

CodeArts Artifact

Service Overview

Issue 01
Date 2026-04-03



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2026. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI 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 CodeArts Artifact Infographic.....	1
2 What Is CodeArts Artifact?.....	3
3 Product Advantages.....	6
4 Applications.....	8
5 Functions.....	9
6 Security.....	11
6.1 Identity Authentication and Permission Management.....	11
6.2 Data Protection Technologies.....	11
6.3 Auditing.....	12
6.4 Service Resilience.....	12
6.5 Certificates.....	12
7 Permission Management.....	14
8 Constraints.....	16
9 Glossary.....	18

1 CodeArts Artifact Infographic



CodeArts Artifact

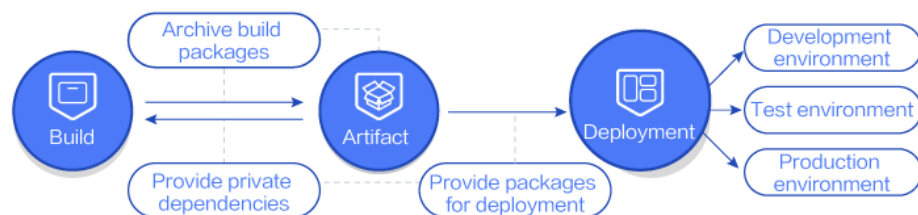


As software R&D scales, managing massive artifacts becomes complex and costly. Teams need efficient solutions to keep development on track. Therefore, a stable, efficient, secure, and reliable artifact repository is essential for software R&D.

What Is CodeArts Artifact?

CodeArts Artifact stores compiled binaries from source code. It provides trusted storage that underpins reliable software development.

CodeArts Artifact stores and manages packages in multiple formats. It serves as a source for deployments, provides private dependencies for builds, and bridges continuous integration and delivery.



How Is CodeArts Artifact Structured?

2 What Is CodeArts Artifact?

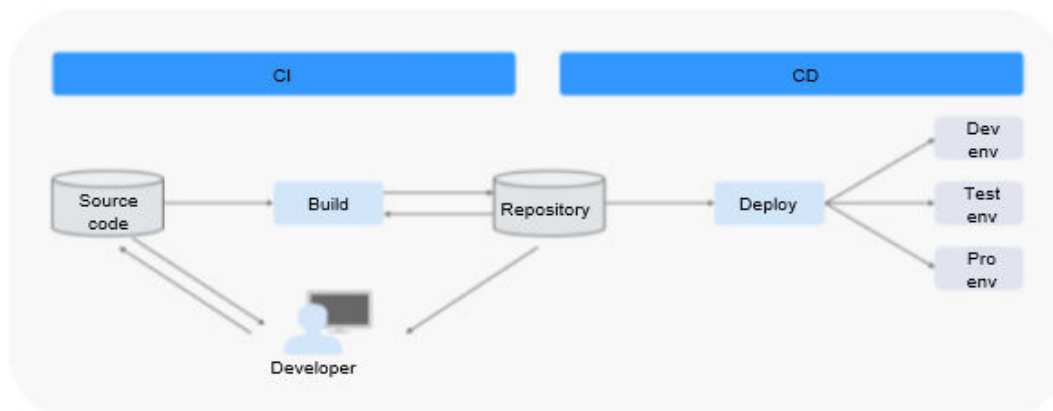
Service Overview

CodeArts Artifact helps software development enterprises manage the full lifecycle of software release process in a standardized, visualized, and traceable way.

CodeArts Artifact focuses on and manages the staging software packages (usually built from or packed from the source code) and their lifecycle metadata. The metadata includes basic properties, including the name and size, repository URLs, code branch information, build tasks, creators, and build time.

The management of **software packages** and their metadata is the basis of software release management. **Figure 2-1** shows the common software development process.

Figure 2-1 Software development process



Repository in **Figure 2-1** is CodeArts Artifact and is used to manage packages generated during software development. It plays a key role in connecting continuous integration (CI) and continuous delivery (CD). Operations such as software package release review, full-link tracing, and security control are all performed here.

CodeArts Artifact provides the following two types of repositories:

- **Release Repos**
A release repo stores software packages and tools in any formats.

Build products can be archived to release repos. You can view and manage the archived software packages and their lifecycle properties. These software packages are used for deployment.

- Self-Hosted Repos

A self-hosted repo manages private packages (such as Maven) across various development languages.

Different development language components have different requirements in the archive format (for example, the Maven component needs to be archived in GAV format). This repository is used to manage private development language components and share them with other developers in an enterprise or team.

What Functions Does CodeArts Artifact Provide?

Table 2-1 Release repos

Function	Description
Managing software packages	You can manage software packages in the same way as web disk files, including uploading, downloading, searching, deleting packages, and creating folders.
Querying software package properties	You can view the full lifecycle attributes of software packages in release repos. The lifecycle attributes include basic information (such as the name, size, and checksum), and build information (such as the build task, build time, and source code repository).
Uploading software packages to release repos using CodeArts Build	Release repos integrate CodeArts Build. Through configuration, all software packages generated by CodeArts Build can be automatically uploaded to release repos for archiving.
Integration with CodeArts Deploy	Software packages stored in release repos can be directly used by CodeArts Deploy for deployment scenarios.
Repo view and version view	You can view a software package in the repo view (storage directory structure) or version view (associated with build task and pipelines).

Table 2-2 Self-hosted repos

Function	Description
Managing private packages	You can manage private packages in the same way as web disk files, including uploading, downloading, searching, and deleting packages.

Function	Description
Releasing packages to the self-hosted repo using CodeArts Build	In a build task, you can configure rules to directly release build products to a self-hosted repo.
Connecting the local development environment	You can generate a configuration file in one click. After the generated file is configured in your local development tool, you can directly connect your local development environment to the private packages in the self-hosted repo. You can also use command lines to upload and download packages in the self-hosted repo.
Managing repository permissions	Administrators can assign roles to members to restrict their operation permissions (such as upload, download, and deletion) in self-hosted repos.

3 Product Advantages

Huawei Cloud CodeArts Artifact enriches artifact repository management in common languages by offering custom proxy and virtual repositories, artifact lifecycle management, and efficient check and search. It will continue to provide customers with comprehensive, efficient, and trusted artifact management.

In-House, Secure Artifact Repository with High Performance for Service Continuity

CodeArts Artifact is developed based on the cloud native architecture to resolve service continuity issues caused by external uncontrollable factors. Huawei Cloud CodeArts Artifact has the following features:

- **Security:** CodeArts Artifact provides multi-dimensional and fine-grained permission control to meet access control requirements of different roles in an enterprise. It uses the cloud native architecture for physical isolation to reduce the risk of malicious artifact theft.
- **Traceability:** records user operations.
- **Reliability:** Huawei Cloud CodeArts Artifact supports dual-AZ DR and cross-region DR, API traffic limiting and degradation, service dependency and isolation, and automatic service fault detection. These features allow a 99.99% SLA.
- **Ultimate Speed:** CodeArts Artifact provides cache acceleration for popular files, incremental upload and download, and full use of cache acceleration advantages for large and small files to improve the build speed, break through the underlying storage bandwidth limit, and implement high-speed concurrent transmission in the same region. Compared with similar open source products, CodeArts Artifact upload performance is improved by 5 times and the download performance by 10 times.

Over 10 Repository Types to Meet Your Needs

Huawei Cloud CodeArts Artifact supports mainstream artifact types in Generic, Docker, Maven, npm, Go, NuGet, PyPI, Conan, Debian, RPM, and more, meeting the requirements of embedded, web, and mobile application development scenarios. It can seamlessly integrate with on-premises builds, deployment tools,

and CI/CD on the cloud. Huawei Cloud CodeArts Artifact also provides artifact and metadata integrity verification capabilities. It supports fine-grained control and version-based package locking permissions to ensure the integrity of software release tests and comprehensively protect enterprise artifact security.

Third-Party Repository Connection and Unified Virtual Repository URLs

In scenarios where users use multiple image sources or artifact repositories at the same time, CodeArts Artifact provides repository aggregation, allows flexible aggregation of multiple proxy repositories, and provides a unified Artifact homepage to allow users to easily find artifact packages and simplify configurations.

The support for custom proxy repositories enables you to create proxies for open-source repositories and third-party dependency repositories. After files are downloaded from the proxy repositories, they can be cached to CodeArts Artifact, allowing downloading files from third-party dependencies as fast as from the local repositories.

Search by File Name and Checksum, Locating Hundreds of Millions of Artifacts in Seconds

Huawei Cloud CodeArts Artifact has powerful search capabilities by using data engine. You can search artifacts quickly from tens of billions of artifact files from multiple dimensions.

It covers multiple package types, such as Maven, npm, Go, PyPI, RPM, Debian, Conan and NuGet. You can search for and locate artifacts in seconds by file name or hash information (MD5, SHA1, SHA256, and SHA512). Based on this, CodeArts Artifact also supports efficient query associating hundreds of millions of metadata and SBOM to quickly trace artifact files. Compared with similar open-source products, CodeArts Artifact improves the search performance by 20 times.

4 Applications

DevOps and CI/CD

The development team often relies on third-party software packages during compilation and builds. CodeArts Artifact serves as a reliable, centralized storage hub for software packages. The build tool searches for and downloads the required dependencies from CodeArts Artifact based on the project settings. This process mitigates security risks from untrusted sources, enhances download efficiency, and reduces network-related download failures.

After the build, the generated software packages are stored in CodeArts Artifact. During deployment, you can directly pull the required packages. This enables quick package retrieval for application updates, deployments, and new environment setups, ensuring accuracy and efficiency throughout the process.

In addition, CodeArts Artifact manages software package versions, allowing team members to easily track changes and features in each version. This ensures that the correct version is selected during deployment, preventing errors caused by version mismatches. CodeArts Artifact significantly streamlines the build and deployment processes, providing strong support for DevOps practices.

Artifact Promotion

After coding and local tests, developers use a CI/CD tool to archive the generated artifacts to the repository in the development environment. Once the artifact passes the unit tests, the R&D team updates the version status and uses the replication function to push the artifacts to the repository in the test environment. After the test team completes a comprehensive test, the qualified artifacts are pushed to the pre-release environment for final integration verification. After all checks are passed, these artifacts are pushed to the repository in the production environment for official release. The artifact promotion feature of CodeArts Artifact helps R&D teams easily identify artifact maturity and ensure high-quality delivery.

5 Functions

This section describes CodeArts Artifact's functions. You can query the regions supported by each function on the console.

Release Repos

A release repo is a collection of software artifacts. It can store any software packages and tools in any formats.

You can upload, download, edit, search for, and delete software packages, as well as create, edit, search for, and delete folders.

You can also view and edit software package details, including basic information, build data, and archive information. You can also modify the folder name, package name, status, and release version.

For details, see [Release Repos](#).

Uploading a Software Package to Release Repos

You can manually upload software packages to release repos for storage and management. For details, see [Uploading a Package](#).

Clearing Policies

Software packages or folders deleted from release repos are moved to the recycle bin, where you can manage them.

Release repos automatically clears files on a scheduled basis. You can set a clearing policy to move expired files to the recycle bin or permanently delete them after the specified retention period.

You can configure clearing policies by file type and package status.

For details, see [Clearing Policies](#).

Self-Hosted Repos

A self-hosted repo manages private packages, including Maven, npm, Go, PyPI, RPM, Debian, and Docker.

You can edit and delete repositories and manage user permissions.

You can also connect self-hosted repos to your local development environment to use private packages during local development. When setting up the connection, you can obtain the repository URL.

For details, see [Managing a Repository](#).

Uploading/Downloading Packages on the Self-Hosted Repo Page

You can upload multiple private packages to target repositories. For details, see [Uploading/Downloading Packages on the Self-Hosted Repo Page](#).

Uploading/Downloading Packages on the Client

You can upload and download private packages such as Maven, npm, PyPI, Go, RPM, Debian, Conan, Docker, and CocoaPods on the client. For details, see [Uploading/Downloading Packages on the Client](#).

Managing Packages

You can search for, download, delete, and add or remove private packages from your favorites. For details, see [Managing Packages](#).

Recycle Bin

Repositories and packages deleted from a self-hosted repo are moved to the recycle bin, where you can manage them. For details, see [Recycle Bin](#).

6 Security

6.1 Identity Authentication and Permission Management

Authentication

You can access CodeArts Artifact through the management console or APIs.

Before calling an API, you need to pass the Identity and Access Management (IAM) authentication and obtain the corresponding token to access the API.

Permission Management

CodeArts Artifact provides two types of repositories: Release Repos and self-hosted repo.

- Release Repos: The permissions can be customized for each role in projects. For details, see [Configuring Permissions 2.0](#).
- Self-hosted repo: The permissions are determined by user roles and repository roles. User roles are essentially IAM permissions. To authorize IAM permissions, an administrator needs to create IAM users, add them to user groups, and assign policies or roles to the user groups. Users in the user groups also obtain the corresponding permissions. The repository role can be assigned by a user with the tenant administrator user role. For details, see "Managing User Permissions" in [Managing a Self-Hosted Repo 2.0](#). For details about fine-grained permission management, see the permission list following "Managing User Permissions" in [Managing a Self-Hosted Repo 2.0](#).

6.2 Data Protection Technologies

CodeArts Artifact takes different methods and features to keep data secure and reliable. For details, see [Table 6-1](#).

Table 6-1 CodeArts Artifact data protection methods

Method	Description
Transmission encryption (HTTPS)	CodeArts Artifact uses HTTPS to secure data transmission.
Personal data protection	CodeArts Artifact records operation logs to prevent personal data leakage and secure personal data.
Privacy protection	CodeArts Artifact encrypts sensitive data such as repository passwords before storing them.

6.3 Auditing

Cloud Trace Service (CTS) 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 Cloud Artifact for auditing.

For details about how to enable and configure CTS, see [Enabling CTS](#).

6.4 Service Resilience

CodeArts Artifact uses multi-active stateless cross-AZ deployment and inter-AZ data disaster recovery (DR) to enable service processes to be quickly started and recovered if a fault occurs, keeping service continuous and reliable.

To achieve this, an isomorphic DR cluster is provided for CodeArts Artifact in a separate AZ. In the event of a natural disaster affecting the production cluster's location, the DR cluster can take over.

If the production cluster fails due to internal faults, it cannot handle read and write requests. In such cases, the DR cluster assumes the role of the production cluster.

6.5 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 6-1 Downloading compliance certificates

Download Compliance Certificates

Please enter a keyword to search

BS 10012:2017

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

Download

ENS

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

Download

Singapore Multi Tier Cloud Security (MTCS) Level 3

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

Download

Trusted Partner Network (TPN)

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

Download

ISO 27001:2022

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

Download

ISO 27017:2015

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

Download

Resource Center

Huawei Cloud also provides the following resources to help users meet compliance requirements. For details, see [Resource Center](#).

Figure 6-2 Resource center

Resource Center

White Papers

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

Compliance with Argentina PDPL

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

Compliance with Brazil LGPD

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

Compliance with Chile PDPL

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

Compliance with PDPO of the HK

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

7 Permission Management

CodeArts Artifact provides a permission model that contains the tenant and project levels.


The application scope of this model is as follows: tenant-level permissions > project-level permissions.

If the permission configuration in this model conflicts, this permission priority is used: project-level > tenant-level.

Tenant-Level Permissions

These permissions take effect for all projects in your account, including creating, deleting, and modifying projects, and creating workspaces.

By default, users with the **Tenant Administrator** permission have tenant-level permissions. You can also grant project permissions to users without this permission. After logging in to the CodeArts homepage with an authorized account, perform the following steps to authorize another account:

- Step 1** Log in to CodeArts homepage and click the username in the upper right corner of the navigation bar.
- Step 2** Choose **All Account Settings**.
- Step 3** Choose **General > Project Creators**.
- Step 4** Select **Set some members to be able to create projects**. The member list is displayed.
- Step 5** Enable or disable the authorization by clicking . Unauthorized members cannot create projects.

----End

Project-Level Permissions

These permissions take effect for the current project, including editing and archiving projects, configuring roles and permissions, and configuring members. You can also configure operation permissions for each service, including the permissions to create, submit, and copy raw requirements in CodeArts Req, and

the permissions to commit and merge code in CodeArts Repo. These permissions take effect for all instances of the service.

CodeArts provides role-based access control (RBAC). By default, new users do not have permissions assigned. You need to add a user to a project, and assign roles to the user. The user then has the permissions specified in the roles and can perform specified operations on cloud services based on the permissions.

CodeArts provides 11 system roles for R&D processes, such as IPD and DevOps. You can also create custom roles and assign them different permissions.

Table 7-1 Built-in project roles in CodeArts

Role Name	Description
Project administrator	The general owner of a project who manages all settings and members of the project, including creating, deleting, and modifying projects, assigning and canceling permissions of other roles.
Project manager	A primary owner of a project who manages requirements, plans, progress, and risks of the project, and coordinates work in the project team.
Product manager	Responsible for product design and planning, including defining product requirements, prototypes, and user stories, communicating and collaborating with developers and testers.
Test manager	Responsible for testing, including managing test plans, test cases, test execution, and bug tracking, guiding and supervising test personnel.
O&M manager	Responsible for project O&M, including project deployment, monitoring, and fault locating and rectification.
System engineer	Responsible for the system architecture and infrastructure of a project, including designing, setting up, and maintaining project resources such as servers, networks, and databases.
Committer	Reviews and merges code committed by developers.
Developer	Writes, commits, merges, and branches code, creates and runs services such as CodeArts Pipeline and CodeArts Build.
Tester	Executes test cases, reports bugs, and verifies fixes.
Participant	Participates in a project and creates work items.
Viewer	Views a project and cannot perform any operations in any services.

8 Constraints

Table 8-1 describes the constraints on CodeArts Artifact.

Table 8-1 Constraints

Category	Item	Limit
Browser	Type	The following browsers are supported: <ul style="list-style-type: none">• Chrome: The latest three stable versions are supported and tested.• Firefox: The latest three stable versions are supported and tested.• Edge: Windows 10 uses Edge by default. The latest three stable versions are supported and tested.• Internet Explorer is no longer supported and tested. Chrome and Firefox are recommended.
Resolution	Resolution	(Recommended) 1280 x 1024 or higher
Total storage capacity	Capacity of Release Repos and self-hosted repos	Total capacity: 10 GB
Total download capacity	Traffic of Release Repos and self-hosted repos	Total traffic: 5 GB/month
Constraints on Release Repos	Maximum size of a file uploaded on the console	2 GB
	Maximum size of a file uploaded by a build task	10 GB

Category	Item	Limit
Constraints on self-hosted repos	Maximum size of a file uploaded on the console	Maven/npm/PyPI/Go/RPM/Debian/ Conan: 100 MB NuGet: 20 MB
	Repository quantity	<ul style="list-style-type: none">• Max. 100 non-Maven repositories• Max. 50 pairs of Maven repositories
	Maximum size of a file uploaded by a build task	2 GB

 **NOTE**

- For details about the specifications of different CodeArts packages, see [Specifications](#).
- The current edition is CodeArts Free Edition.

9 Glossary

Before using the release repos and self-hosted repos of CodeArts Artifact, review the basic concepts.

Package

A package is a collection of source code files or build products. It groups related components that are distributed together to provide specific functionality. Packages are typically stored in binary or compressed formats. Managing packages and their metadata is the foundation of release management.

Release Repos

A release repo is a general-purpose artifact repository that stores and manages artifacts in various formats. In addition to basic storage functions, it also supports important functions such as build and deployment tool integration, version control, and access control. It provides a standardized way for enterprises to manage all artifact types generated during software development.

Self-Hosted Repos

A self-hosted repo manages private packages (often called private repositories). It provides a unified way to manage packages generated during software development. Different programming languages use different package formats. Developers often need to share packages with their team during development. Self-hosted repos serve as a central place where these packages can be stored and accessed by others, making it easy for team members to obtain packages from repositories. This enables secure management and sharing of packages within an organization.

Local Repository

A local repository is a type of self-hosted repos. It is a physical repository where artifacts are fully hosted on the CodeArts Artifact server. You can upload various artifact types to a local repository to store and manage your build outputs, dependencies, and third-party artifacts.

This repository provides the following features:

- **Privacy:** The repository is fully controlled by you and is not shared with external systems.
- **Durability:** Data is permanently stored on the CodeArts Artifact server.
- **Performance:** Fast access speeds.
- **Usage:** It is suitable for storing self-developed components, binary files, and configuration files.

Virtual Repository

A virtual repository is a type of self-hosted repos. You can configure proxy sources in a virtual repository to connect to local third-party repositories. It also offers local repository functions and provides a unified entry to make setup easier for you.

This repository provides the following features:

- **Aggregation:** A virtual repository can include multiple local repositories.
- **Single URL:** Resources in multiple repositories can be accessed through a single logical URL.
- **Transparency:** Clients do not need to know which repository stores the resources.
- **Flexibility:** Repositories can be added or removed as needed.
- **Usage:** It is used to simplify dependency management and build processes, and provides a unified view.

Project Object Model

A project object model (POM) is the basic unit of work in a Maven project. It is an XML file that defines project configuration, build instructions, and dependencies. It also manages project metadata, including source code structure, configuration files, developer and organization information, access permissions, project URLs, and dependencies. When a build task is run, Maven searches for the POM in the current directory, reads its content, obtains the required configuration information, and builds the target package.