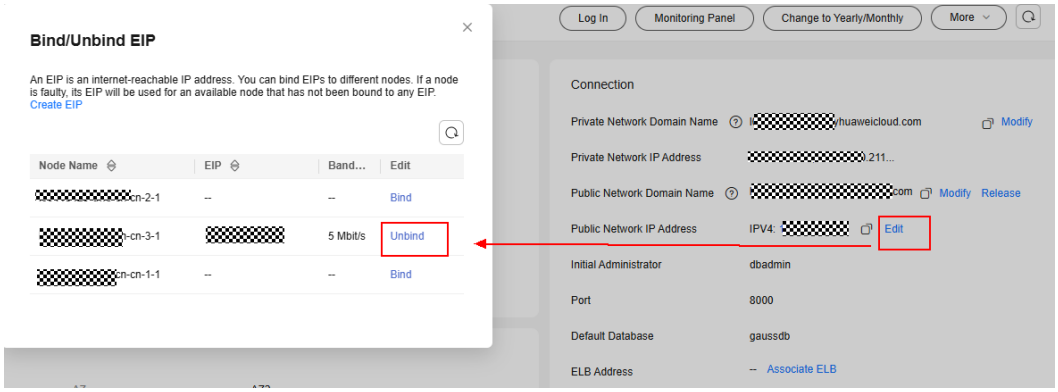
If you do not require public network access, it is best to consider the database as an internal component and use an internal IP address for accessing it. In this case, it is advised to detach the EIP from the database.

Unbinding procedure

Step 1 Log in to the GaussDB(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**.



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

GaussDB(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, GaussDB(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 GaussDB(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.

10.4 Data Protection Technologies

10.4.1 Data Backup

Data in GaussDB(DWS) can be backed up and restored using snapshots. A snapshot is a full or incremental backup of a GaussDB(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. GaussDB(DWS) allows you to manually create snapshots on the management console. It can also automatically create snapshots as scheduled to prevent data loss.

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

10.4.2 Transparent Data Encryption

GaussDB(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 GaussDB(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. GaussDB(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. GaussDB(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.

GaussDB(DWS) supports database-level transparent encryption. Each GaussDB(DWS) cluster has a CEK, and each database is separately configured with a DEK for encryption. The DEK is encrypted using the CEK and stored in the GaussDB(DWS) cluster. 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.

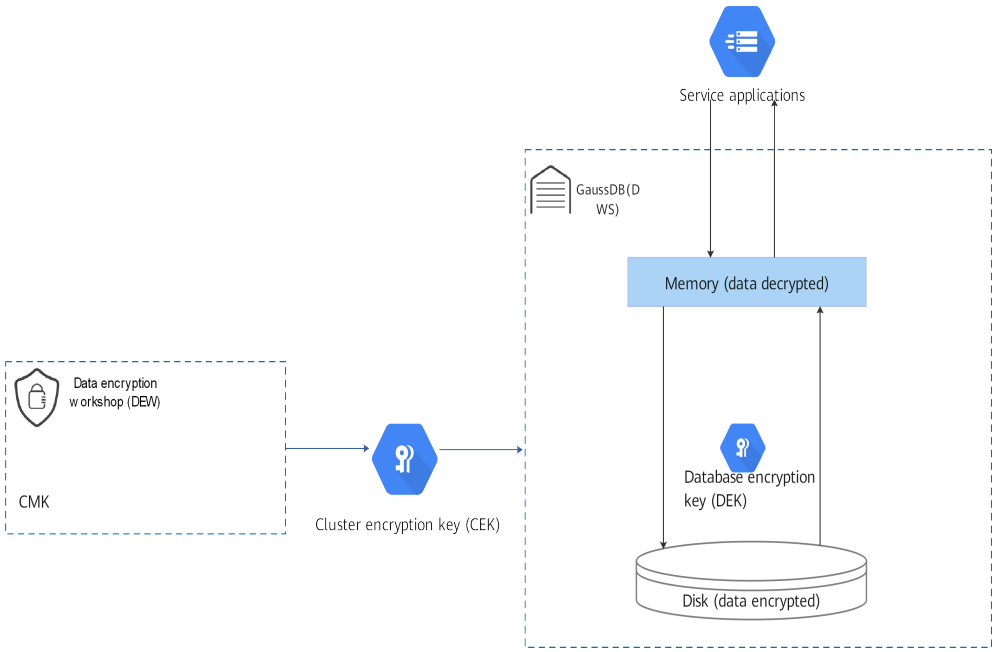
Currently, database-level transparent encryption is supported. You need to configure encryption when creating a cluster.

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 10-3 TDE encryption and decryption principles



10.4.3 SSL-encrypted Data Transmission

GaussDB(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, GaussDB(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 GaussDB(DWS) by default.

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

10.4.4 Data Redaction

GaussDB(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.

For details, see **Data Redaction**.

10.4.5 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 GaussDB(DWS) data warehouse also provides data encryption functions, including transparent encryption and encryption using SQL functions.

GaussDB(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.

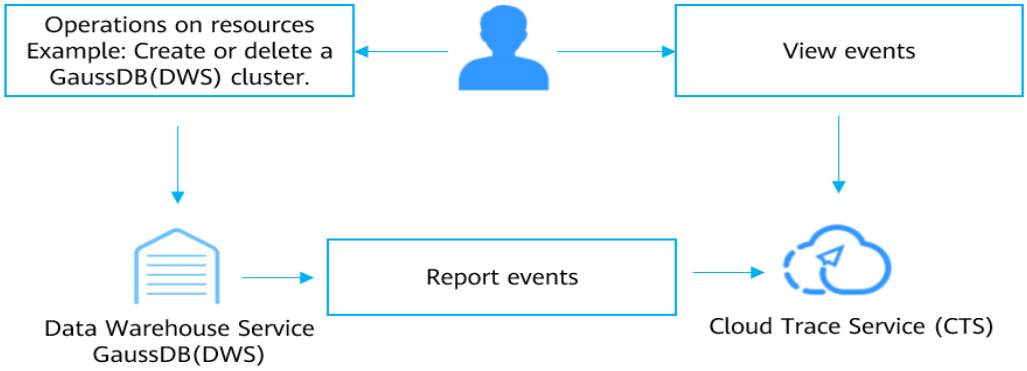
For details, see **Encrypting and Decrypting Data Columns**.

10.5 Audit and Logging

Operation Logs on the Management Console

Cloud Trace Service (CTS) records GaussDB(DWS) operation logs on the management console, such as creating or deleting GaussDB(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 GaussDB(DWS) for auditing.



For more information, see **Management Console Audit Logs**.

Database Audit Logs

In GaussDB(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 GaussDB(DWS).

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

10.6 Service Resilience

Security Hardening on the Management Plane

- Tomcat hardening: In the container images on the GaussDB(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: GaussDB(DWS) has preset security parameters on underlying VMs to enhance the OS security of ECS and BMS.

Isolation Between the Database and External Networks

GaussDB(DWS) is deployed in an independent VPC, which is isolated from other VPCs. Regarding firewall security zones, GaussDB(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

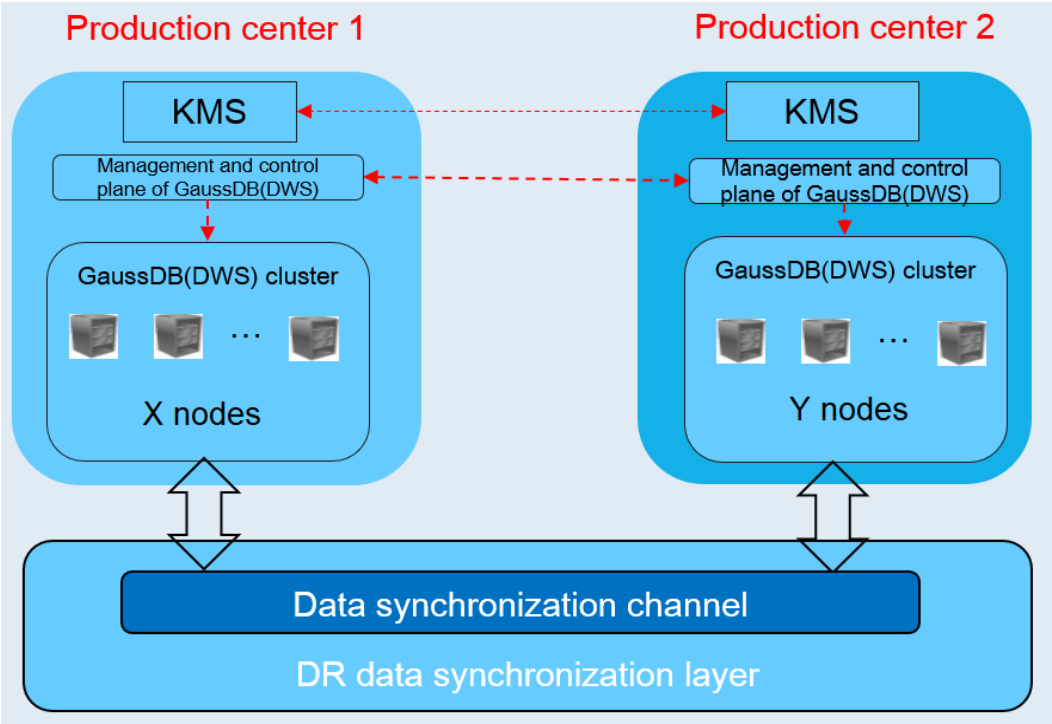
Cluster high availability (HA) is a practice of write ahead logging (WAL), using mechanisms such as primary/standby data synchronization, switchover, and reconstruction for database instance recovery and self-healing. By doing this, data reliability and integrity, and more importantly, service continuity,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

Intra-Region DR Deployment

GaussDB(DWS) provides dual-cluster intra-region DR capabilities. A GaussDB(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**.



10.7 Risk Monitoring

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

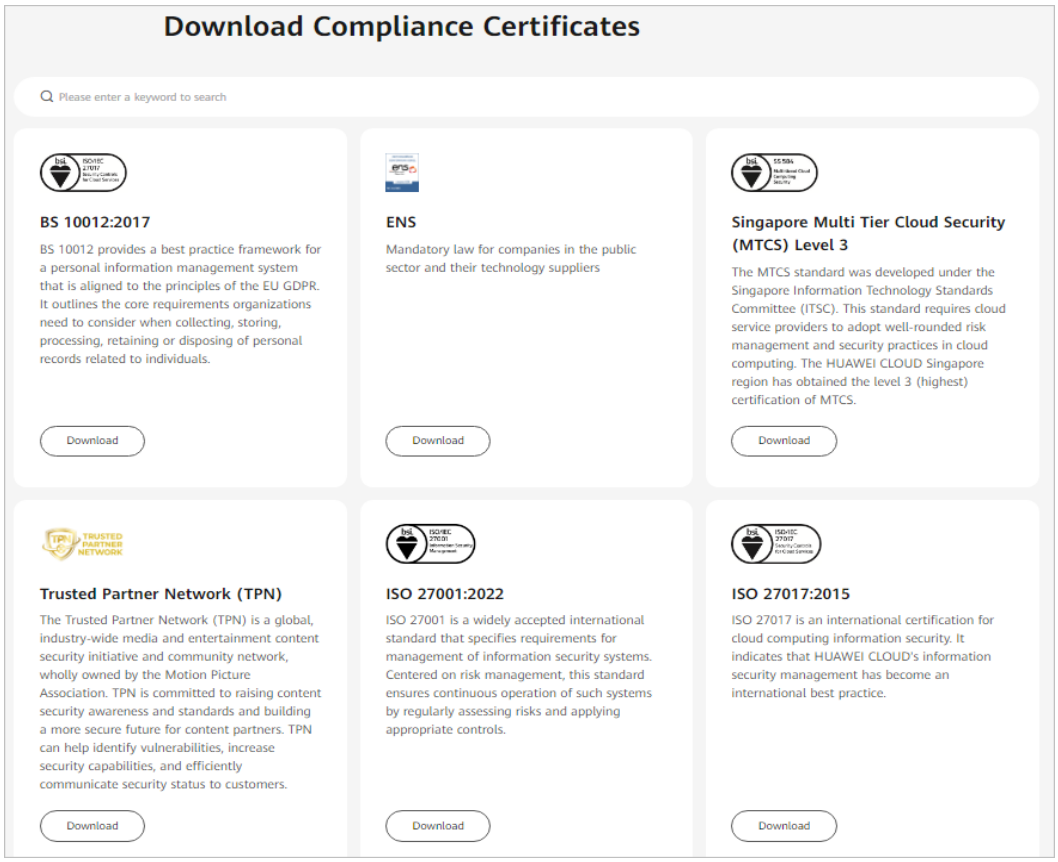
For details, see **Cluster Monitoring**.

10.8 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.

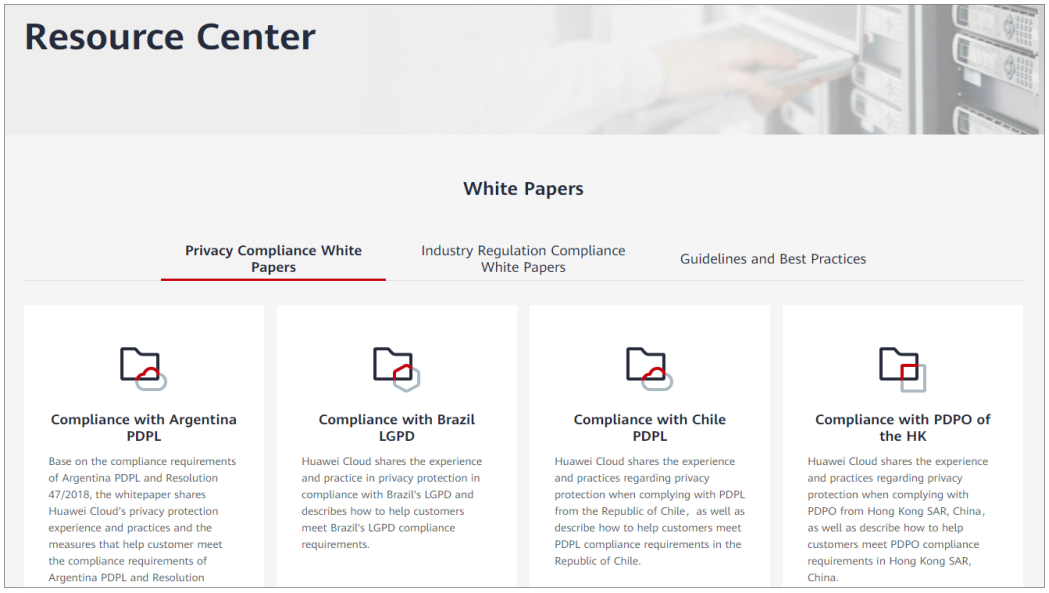
Figure 10-4 Downloading compliance certificates



Resource Center

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

Figure 10-5 Resource center



10.9 Security Notices

10.9.1 Vulnerability Fixing Description

Table 10-1 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

11 GaussDB(DWS) Permissions Management

If you need to assign different permissions to employees in your enterprise to access your GaussDB(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. To allow software developers in your company to use GaussDB(DWS) resources while restricting high-risk actions and preventing resource deletion, you can create IAM users tailored for these developers and grant them only the essential permissions for GaussDB(DWS) usage.

If your Huawei Cloud account does not need individual IAM users for permissions management, 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**.

GaussDB(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.

GaussDB(DWS) is a project-level service deployed and accessed in specific physical regions. To assign GaussDB(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 GaussDB(DWS), the users need to switch to a region where they have been authorized to use GaussDB(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 GaussDB(DWS) users only the permissions for managing a certain type of GaussDB(DWS) resources.

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

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

Table 11-1 lists all the system-defined roles and policies supported by GaussDB(DWS).

Table 11-1 GaussDB(DWS) system permissions

Role/Policy Name	Description	Category	Dependencies
DWS ReadOnlyAccess	Read-only permissions for GaussDB(DWS). Users granted these permissions can only view GaussDB(DWS) data.	System-defined policy	N/A
DWS FullAccess	Database administrator permissions for GaussDB(DWS). Users granted these permissions can perform all operations on GaussDB(DWS).	System-defined policy	N/A
DWS Administrator	Database administrator permissions for GaussDB(DWS). Users granted these permissions can perform operations on all GaussDB(DWS) resources. • Users granted permissions of the **VPC Administrator** policy can create VPCs and subnets. • Users granted permissions of the **Cloud Eye Administrator** policy can view monitoring information of data warehouse clusters.	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	GaussDB(DWS) database access permission. Users with this permission can generate the temporary database user credentials based on IAM users to connect to the database in the GaussDB(DWS) cluster.	System-defined role	Dependent on the **DWS Administrator** policy, which must be assigned in the same project as the **DWS Database Access** policy.

Table 11-2 lists the common operations supported by each system-defined policy or role of GaussDB(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 GaussDB(DWS) to access VPC. After the authorization is successful, GaussDB(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, the IAM users in those accounts cannot query or create agencies. When the users use the EIP, the system makes the binding function unavailable. Contact a user with the **DWS Administrator** permissions to authorize the agency on the current page.

Table 11-2 Common operations supported by GaussDB(DWS) system-defined permissions

Operation	DWS FullAccess	DWS ReadOnlyAccess	DWS Administrator	DWS Database Access
Creating/ Restoring clusters	√	x	√	x
Obtaining the cluster list	√	√	√	x

Operation	DWS FullAccess	DWS ReadOnlyAccess	DWS Administrator	DWS Database Access
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
Modifying DNS domain names	√	x	√	x

Operation	DWS FullAccess	DWS ReadOnlyAccess	DWS Administrator	DWS Database Access
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

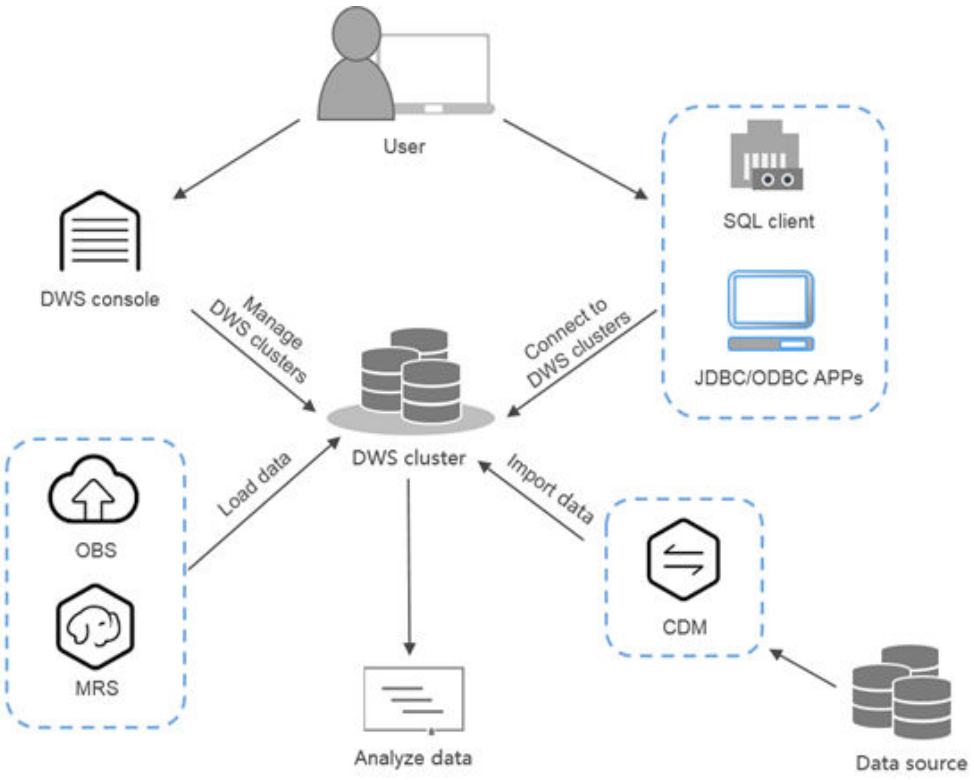
Helpful Links

- IAM Service Overview
- Creating a User and Granting GaussDB(DWS) Permissions
- Permissions Policies and Supported Actions

12 GaussDB(DWS) Access

The following figure shows how to use GaussDB(DWS).

Figure 12-1 Process for using GaussDB(DWS)



Accessing a Cluster

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

NOTE

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

Accessing the Database in a Cluster

GaussDB(DWS) supports database access using the following methods:

- GaussDB(DWS) client

  Use the GaussDB(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 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 GaussDB(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

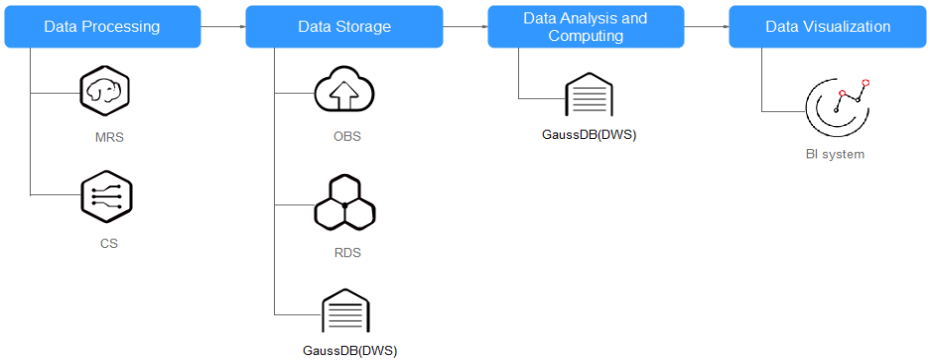
  Currently, GaussDB(DWS) does not support cross-database access. Schemas can be used to isolate resources. For details, see **CREATE SCHEMA**.

End-to-End Data Analysis Process

GaussDB(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 12-2 End-to-end data analysis process



13 Restrictions

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

After creating a GaussDB(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:

Table 13-1 GaussDB(DWS) constraints

Item	Constraint
Creating a Cluster	• 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. • After a cluster is created, its type cannot be changed. For details about the differences between product types, see **Data Warehouse Types**. • You are advised to purchase a cluster in yearly/monthly mode. After the monthly package expires, the cluster will automatically be released.
Connecting to a Cluster	• If you use a client to connect to a cluster, its VPC subnet must be the same as that of the cluster. • You can manage clusters only and cannot directly access nodes in a cluster. You can use a cluster's IP address and port to access the database in the cluster.

Item	Constraint
SQL Syntax Changes	• A storage-compute coupled data warehouse (standalone) does not support full-text search, OBS and HDFS foreign table import and export, automatic partition creation, sequence, and related functions. • A storage-compute coupled data warehouse (standalone) has only one DN. Therefore, the distribution rule is ignored and cannot be modified. • If you copy commands from the document to the operating environment, the text wraps automatically, causing command execution failures. To solve the problem, delete the line break. • You are advised to create databases as required. Do not use the default gaussdb database of a cluster.
Changing Flavor	• Only clusters of version 8.1.1.300 or later support flavor change. • Currently, only offline flavor chage is supported. The change takes about 10 minutes. • If your cluster is created using local disks or compute-storage integration, you cannot change the flavor of the cluster. If you need nodes with a higher flavor, create a new cluster. Elastic specification change is supported only for storage-compute coupled clusters with cloud SSDs. You can choose **Change node flavor** to enable elastic specification change. • The new flavor of a yearly/monthly cluster is billed in yearly/monthly mode by default. • If the cluster is billed in yearly/monthly mode, the **Change all specifications** option is not supported. • The new cluster does not incur charges before the change completes. The old cluster enters the retention period and will not incur charges after the change completes. • The **Change all specifications** option is available only for storage-compute coupled clusters, not for standalone systems. Logical clusters do not support classic changes. • A cluster can have up to 240 nodes. The old and new clusters can have up to 480 nodes in total. • Disk capacity expansion can be performed only for storage-compute coupled data warehouses with cloud SSDs. Only version 8.1.1.203 and later are supported. • Disk capacity can be expanded only if the cluster is in **Available**, **To be restarted**, **Read-only**, or **Node fault**, **Unbalanced** state.

Item	Constraint
Scaling Out	• If a cluster is billed in yearly/monthly mode, new nodes in the cluster will also be billed in this mode. • When you scale out a storage-compute coupled data warehouse cluster, use the same storage specifications as the cluster. • Nodes cannot be added to a storage-compute coupled data warehouse (standalone). • The cluster redistribution function is supported in 8.1.1.200 or later cluster versions. • This function can be manually enabled only when the cluster task information displays **To be redistributed** after scale-out. • Scale-in is supported only by clusters of version 8.1.1.300 and later. For clusters in yearly/monthly billing mode, the function is supported only in version 8.2.1 and later. • By default, scaled in nodes are charged by quantity. • When you scale in a storage-compute coupled data warehouse cluster, you can only modify the same storage specifications as used by the cluster. • A storage-compute coupled data warehouse (cluster mode) cannot be scaled in to a standalone cluster.
Backing Up a Cluster	• The new GaussDB(DWS) cluster created based on the snapshot must have the same configurations as the original cluster. That is, the number and specifications of nodes, memory, and disks in the new cluster must be the same as those in the original cluster. • If you create a new cluster based on a snapshot without modifying parameters, the parameters of the new cluster will be the same as those of the snapshot. • A storage-compute coupled data warehouse (standalone) does not support snapshots. • During snapshot creation, do not perform the VACUUM FULL operation, or the cluster may become read-only. • Snapshot creation affects disk I/O performance. You are advised to create snapshots during off-peak hours. • During the snapshot creation, some intermediate files are retained, which occupy extra disk space. Therefore, avoid peak hours and ensure that the disk capacity usage is less than 70%. • Snapshots can be restored to the current cluster, but logical clusters and resource pools cannot be restored to the current cluster.

Item	Constraint
Version Upgrade	• Cluster 8.1.1 and later versions allow users to deliver cluster upgrade operations on the console. • If the cluster is interrupted for a long time due to a node fault or system upgrade, contact technical support.
Data Migration	• Ensure that no Chinese characters are contained in paths used for importing data to or exporting data from OBS. • Data cannot be imported to or exported from OBS across regions. Ensure that OBS and the GaussDB(DWS) cluster are in the same region.
Failover	• When the DR task is created, the snapshot function of the production cluster is normal, but that of the DR cluster is disabled. Besides, snapshot restoration of both clusters is disabled. • DR does not synchronize data from external sources. • DR management refers to dual-cluster DR under the same tenant. • The DR cluster and the production cluster must be logically homogeneous and in the same type and version. • The production cluster and DR cluster used for intra-region DR must be in the same VPC. • In intra-region DR, after services are switched over from the production cluster to the DR cluster, the bound ELB is automatically switched to the new production cluster. During the switchover, the connection is interrupted for a short period of time. Do not run service statements to write data during the switchover. • During intra-region DR, the EIP, intranet domain name, and connection IP address of the original production cluster are not automatically switched with the cluster switchover. The EIP, domain name, or IP address used for connection in the service system need to be switched to the new cluster.

Item	Constraint
Hot and Cold Data Management	• A storage-compute coupled data warehouse (standalone) does not support cold and hot partition switchover. • Currently, cold and hot tables support only column-store partitioned tables of version 2.0. Foreign tables do not support cold and hot partitions. • Only hot data can be switched to cold data. Cold data cannot be switched to hot data. • A partition on a DN is either hot or cold. For a partition across DNs, its data on some DNs may be hot, and some may be cold. • Only the cold and hot switchover policies can be modified. The tablespace of cold data in cold and hot tables cannot be modified.

14 Technical Support

GaussDB(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, GaussDB(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 GaussDB(DWS) cloud services, including those used in application systems.

Technical Support Scope

- **Supported services**

  The GaussDB(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**
  - GaussDB(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.
  - GaussDB(DWS) does not troubleshoot or analyze job running exceptions if there are no visible problems or defects with its cluster component service.

15 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 15-1 shows the default user quotas of GaussDB(DWS). For how to view and increase quotas, see **Quotas**.

Table 15-1 Service quotas

Resource Type	Total Quota
Nodes	256

16 GaussDB(DWS) Technical Specifications

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

Table 16-1 Technical specifications of GaussDB(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 1024 DNs. It is recommended that you limit VWs to 32 or fewer and DNs to 128 or fewer per VW.

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: 5000

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 logical cluster. DWS decoupled storage and compute allows a physical cluster to be split into multiple VWs. Different services can be bound to different VWs to isolate service loads and increase the number of concurrent requests.

Table 16-2 Technical specifications of GaussDB(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.

