DataArtsFabric

Product Introduction

Issue	01
Date	2025-07-08

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.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: https://www.huaweicloud.com/intl/en-us/

Contents

1 What Is DataArts Fabric?

DataArts Fabric (DataArtsFabric) is Huawei Cloud's comprehensive, one-stop platform for data and AI development. It offers full lifecycle management, encompassing data processing, analysis, model fine-tuning, inference, deployment, and rollout. Data engineers, data scientists, and AI application developers can collaborate efficiently using familiar tools within a unified workbench, accelerating workflows from development to production. DataArtsFabric scales automatically to meet demanding application requirements, expanding resources incrementally based on actual needs. It saves up to 50% of costs compared to services with resource pools preset for peak loads.

This serverless approach, leveraging shared resource pools for diverse data and AI workloads (including CPU and NPU heterogeneous resources, and shared development/production environments), optimizes resource investment. It enables hybrid offline/online deployment and integrates training with inference, smoothing out peak and trough resource demands and significantly improving utilization. Customers benefit from a frictionless experience with zero resource thresholds, eliminating the need for cluster management and facilitating low-cost experimentation in dynamic business environments.

Architecture

DataArts Fabric offers a high-performance, highly reliable, low-latency, and cost-effective mass storage system. When integrated with Huawei Cloud big data services, it significantly reduces costs and simplifies big data management for enterprises.

- **SQL engine**

  DataArtsFabric's distributed SQL engine features layered decoupling of metadata services, computing, caching, and storage, enabling elastic resource allocation at each layer without impacting performance or availability. Statement-level elastic scaling and high-performance distributed analysis engines facilitate TB-level data queries in seconds and PB-level queries in minutes.

- **Distributed Ray**

  To overcome distributed computing challenges in data processing and ML/DL workloads, DataArtsFabric supports the Ray framework. This integration offers a unified workflow for data and machine learning engineering. DataArtsFabric

Ray's Ray-Data, Ray-Train, and Ray-Serve modules facilitate distributed data preprocessing, model training, and inference services.

- **Online inference**

  DataArtsFabric includes a proprietary, high-performance elastic inference engine. Users can deploy inference jobs via the default inference service or by independently deploying custom models.

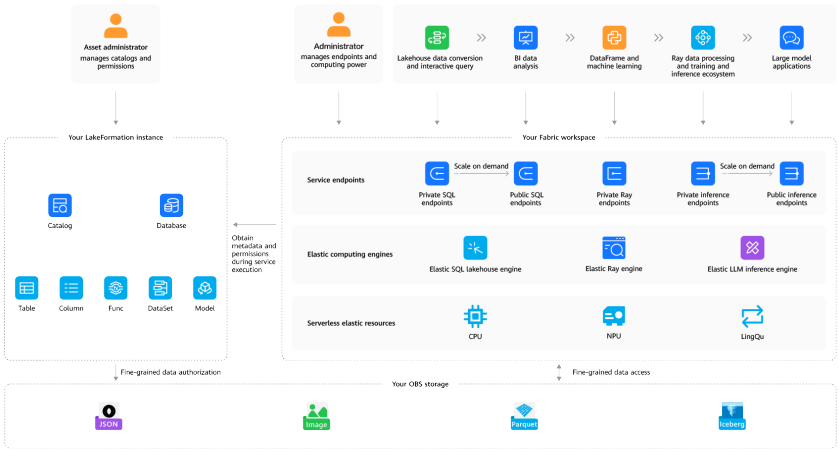
- **Heterogeneous resource management**

  DataArtsFabric offers unified management and allocation of CPU and NPU resources. Resource scheduling is supported at container-level or actor-level granularity. Furthermore, DataArtsFabric provides secure sandboxes for resource isolation and robust fault tolerance.

- **Multi-semantic cache acceleration**

  DataArtsFabric delivers cross-engine, multi-modal, and multi-semantic acceleration through various caching mechanisms, including data, model, and checkpoint caches.

Figure 1-1 Product architecture



Access Methods

DataArtsFabric offers multiple access methods:

a web-based management console, HTTPS-based APIs, and SDK clients for seamless compute engine integration.

- Management console

  DataArtsFabric can be accessed via its management console for managing Ray jobs, SQL jobs, model deployment, and model inference. This enables end-to-end data and AI development directly from the console.

- APIs

  For integrating DataArtsFabric into third-party systems or for secondary development, use the provided HTTPS APIs.

- SDKs

To integrate DataArtsFabric functionalities into third-party systems for secondary development, utilize the SDKs. DataArtsFabric SDKs encapsulate the REST APIs in Python and Java, simplifying development.

2 Advantages

DataArtsFabric has the following advantages:

One-stop data intelligence development with unified experience

- A single workspace supports multiple workloads, including SQL, Ray-based data engineering, and model inference.
- Structured, semi-structured, and unstructured data are uniformly managed through LakeFormation, ensuring consistent metadata and permission control across the entire data intelligence development process.
- Data and AI workloads share a single copy of data, eliminating the need for data replication.

Out-of-the-box, elastic, and on-demand resources

- Mainstream open-source third-party large model inference services are pre-configured. Customers can directly call preset inference service APIs for tasks like text dialogues without acquiring additional resources. Pay-per-use billing is supported.
- The inference service automatically scales to meet demand.
- Ray workloads benefit from pod-level automatic scaling, dynamically allocating resources to handle peak loads.
- SQL supports both resource-based and query-based billing, with rapid, query-level elasticity for computing resources.

Open-source ecosystem

- Open-source Ray capabilities are provided within the Ascend ecosystem, with high-reliability Redis integration.
- The Ray dashboard offers visualized monitoring, troubleshooting, performance tuning, and application runtime management.
- SQL supports data formats such as ORC, Parquet, and Iceberg, leveraging the open lakehouse ecosystem.

3 Scenario

This section describes the main application scenarios of DataArtsFabric.

- **Data engineering**

  Efficiently process data through parallel computing, accelerating tasks such as data cleansing, conversion, and aggregation.
- **Distributed machine learning**

  Leverage Ray for distributed training and tuning, enabling efficient model training with large-scale datasets and models.
- **Foundation models**

  Utilize large models to implement intelligent dialogues, automatic summarization, machine translation, text classification, and image generation.
- **Real-time data analysis**

  Perform queries and analysis on massive datasets using standard SQL APIs.

4 Features

4.1 DataArtsFabric SQL Features

Introduction to DataArtsFabric SQL

DataArtsFabric SQL is a fully managed data platform designed for superb elasticity and lakehouse capabilities. It leverages Huawei Cloud's robust infrastructure, featuring resource pooling and massive storage. This, combined with its unique architecture—including parallel execution, metadata decoupling, and compute-storage separation—delivers advanced Software as a Service (SaaS) technologies. Its serverless architecture empowers you to process complex business logic using SQL, eliminating the need for infrastructure management.

Built upon the Huawei Cloud DataArtsFabric platform, DataArtsFabric SQL's architecture comprises a service access layer, computing layer, and storage layer. This design ensures hierarchical decoupling and elasticity across metadata services, computing, cache, and storage. Each layer can dynamically allocate resources without impacting the performance or availability of others. Statement-level elastic scaling and high-performance distributed analysis engines facilitate TB-level data queries in seconds and PB-level queries in minutes.

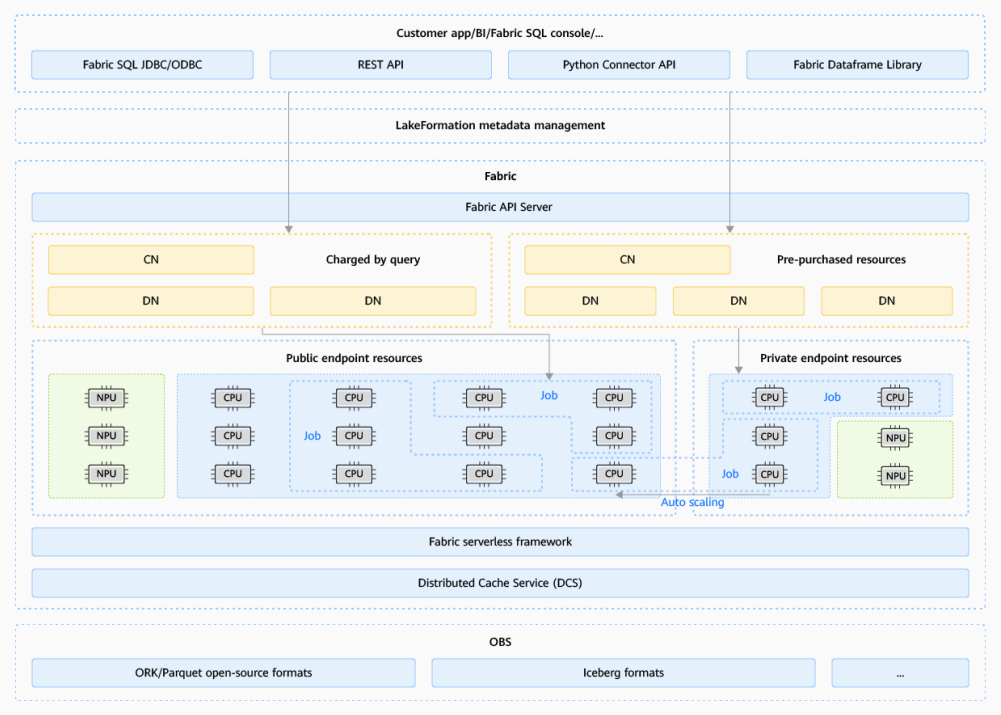
DataArtsFabric SQL supports processing and analyzing open structured data formats like Iceberg, ORC, and Parquet. It embraces the open lake ecosystem, enabling seamless data sharing when utilizing multiple data lake ecosystem services.

Embracing the Data+AI ecosystem, DataArtsFabric SQL offers a Python UDF feature. This allows users to execute Python scripts directly within SQL for one-stop AI data processing.

DataArtsFabric SQL provides a visualized interface and a JDBC driver for easy interaction with existing applications and third-party tools. Additionally, it offers REST and Python APIs, allowing developers to manage and transform data using familiar programming languages.

Product Architecture

Figure 4-1 DataArtsFabric SQL architecture



Functions

The following table describes the key functions of DataArtsFabric SQL.

Table 4-1 DataArtsFabric SQL features

Feature	Description
Standard SQL statements	Supports ANSI standard SQL, extended to include GBK, UTF-8, SQL ASCII, and Latin-1 character sets.
DDL	Supports CREATE, ALTER, DROP, SHOW, and DESCRIBE operations for schemas and tables.
Data types	Supports smallint, int, bigint, float, double, numeric, timestamp, date, varchar, char, bool, binary, and string.
APIs	Supports standard JDBC 4.0, RESTful APIs, and Python Connector APIs.
Transaction capabilities	Supports partition-level transactions and concurrency control. Iceberg offers full transaction capabilities.
Multi-tenant management	Tenants are isolated via dedicated CNs/DNs. Each CN/DN occupies an exclusive POD for isolation.

Feature	Description
Importing and exporting data	Data import and export use INSERT INTO. Data export from foreign tables supports format conversion (Parquet <-> ORC).
Scalability	Supports two-level elasticity. Elastic compute units (nodes) scale within seconds (less than 2s) in a resource pool based on query characteristics. Additional compute resources scale based on resource usage. **NOTE** This is a beta feature. Capacity expansion operations require the DataArtsFabric service to provision containers via the management plane.
SMP	Provides intra-node Symmetric Multi-Processing (SMP) for full utilization of multi-core CPUs. **query_dop** is enabled by default.
Vectorized execution	The vectorized execution engine enhances OLAP performance.
Statistics collection	ANALYZE collects statistics to improve optimizer accuracy and ensure stable, efficient database performance.
Storage formats	Supports Parquet, ORC, and Iceberg.
DML	Supports INSERT INTO and INSERT OVERWRITE.
Partitioned tables	Supports partitioned tables for Parquet, ORC, and Iceberg.
Views	Supports views.
User-defined functions (UDFs)	Extends SQL statements with user-defined functions for unified execution. Currently, only Python is supported.
Elastic computing scale	A single query supports up to 256 elastic compute nodes.
Fine-grained access control	Table metadata is managed by LakeFormation, utilizing IAM permissions. LakeFormation currently oversees overall permission control.

5 Product Specifications

DataArtsFabric SQL Specifications

The following table lists the DataArtsFabric SQL specifications.

Table 5-1 SQL specifications

Type	Specifications	Compute
DPU	fabric.sql.dcu.std	It offers approximately 1 CPU core and 4 GB of memory.

6 Permissions Management

If you need to assign different permissions to employees in your enterprise to access your DataArtsFabric resources, IAM is a good choice for fine-grained permissions management. This service provides identity authentication, permissions management, and access control, helping you to securely access your Huawei Cloud resources. If your Huawei Cloud account does not require IAM for permissions management, you can skip this section.

IAM can be used free of charge. You pay only for the resources in your account.

With IAM, you can assign permissions to control users' access to specific resources. For example, if you want some software developers in your enterprise to be able to use DataArtsFabric resources but do not want them to be able to delete resources or perform any other high-risk operations, you can create IAM users and grant permission to use DataArtsFabric resources but not permission to delete them.

DataArtsFabric supports role/policy-based authorization.

Table 6-1 Role/Policy-based authorization

Policy	Core Relationship	Permission	Authorization Method	Application Scenario
Role/Policy-based authorization	User-permission-authorization scope	• System-defined roles • System-defined policies • Custom policies	Assigning roles or policies to principals	To authorize a user, you need to add it to a user group first and then specify the scope of authorization. It provides a limited number of condition keys and cannot meet the requirements of fine-grained permissions control. This method is suitable for small- and medium-sized enterprises.

Assume that you want to grant IAM users permission to create ECSs in CN North-Beijing4 region A and OBS buckets in CN South-Guangzhou region B. With role/policy-based authorization, the administrator needs to create two custom policies and assign both to the IAM users. With identity policy-based authorization, you only need to create one custom identity policy, use the condition key **g:RequestedRegion** for the policy, and then attach the policy to the users or grant the users the access permissions to the specified regions. Identity policy-based authorization is more flexible than role/policy-based authorization.

For more information about IAM, see **IAM Service Overview**.

Role/Policy-based Authorization

DataArtsFabric supports role/policy-based authorization. By default, new IAM users do not have any permissions. You need to add a user to one or more groups, and assign permissions policies or roles to these groups. Users inherit permissions of the groups to which they are added. This process is called authorization. The users then inherit permissions from the groups and can perform specified operations on cloud services.

DataArtsFabric is a project-level service deployed and accessed in specific physical regions. When you set **Scope** to **Region-specific projects** and select the specified projects (for example, **cn-north-4**) in the specified regions (for example, **CN North-Beijing4**), the users only have permissions for resources in the selected projects. If you set **Scope** to **All resources**, the users have permissions for resources in all region-specific projects. When accessing DataArtsFabric, the users need to switch to a region where they have been authorized to use this service.

The following table lists all system-defined permissions of DataArtsFabric.

Table 6-2 System-defined permissions for DataArtsFabric

Role/Policy Name	Description	Category	Dependency
DataArtsFabric FullPolicy	Full permissions for DataArtsFabric.	System-defined policy	• IAM Agency Management FullAccess • OBS OperateAccess • LakeFormation ReadOnlyAccess • KMS Administrator (optional)

Role/Policy Name	Description	Category	Dependency
DataArtsFabricConsoleFullPolicy	All permissions for using DataArtsFabric on the console, including all permissions of DataArtsFabricFullPolicy and some permissions required on the console.	System-defined policy	• IAM Agency Management FullAccess • OBS OperateAccess • LakeFormation ReadOnlyAccess • IAM PolicyFullAccess • KMS Administrator (optional)
DataArtsFabricReadOnlyPolicy	Read-only permissions for DataArtsFabric.	System-defined policy	LakeFormation ReadOnlyAccess

The following table lists the common operations supported by system-defined permissions for DataArtsFabric. You can refer to this table to select the permissions as required.

Table 6-3 Common DataArtsFabric operations supported by system-defined permissions

Operation	DataArtsFabricConsoleFullPolicy	DataArtsFabricFullPolicy	DataArtsFabricReadOnlyPolicy
Listing workspaces	√	√	√
Creating a workspace	√	√	×
Modifying a workspace	√	√	×
Modifying workspace monitoring configuration	√	√	×
Deleting a workspace	√	√	×
Querying compute resources	√	√	√

Operation	DataArtsFabricConsoleFullPolicy	DataArtsFabricFullPolicy	DataArtsFabricReadOnlyPolicy
Creating a computing resource	√	√	×
Modifying a compute resource	√	√	×
Deleting a compute resource	√	√	×
Listing the endpoints of a workspace	√	√	√
Creating an endpoint for a workspace	√	√	×
Querying the endpoint details of a workspace	√	√	√
Modifying an endpoint of a workspace	√	√	×
Deleting an endpoint of a workspace	√	√	×
Listing jobs	√	√	√
Creating a job	√	√	×
Querying a job	√	√	√
Modifying a job	√	√	×
Deleting a job	√	√	×
Listing services	√	√	√
Creating a service	√	√	×
Modifying a service	√	√	×

Operation	DataArtsFabricConsoleFullPolicy	DataArtsFabricFullPolicy	DataArtsFabricReadOnlyPolicy
Querying a service	√	√	√
Deleting a service	√	√	×
Creating a model	√	√	×
Listing models	√	√	√
Querying a model	√	√	√
Deleting a model	√	√	×
Modifying a model	√	√	×
Creating a tag	√	√	×
Deleting a tag	√	√	×
Listing tags	√	√	√
Querying tags of a specific resource	√	√	√
Listing resources by tag	√	√	√
Creating a notification policy	√	√	×
Listing notification policies	√	√	√
Deleting a notification policy	√	√	×
Listing running jobs	√	√	√
Running a job	√	√	×

Operation	DataArtsFabricConsoleFullPolicy	DataArtsFabricFullPolicy	DataArtsFabricReadOnlyPolicy
Querying a running job	√	√	√
Deleting a running job	√	√	×
Canceling a running job	√	√	×
Invoking an inference service instance	√	√	×
Listing routes	√	√	√
Querying session information	√	√	√
Subscribing to a public endpoint	√	√	×

Role/Policy Dependencies of the DataArtsFabric Console

Table 6-4 Roles or policies that are required for performing operations on the DataArtsFabric console

Console Function	Dependency	Role/Policy Required
Granting service permissions	IAM	Granting permissions on the authorization page requires the IAM user to have the **IAM Agency Management FullAccess** policy.
Creating a workspace	LakeFormation	Users with the **DataArtsFabricFullPolicy** policy can create workspaces. Specifying a LakeFormation metastore during workspace creation requires the **LakeFormation ReadOnlyAccess** policy.
Creating a model	OBS	To create a model and specify its OBS file path on the model management page, an IAM user must have the **DataArtsFabricFullPolicy** and **OBS OperateAccess** policies.

Console Function	Dependency	Role/Policy Required
Creating a notification policy	IAM SMN	Creating a notification policy requires an IAM user to have the **DataArtsFabricFullPolicy**, **IAM Agency Management ReadOnly**, and **SMN ReadOnlyAccess** policies.

7 Constraints

7.1 Ray and XDS Restrictions

Large Model License Restrictions

Open-source large models are subject to varying license restrictions. Refer to the table below for details.

Table 7-1 Large model license restrictions

Model Name	License Address
Llama 3 8B Chinese Instruct	https://github.com/meta-llama/llama/blob/main/LICENSE
Llama 3 70B	https://github.com/meta-llama/llama/blob/main/LICENSE
Llama 3.1 8B Chinese Chat	https://huggingface.co/meta-llama/Meta-Llama-3.1-8B/blob/main/LICENSE
Llama 3.1 70B	https://huggingface.co/meta-llama/Meta-Llama-3.1-8B/blob/main/LICENSE
Qwen 2 72B Instruct	https://huggingface.co/Qwen/Qwen2-72B-Instruct/blob/main/LICENSE
Glm 4 9B Chat	https://huggingface.co/THUDM/glm-4-9b-chat/blob/main/LICENSE

Restrictions on the Common Inference Services

- Token quota: Each public inference service includes a free token quota. Once depleted, the service becomes unavailable, and additional tokens cannot be purchased. This quota is shared across all workspaces of the current user within the current region.

- Time restriction: Services are valid for 90 days. Upon expiration, the service becomes invalid. If the same inference service is enabled across multiple workspaces, the validity period commences from the first activation.
- Different models have varying context length restrictions.
- Service Level Agreements (SLAs) are not guaranteed. For enhanced performance, it is recommended to deploy a dedicated inference service.

7.2 DataArtsFabric SQL Restrictions

DataArtsFabric SQL Restrictions

Technical Metric	Max. Value
Number of concurrent sessions	500
Maximum length of the waiting queue for creating sessions asynchronously	1000

DataArtsFabric SQL Service Usage Restrictions

Table 7-2 DataArtsFabric SQL service usage restrictions

Item	Description
Service enabling	An account can enable only one DataArtsFabric SQL service, which is shared among all sub-users.
Connection operations	The service supports multiple connection modes, including SQL editor, JDBC, SDK, and API.
O&M operations	The service operates in serverless mode, eliminating the need for O&M tasks such as capacity expansion, upgrades, backup and restoration, or disaster recovery.
Timeout limit	The user authorization token lasts for 8 hours. Requests made after this period will fail to execute.
SQL syntax	For details, see **Database Operation Restrictions**.

Database Operation Restrictions

Table 7-3 Database operation restrictions

Type	Syntax	Supported
Basic functions	CREATE EXTERNAL TABLE	Yes

Type	Syntax	Supported
	DROP TABLE	Yes
	CREATE VIEW	Yes
	DROP VIEW	Yes
	INSERT	Yes
	SELECT	Yes
	TRUNCATE	Yes
	EXPLAIN	Yes
	ANALYZE	Yes
	ALTER TABLE DROP PARTITIONS	Yes
	ALTER TABLE SET TABLEPROPERTIES	Yes
	ALTER TABLE UNSET TABLEPROPERTIES	Yes
	ALTER TABLE DROP COLUMNS	No (supported only by Iceberg)
	ALTER TABLE ADD COLUMNS	No (supported only by Iceberg)
	ALTER TABLE COLUMN RENAME	No (supported only by Iceberg)
	CREATE EXTERNAL TABLE AS	Yes

