CodeArts Artifact

Service Overview

Issue	01
Date	2025-10-29

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 What Is CodeArts Artifact?

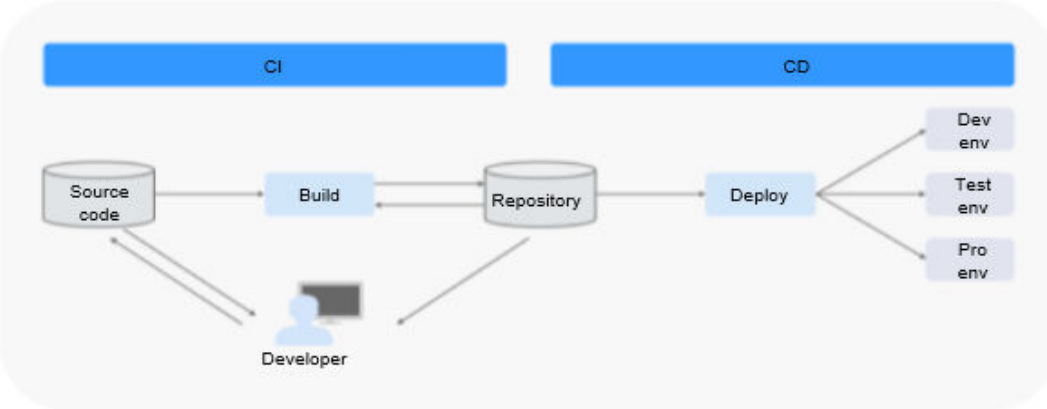
Service Overview

CodeArts Artifact helps software development enterprises manage the 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 such as the name and size, repository paths, code branch information, build tasks, creators, and build time.

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

Figure 1-1 Software development process



Repository in **Figure 1-1** is a collection of software artifacts and is used to manage packages generated during software development. It is an important link between continuous integration (CI) and delivery (CD). Operations such as release review, tracing, and security control of packages are usually performed in the repository.

CodeArts Artifact provides the following two types of repositories:

- Release Repos

  Release Repos can store any 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 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 packages vary in the archive format (for example, the Maven artifact needs to be archived in **GAV** format). CodeArts Artifact manages private packages and shares them with other developers in an enterprise or team.

What Functions Does CodeArts Artifact Provide?

Table 1-1 Release Repos

Function	Description
Managing software packages	You can upload, download, search for, and delete software packages. Folders can also be created for better management.
Querying software package properties	You can view the software package lifecycle properties in Release Repos. The lifecycle properties 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 integrates CodeArts Build. Through configuration, all software packages generated by CodeArts Build can be automatically uploaded to Release Repos for archiving.
Deployment	Software packages stored in Release Repos can be used by CodeArts Deploy.
Repo view and version view	You can view a software package in the repo view (storage directory structure) or version view (build task and pipeline).

Table 1-2 Self-hosted repos

Function	Description
Managing private packages	You can upload, download, delete, and search for private packages.

Function	Description
Releasing packages to the self-hosted repo using CodeArts Build	In a build task, you can configure build products to be directly released 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 the local development tool, you can directly connect the local development environment to the private packages in the self-hosted repo. For example, you can use command lines to upload and download packages in the self-hosted repo.
Managing repository permissions	By setting user roles in repositories, administrator can restrict operation permissions

2 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 checking and search. It will keep providing customers with comprehensive, efficient, and trusted artifact management.

Self-managed, Secure Artifact Repository with Optimal 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 Various Needs

Huawei Cloud CodeArts Artifact supports mainstream artifact types in Generic, Maven, npm, Go, PyPI, RPM, Debian, Conan, NuGet, 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.

Connection to Third-party Repositories and Unified Virtual Repository Path

In scenarios where users use multiple image sources or artifact repositories at the same time, CodeArts Artifact provides repository combination, allows flexible combination 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 users 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.

Searching 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. In a few seconds, you can search artifacts quickly from tens of billions of artifact files from multiple dimensions.

It covers multiple artifact 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.

3 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 Security Scanning

As open-source software is widely used, after the development team uploads artifacts containing open-source components to the repository, you can enable the artifact security scanning. This feature performs an in-depth scanning of the open-source software packages in the artifacts, comparing them against a vulnerability database to identify security risks. When a vulnerability is found, the system generates a detailed report, highlighting its location, severity, and recommended fixes. The development team can act on the report by updating open-source component versions or applying security patches, effectively mitigating risks from vulnerabilities. This ensures the security and stability of artifacts, providing a strong defense for future development, testing, and deployment.

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.

4 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 detailed, 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**.

5 Security

5.1 Shared Responsibilities

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

Unlike traditional on-premises data centers, cloud computing separates operators from users. This approach not only enhances flexibility and control for users but also greatly reduces their operational workload. For this reason, cloud security cannot be fully ensured by one party. Cloud security requires joint efforts of Huawei Cloud and you, as shown in **Figure 5-1**.

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

Figure 5-1 Huawei Cloud shared security responsibility model

Cloud security responsibilities are determined by control, visibility, and availability. When you migrate services to the cloud, assets, such as devices, hardware, software, media, VMs, OSs, and data, are controlled by both you and Huawei Cloud. This means that your responsibilities depend on the cloud services you select. As shown in **Figure 5-1**, customers can select different cloud service types (such as IaaS, PaaS, and SaaS services) based on their service requirements. As control over components varies across different cloud service categories, the responsibilities are shared differently.

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

5.2 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**.

5.3 Data Protection Technologies

CodeArts Artifact takes different methods and features to keep data secure and reliable. For details, see **Table 5-1**.

Table 5-1 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.

5.4 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**.

5.5 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.

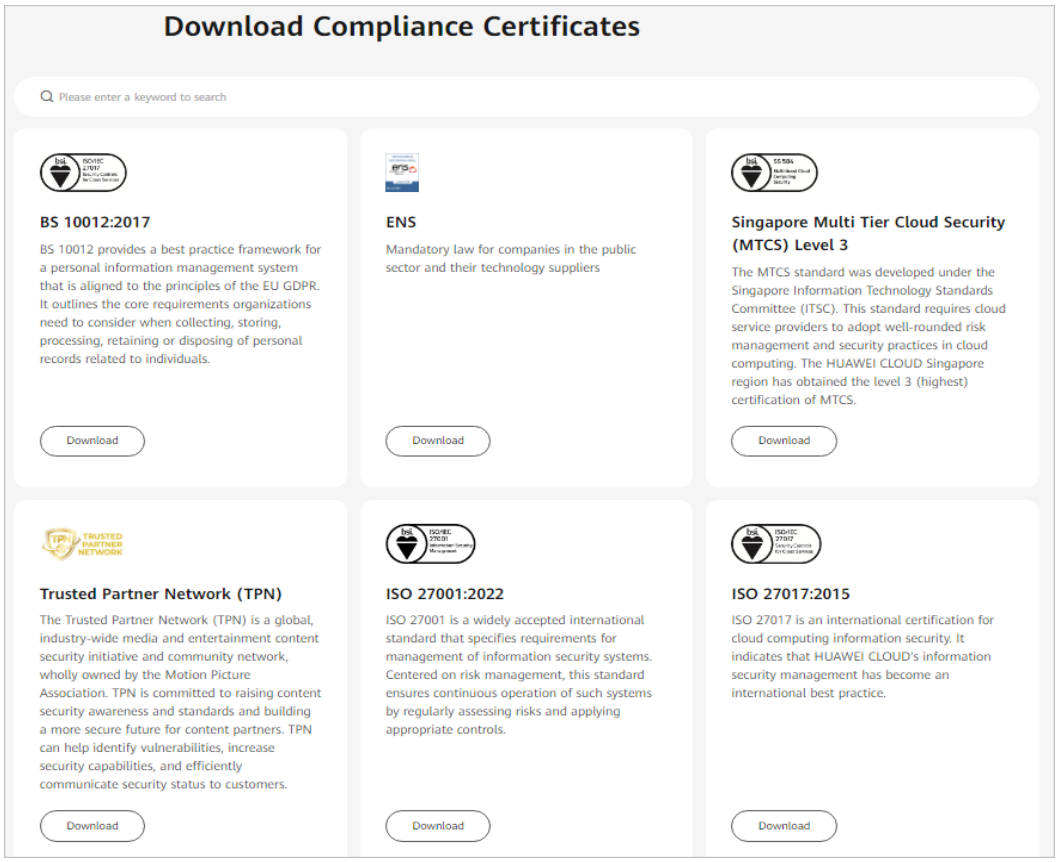
If the production cluster fails due to internal faults, it cannot handle read and write requests. In such cases, the DR cluster assumes the role of the production cluster.

5.6 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.

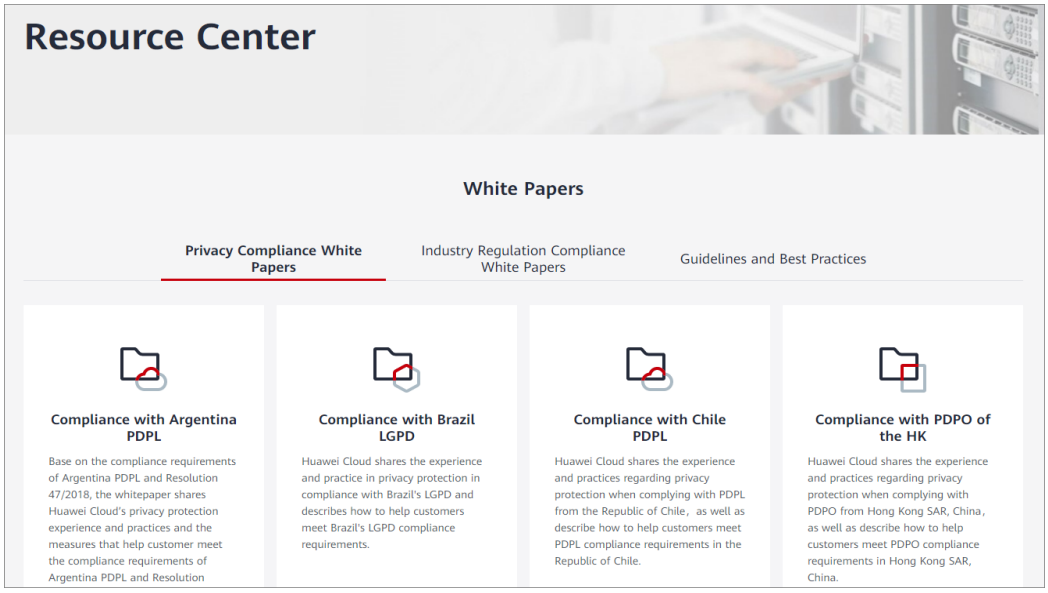
Figure 5-2 Downloading compliance certificates



Resource Center

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

Figure 5-3 Resource center



6 Permission Management

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

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

If the permission configuration in this model conflicts, this permission priority is used: instance-level > 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 6-1 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.

Instance-level Permissions

These permissions take effect for a specific repository or pipeline, for example, viewing, executing, editing, and deleting a pipeline.

Instance-level permissions are configured by the creator of the corresponding instance.

7 Constraints

Table 7-1 describes the constraints on CodeArts Artifact.

Table 7-1 Constraints

Category	Item	Limit
Browser	Type	The following browsers are supported: • 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	• 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.

8 Glossary

Table 8-1 Basic concepts

Glossary	Definition
Source code	A series of human-readable computer language instructions.
Software package	A software package is a collection of source code files or a compiled product. There are binary and compressed packages.
Release Repos	Release Repos is used to centrally manage various types of binary artifacts. It can seamlessly interconnect with existing software platforms that standardize build and release tools.
Self-hosted repo	A self-hosted repo is used to centrally manage private packages generated during software development. Binary file formats vary across programming languages. Private binary packages in an organization can be managed and shared through self-hosted repos.
Local repository	An actual physical repository hosted on the server, where you can upload different types of artifacts.
Virtual repository	You can configure proxy sources in the virtual repository to connect with local third-party repositories. It also offers local repository functions and provides a single entry to make setup easier.
Project template version file	A file that defines the project name, module name, and project version. The GAV upload of Maven repository means to upload private packages to a self-hosted repo. During the upload, you need to set *Groupid* (usually organizations, for example, **com.devops**), *artifactid* (usually modules or project names), and *version* to identify the package. This feature facilitates GAV-based download and use.

Glossary	Definition
Project object model	The **pom.xml** file describes the Maven coordinates of a project. This file is used to manage source code, configuration files, developer information and roles, organization information, project authorization, project URL, and project dependencies.

