

CodeArts PerfTest

Service Overview

Issue 01
Date 2026-02-12



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.

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 CodeArts PerfTest?	1
2 Advantages	4
3 Application Scenarios	6
4 Functions	10
5 Security	12
5.1 Authentication and Access Control	12
5.2 Data Protection	12
5.3 Audit and Logs	13
5.4 Resilience	13
5.5 Certificates	14
6 Permissions	16
7 Notes and Constraints	24
8 Related Services	28
9 Basic Concepts	31

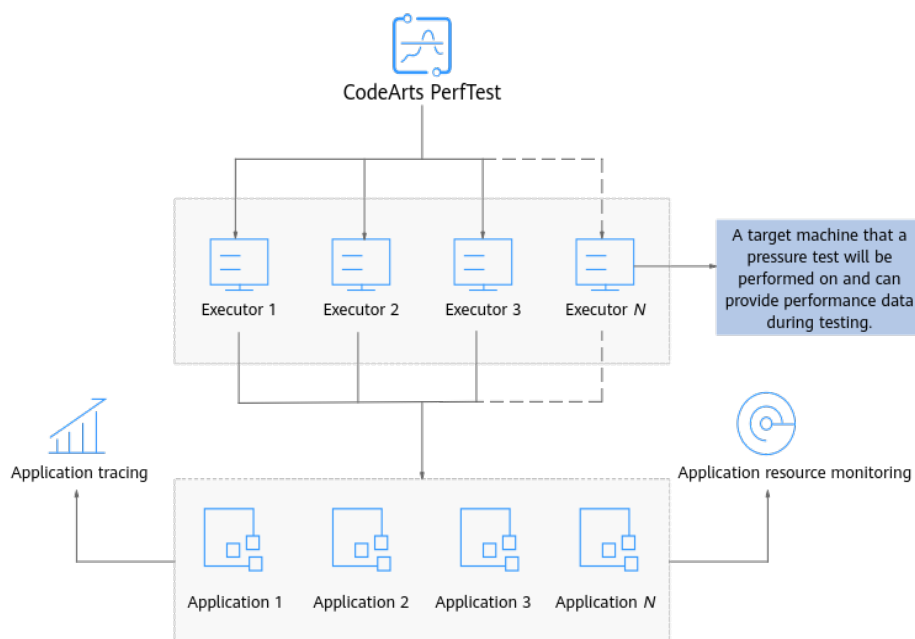
1 What Is CodeArts PerfTest?

Distributed architecture and microservice technologies have made applications more complex. This complexity results in architecture decoupling and performance improvements. However, it also makes it harder to locate performance problems in the production environment, and the repair periods become longer. Performance tests in advance of application launches are therefore necessary.

CodeArts PerfTest provides performance test services for cloud applications that are built based on HTTP, HTTPS, TCP, UDP, HLS, RTMP, WebSocket, MQTT, or HTTP-FLV. CodeArts PerfTest quickly simulates service peaks with a large number of concurrent users. It allows you to define the contents and time sequences of packets and supports complex combinations of multiple transactions. After tests are complete, CodeArts PerfTest provides professional test reports to evaluate your service quality.

CodeArts PerfTest simplifies performance pressure tests, helping you focus more on services and performance problems, reduce costs, enhance stability, optimize user experience, and improve the business value of enterprises.

Figure 1-1 CodeArts PerfTest



Functions

CodeArts PerfTest provides tests for HTTP/HTTPS/TCP/UDP/HLS/RTMP/WebSocket/HTTP-FLV/MQTT applications with high user concurrency. It allows you to flexibly define multi-protocol packet contents, transactions, and test task models. You can view performance statistics, such as concurrency, RPS, and response time during or after testing. You can also create private test clusters or scale in or out resource groups to test at different scales.

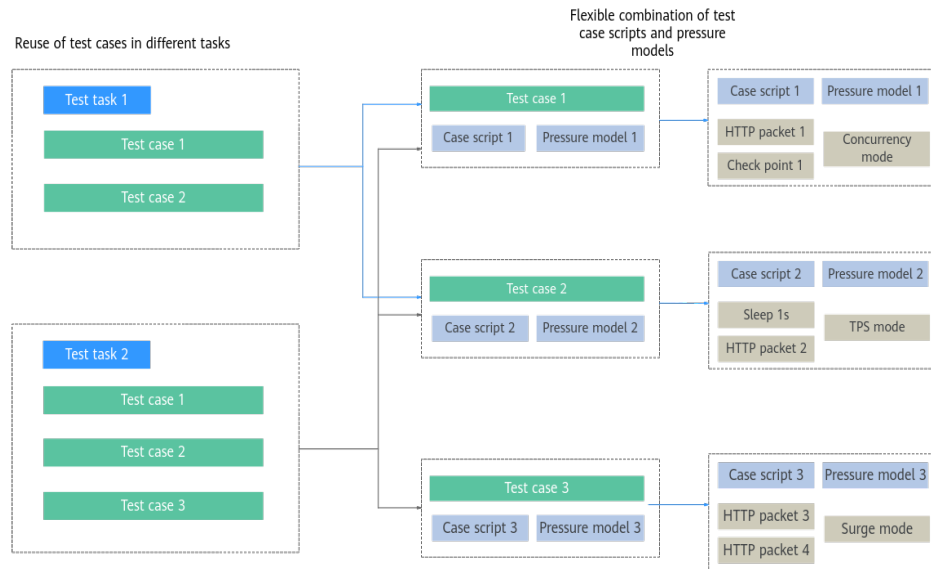
Multi-protocol and high-concurrency performance tests

- Quickly define standard HTTP/HTTPS/TCP/UDP/HLS/RTMP/WebSocket/HTTP-FLV/MQTT packet contents. You can send pressure test traffic to different tested applications through simple adjustments.
Define any fields in HTTP/HTTPS/TCP/UDP/HLS/RTMP/WebSocket/HTTP-FLV/MQTT packets based on the requirements of tested applications. For example, you can configure methods such as GET/POST/PATCH/PUT/DELETE, and URLs, headers, and bodies of HTTP requests.
- Define the behavior of virtual users for different test scenarios.
Specify the interval for sending requests of the same user by setting the think time, or define multiple request packets in a transaction to set the number of requests initiated by each user per second.
- Customize the validation for the response result to make the checkpoint for successful requests more accurate.
CodeArts PerfTest allows you to configure checkpoints based on your service requests. After obtaining response packets, CodeArts PerfTest verifies their response code, header and body fields. Only response packets that meet the specified conditions are considered normal responses.

Defining test task models for complex scenario tests

- You can test application performance by combining case scripts and pressure models for different high-concurrency scenarios.
- Test cases can be reused. You can define their pressure model and parameters such as the duration, number of concurrent users, number of flapping, and surge times, to simulate complex scenarios with traffic peaks and troughs.

Figure 1-2 Complex scenario tests



Providing professional performance test reports for easy understanding of application performance

- CodeArts PerfTest collects application performance statistics from multiple aspects, such as RPS, number of concurrent users, response latency, number of access requests, response verification failures, and response timeout.
- CodeArts PerfTest provides real-time and offline test reports for you to view and analyze test data at any time.

Managing private test clusters, isolating tenants' traffic, and providing on-demand use

- You can create test clusters as required to isolate the traffic between tenants and complete pressure tests of the internal network (Huawei Cloud VPCs) and external network. After tests are complete, you can delete clusters at any time.
- CodeArts PerfTest supports real-time scale-ins, scale-outs, and upgrades of test clusters.

2 Advantages

CodeArts PerfTest provides a one-stop performance test solution, helping you identify performance bottlenecks of applications in advance.

Cost-Efficient Simulation of Ultra-High Concurrency

- CodeArts PerfTest provides you with private test clusters. In such a test cluster, a single execution node can simulate tens of thousands of virtual users, and the entire test cluster can simulate tens of millions virtual users.
- CodeArts PerfTest is easy to use and greatly reduces test time. It simulates tens of millions instantaneous concurrent requests. In this way, you can identify application performance bottlenecks in high-concurrency scenarios and prevent system breakdown caused by a large number of access requests.
- CodeArts PerfTest supports execution of multiple concurrent tasks. It enables you to test the performance of multiple applications at the same time, greatly improving test efficiency.

Flexible and Fast Performance Testing, Achieving Quick Application Rollout

- Flexible protocol customization: HTTP/HTTPS tests are used to test the performance of various applications and microservice interfaces developed based on the HTTP/HTTPS protocol. TCP/UDP/WebSocket tests support the string and hexadecimal code stream modes, which meet the data construction requirements of various non-HTTP protocols. HLS/RTMP/HTTP-FLV/MQTT tests are supported.
- Flexible combination of multiple transaction elements and test task phases: CodeArts PerfTest provides flexible definition of data packets and transactions, as well as simulates scenarios where multiple users perform transaction operations during traffic peaks and troughs of test tasks. All of these features make CodeArts PerfTest ideal for complex scenario tests. In addition, CodeArts PerfTest allows you to specify the number of concurrent users for each transaction at each period and simulates instantaneous service traffic.

On-demand Use of Resources in Performance Tests

- Private resource group: You can create test clusters as required to isolate the traffic between tenants and complete pressure tests of the internal network (Huawei Cloud VPCs) and external network. After tests are complete, you can

delete clusters at any time. CodeArts PerfTest supports real-time scale-ins, scale-outs, and upgrades of test clusters.

- Shared resource group: You can use shared resource groups directly for debugging and pressure testing with a small number of concurrent users, without the need for creation.

Quick Location of Performance Bottlenecks

- CodeArts PerfTest provides professional performance test reports to show metrics such as transaction concurrency, RPS, throughput, and response latency, metrics that worth attention in order to provide a pleasant use experience. CodeArts PerfTest provides real-time and offline reports, allowing for analysis of test data at any time.
- CodeArts PerfTest seamlessly interconnects with Application Performance Management (APM) and Application Operations Management (AOM). Using intelligent analysis, CodeArts PerfTest associates multiple monitored objects, and displays resource usage of applications, application tracing, and full-link topology. You can view the running statuses of applications in real time, and quickly locate performance bottlenecks.

3 Application Scenarios

CodeArts PerfTest provides distributed pressure tests and is widely used in various industries, such as the Internet, digital marketing, Internet of Vehicles (IoV), and finance.

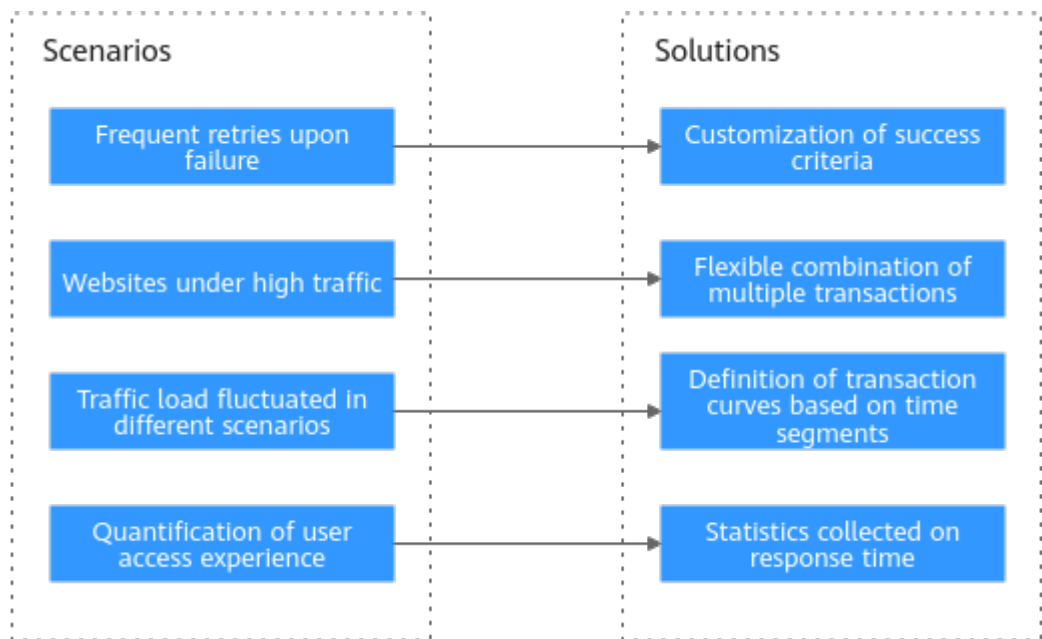
E-Commerce Flash Sale Tests

E-commerce flash sales are characterized by large-scale user concurrency, multiple burst requests, and repeated access attempts. Guaranteeing the availability of websites under heavy load is key.

Advantages

- Scenario simulation: CodeArts PerfTest simulates tens of millions instantaneous concurrent requests to a website, and can simulate a heavy-load website in just one test model.
- Professional test report: CodeArts PerfTest provides statistics on the response latency range that accurately reflect user experiences.
- Retry for failed users: User-defined comparison of results calculated using expressions allows users who failed to log in to retry.

Figure 3-1 E-commerce flash sale tests



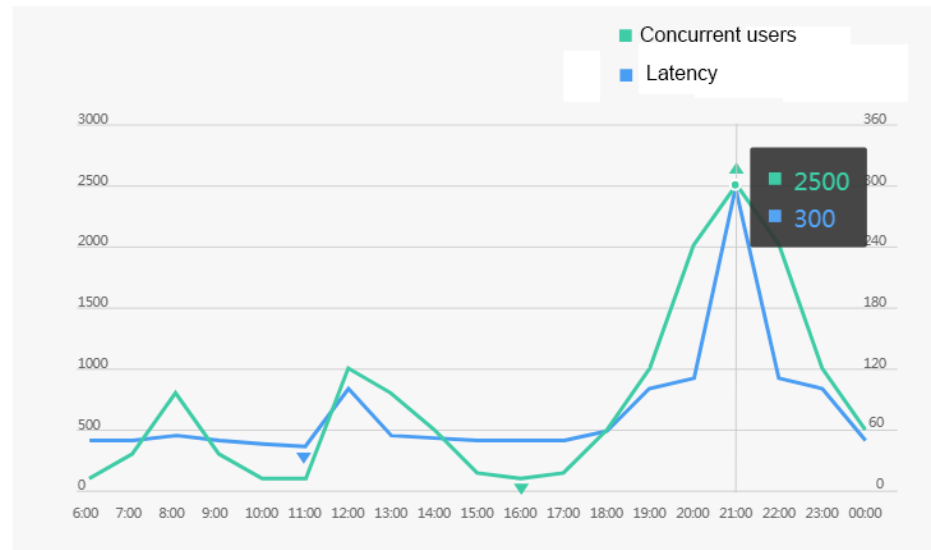
Game Peak Tests

Game services feature auto scaling in peak and off-peak scenarios. You can verify if auto scaling of games is normal, and if KPIs meet requirements in burst traffic scenarios.

Advantages

- Multi-scenario combination simulation: CodeArts PerfTest simulates real scenarios by combining multiple transactions, which include diverse elements, and customizing packets.
- Peak and off-peak scenario simulation: CodeArts PerfTest develops a pressure test curve for each transaction within a defined period to simulate peak and off-peak scenarios.
- KPI measurement: You can verify game KPIs in a peak scenario based on a customized response timeout interval.

Figure 3-2 Game peak tests



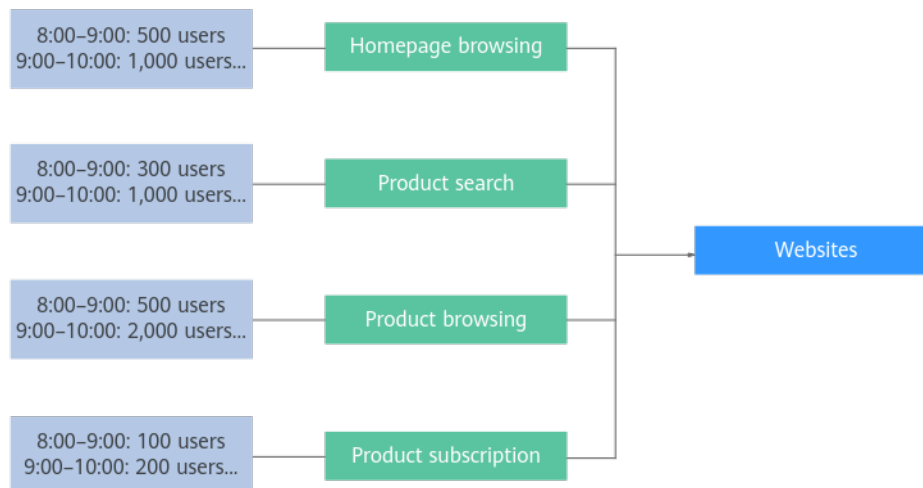
Complex Scenarios Support

CodeArts PerfTest simulates all the complexities of real traffic: A user's access may involve multiple HTTP requests; different users perform different transaction operations; user access fluctuates with transactions, displaying a sharp peak and off-peak trend; there may be instantaneous concurrent users. Performance tests must be performed on services to identify performance bottlenecks in advance.

Advantages

- Flexible model customization: CodeArts PerfTest supports multi-transaction tests in which multiple users perform multiple operations.
- Burst traffic: CodeArts PerfTest allows you to specify the number of concurrent users for each transaction at each period and simulates instantaneous service traffic.
- Result verification: Customized comparison of results calculated using expressions helps you customize transaction standards.

Figure 3-3 Complex scenarios support



Application Performance Optimization

CodeArts PerfTest allows you to build performance test models, transmit simulated traffic to applications with CodeArts PerfTest executors, view resource monitoring and tracing status of applications, and monitor concurrent transaction processing capabilities to optimize the performance.

Advantages

- Flexible expansion: CodeArts PerfTest supports multiple performance tests and on-demand expansion of executor clusters.
- One-stop solution: CodeArts PerfTest can record and report concurrency capabilities, response latency, CPU or memory usage, and microservice processing latency.
- Pay-per-use: You are charged based on the duration of your performance test and the scale of concurrency.

4 Functions

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

PerfTest Project Import and Export

To simplify operations, you do not need to write a test project from scratch. You only need to find a test project similar to the service model, export it, import it, and slightly modify it.

PerfTest Task Management

In a PerfTest project, you can create test tasks to build performance test scenarios.

PerfTest Report Management

CodeArts PerfTest provides real-time and offline test reports for you to view and analyze test data anytime.

Transaction Management

CodeArts PerfTest supports flexible combination of multiple transactions. You can add multiple transactions to a test project.

Global Variables

Global variables can be used in many scenarios. For example, you can store different usernames and passwords as global variables to simulate a scenario with multiple users.

Intelligent Analysis

CodeArts PerfTest is connected to AOM and APM to provide you with one-stop application performance analysis.

SLAs

You can gain a better understanding of the performance of your service by checking if the metrics meet the criteria specified in the SLA.

JMeter Project Management

CodeArts PerfTest supports the native JMeter engine. You can import JMeter scripts into JMeter projects to quickly initiate high-concurrency performance tests with JMeter, and view complete performance test reports.

5 Security

5.1 Authentication and Access Control

Authentication

You can access CodeArts PerfTest through the console, APIs, or SDKs. All these access modes are implemented by sending requests via the REST APIs provided by CodeArts PerfTest. Requests for calling an API can be authenticated using either of the following methods.

- Token-based authentication: Requests are authenticated using a token. During API authentication using a token, the token is added to requests to get permissions for calling the API.
- AK/SK-based authentication. Requests are encrypted using an access key ID (AK)/secret access key (SK) pair. A request must contain a signature value. This value is calculated based on the AK/SK of the requester and the specific information carried in the request body. AK/SK-based authentication is used to authenticate the identity of a request sender. For details, see [Authentication](#).

Access Control

CodeArts PerfTest controls user operations in two ways.

- Roles and permissions: Roles and permissions are required for adding, deleting, editing, and viewing CodeArts PerfTest test projects, test cases, test tasks, test reports, and custom settings.
- Fine-grained permissions: IAM fine-grained authorization is required for operations such as querying tenant projects, setting project creators, and managing tenant project members.

5.2 Data Protection

CodeArts PerfTest uses many methods to secure data.

Method	Description
Transmission encryption (HTTPS)	To ensure data transmission security, CodeArts PerfTest uses HTTPS for data transmission.
Personal data protection	CodeArts PerfTest controls access to data and records logs for operations performed on the data.
Privacy protection	CodeArts PerfTest does not use or store user data.
Data destruction	When you delete service data or deregister your account, non-key data is deleted physically in real time. Key data is marked as soft deleted and then physically deleted 10 minutes later.

5.3 Audit and Logs

- **Audit**

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, trace resource changes, audit compliance, and locate faults.

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

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

For details about CodeArts PerfTest operations that can be tracked by CTS, see [CodeArts PerfTest Operations That Can Be Recorded by CTS](#).

- **Logs**

Log Tank Service (LTS) provides one-stop log collection, log search in seconds, massive log storage, log structuring, transfer and visualization. It can be applied to application O&M, visual analysis of network logs, graded protection compliance, and operation analysis.

To analyze problems, CodeArts PerfTest records system running logs to LTS in real time and stores the logs for three days.

5.4 Resilience

CodeArts PerfTest uses multi-active stateless cross-AZ deployment and inter-AZ data disaster recovery (DR), enabling quick recovery and restart of service processes when a fault occurs.

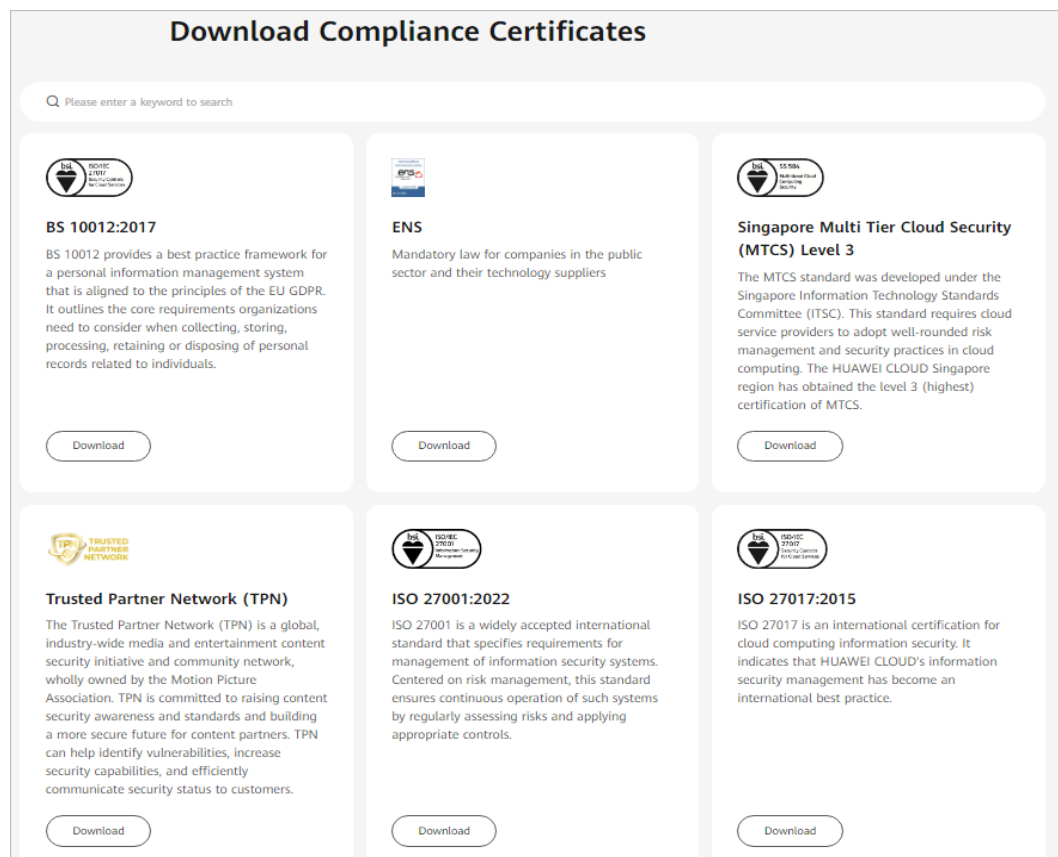
These technical solutions improve service durability and reliability.

5.5 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.

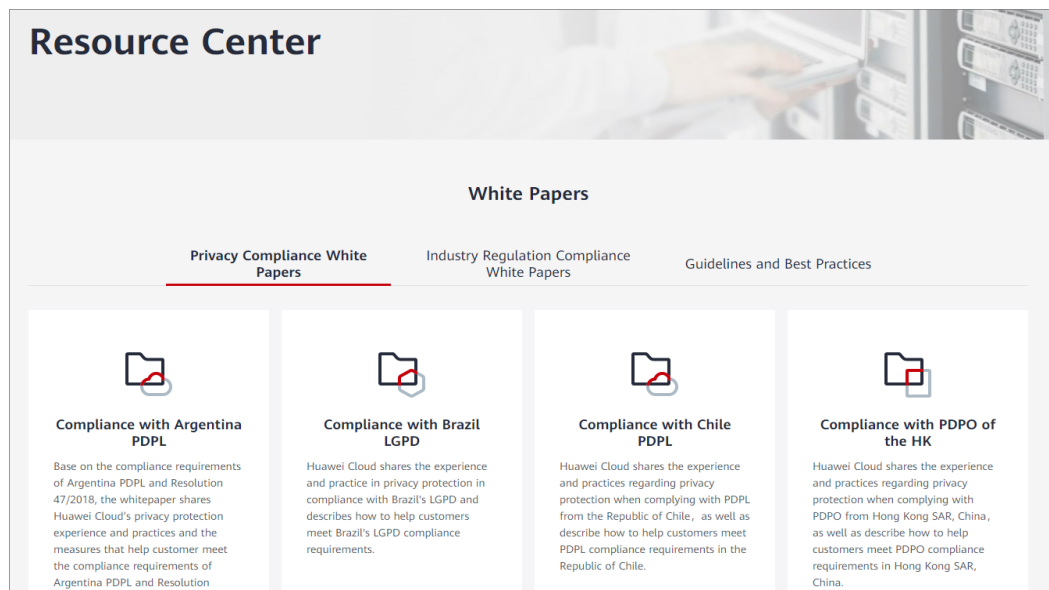
Figure 5-1 Downloading compliance certificates



Resource Center

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

Figure 5-2 Resource center



6 Permissions

If you need to grant your enterprise personnel permission to access your CodeArts PerfTest resources, use Identity and Access Management (IAM). IAM provides identity authentication, fine-grained permissions management, and access control. IAM helps you secure access to your Huawei Cloud resources. If your Huawei Cloud account does not require IAM for permissions management, you can skip this section.

IAM is a free service. You only pay for the resources in your account.

With IAM, you can control access to specific Huawei Cloud resources. For example, if you want some software developers in your enterprise to be able to use CodeArts PerfTest 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 CodeArts PerfTest resources but not permission to delete them.

IAM supports role/policy-based authorization and identity policy-based authorization.

The following table describes the differences between these two authorization models.

Table 6-1 Differences between role/policy-based and identity policy-based authorization

Authorization Model	Core Relationship	Permissions	Authorization Method	Scenario
Role/Policy	User-permission-authorization scope	<ul style="list-style-type: none"> 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.
Identity policy	User-policy	<ul style="list-style-type: none"> System-defined identity policies Custom identity policies 	<ul style="list-style-type: none"> Assigning identity policies to principals Attaching identity policies to principals 	You can authorize a user by attaching an identity policy to it. User-specific authorization and a variety of key conditions allow for more fine-grained permissions control. However, this model can be hard to set up. It requires a certain amount of expertise and is suitable for medium- and large-sized enterprises.

Assume that you want to grant IAM users permission to create ECSs in CN North-Beijing4 and OBS buckets in CN South-Guangzhou. 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, the administrator only needs to create one custom identity policy and configure 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.

Policies and actions in the two authorization models are not interoperable. You are advised to use the identity policy-based authorization model. For details about system-defined permissions, see [Role/Policy-based Authorization](#) and [Identity Policy-based Authorization](#).

For more information about IAM, see [IAM Service Overview](#).

Role/Policy-based Authorization

CodeArts PerfTest supports system role-based authorization but does not support system policy-based authorization. New IAM users do not have any permissions assigned by default. You need to first add them to one or more groups and then attach policies or roles to these groups. The users then inherit permissions from the groups and can perform specified operations on cloud services based on the permissions they have been assigned.

CodeArts PerfTest is a project-level service deployed for specific regions. When you set **Scope** to **Region-specific projects** and select projects (for example, **ap-southeast-3**) in the specified regions (for example, **AP-Singapore**), the users only have permissions for CodeArts resources in the selected projects. If you set **Scope** to **All resources**, the users have permissions for CodeArts resources in all region-specific projects. When accessing CodeArts PerfTest, the users need to switch to the authorized region.

Table 6-2 lists all the system permissions for CodeArts PerfTest.

Table 6-2 System-defined roles for CodeArts PerfTest

Role Name	Description	Dependencies
CodeArts PerfTest Administrator	Administrator permissions for CodeArts PerfTest. Users with these permissions can perform all operations on CodeArts PerfTest and test resources of the current tenant and all IAM users, such as adding, deleting, modifying, and querying resources.	This role depends on the Server Administrator , Tenant Guest , CCE Administrator , and VPC Endpoint Administrator roles.
CodeArts PerfTest Developer	Developer permissions for CodeArts PerfTest. Users with these permissions can perform all operations, such as adding, deleting, modifying, and querying resources, only on a user's own CodeArts PerfTest and test resources.	This role depends on the Tenant Guest role.
CodeArts PerfTest Operator	Operator permissions for CodeArts PerfTest. Users with these permissions can only read their own CodeArts PerfTest and test resources.	This role depends on the Tenant Guest role.
CodeArts PerfTest Resource Administrator	Resource administrator permissions for CodeArts PerfTest. Users with these permissions have all permissions related to test resources in CodeArts PerfTest.	This role depends on the Tenant Guest , CCE Administrator , and VPC Endpoint Administrator roles.

Role Name	Description	Dependencies
CodeArts PerfTest Resource Developer	Resource developer permissions for CodeArts PerfTest. Users with these permissions can only view and use CodeArts PerfTest resources, but cannot create, update, or delete infrastructure resources.	This role depends on the Tenant Guest role.

The following table lists the common operations supported by system-defined permissions for CodeArts PerfTest.

Table 6-3 Common operations supported by system-defined permissions (performance test design)

Operation	CodeArts PerfTest Administrator	CodeArts PerfTest Developer	CodeArts PerfTest Operator
Querying subscribed CodeArts PerfTest packages on the console	√	√	√
Querying all PerfTest projects of a tenant on the console	√	×	×
Querying all PerfTest projects of the current user on the console	√	√	√
Editing all PerfTest projects, test cases, and tasks of a tenant on the console	√	×	×
Editing a PerfTest project, test case, or task of the current user on the console	√	√	×
Querying all JMeter projects of a tenant on the console	√	×	×
Querying JMeter projects of the current user on the console	√	√	√

Operation	CodeArts PerfTest Administrator	CodeArts PerfTest Developer	CodeArts PerfTest Operator
Editing all JMeter projects, test plans, and thread groups of a tenant on the console	√	×	×
Editing a JMeter project, test plan, or thread group of the current user on the console	√	√	×

Table 6-4 Common operations supported by system-defined permissions (performance test execution)

Operation	CodeArts PerfTest Administrator	CodeArts PerfTest Resource Administrator	CodeArts PerfTest Resource Developer
Adding, deleting, and modifying test resources on the console	√	√	×
Querying test resources on the console	√	√	√
Querying test resources on the console when starting a PerfTest/JMeter task	√	√	√

Role/Policy Dependencies of the CodeArts PerfTest Console

Table 6-5 Role/Policy dependencies of the CodeArts PerfTest console

Console Function	Dependency	Role/Policy Required
Test resources	Cloud Container Engine (CCE)	An IAM user can access the CCE cluster information only after being granted the CCE Administrator permission.

Console Function	Dependency	Role/Policy Required
	VPC Endpoint (VPCEP)	An IAM user can enable communication between debugging and execution nodes and CodeArts PerfTest only after being granted the VPCEndpoint Administrator permission.
Intelligent analysis	Application Performance Management (APM)	IAM users can query the APM application group list only after being granted the APM ReadOnlyAccess permission.
	Application Operations Management (AOM)	IAM users can query AOM nodes only after being granted the AOM ReadOnlyAccess permission.

Identity Policy-based Authorization

CodeArts PerfTest supports identity policy-based authorization. [Table 6-6](#) lists all the system-defined identity policies for CodeArts PerfTest.

Table 6-6 System-defined identity policies for CodeArts PerfTest

Identity Policy Name	Description	Type
CodeArtsPerfTestFullAccessPolicy	Full permissions for CodeArts PerfTest	System-defined identity policy
CodeArtsPerfTestReadOnlyPolicy	Read-only permissions for CodeArts PerfTest	System-defined identity policy
CodeArtsPerfTestResourceFullAccessPolicy	Administrator permissions for CodeArts PerfTest resources	System-defined identity policy
CodeArtsPerfTestResourceDeveloperPolicy	Developer permissions for CodeArts PerfTest resources	System-defined identity policy

The following table lists the common operations supported by system-defined identity policies for CodeArts PerfTest.

Table 6-7 Common operations supported by system-defined policies (performance test design)

Operation	CodeArtsPerfTestFullAccessPolicy	CodeArtsPerfTestReadOnlyPolicy
Querying subscribed CodeArts PerfTest packages on the console	√	√
Subscribing to a CodeArts PerfTest package on the console	√	×
Querying all PerfTest projects of a tenant on the console	√	√
Editing all PerfTest projects, test cases, and tasks of a tenant on the console	√	×
Querying all JMeter projects of a tenant on the console	√	√
Editing all JMeter projects, test plans, and thread groups of a tenant on the console	√	×

Table 6-8 Common operations supported by system-defined policies (performance test execution)

Operation	CodeArtsPerfTestFullAccessPolicy	CodeArtsPerfTestResourceFullAccessPolicy	CodeArtsPerfTestResourceDeveloperPolicy
Adding, deleting, and modifying test resources on the console	√	√	×
Querying test resources on the console	√	√	√
Querying test resources on the console when starting a PerfTest/JMeter task	√	√	√

Identity Policy Dependencies of the CodeArts PerfTest Console

The **CodeArtsPerfTestFullAccessPolicy** identity policy already contains all the permissions required by the CodeArts PerfTest console. No additional identity

policies are needed. If you use an identity policy other than **CodeArtsPerfTestFullAccessPolicy**, add the identity policy of the dependent service by referring to [Table 6-9](#).

Table 6-9 Identity policy dependencies of the CodeArts PerfTest console

Console Function	Dependency	Role/Policy Required
Test resources	Cloud Container Engine (CCE)	To create and edit a private resource group, you must be granted the CCEFullPolicy permission to access the CCE cluster information.
	Identity and Access Management (IAM)	To create and edit a private resource group, you must be granted the iam:agencies:createV5 and iam:agencies:attachPolicyV5 permissions to create the perftest_admin_trust tenant agency.
	VPC Endpoint (VPCEP)	To create and edit a private resource group, you must be granted the VPCEPFullAccessPolicy permission to enable your ECSs to communicate with CodeArts PerfTest.
Intelligent analysis	Application Performance Management (APM)	To use Java probes, you must have the APMFullAccessPolicy policy assigned.

Helpful Links

- [IAM Service Overview](#)
- [Permissions Management via IAM](#)
- [Actions Supported by Identity Policy-based Authorization](#)

7 Notes and Constraints

Test Resource Groups and Their Constraints

- Test resource groups are classified into shared resource groups and private resource groups. Shared resource groups are provided by the system by default, and private resource groups need to be created. To obtain the network segment of a shared resource group, submit a service ticket to after-sales.
- Execution nodes of the shared resource group have been bound with an elastic IP address (EIP). When the tested application has network access restrictions, use a private resource group.
- A shared resource group supports a maximum of 1,000 concurrent users and 100 Mbit/s bandwidth. If higher concurrency or bandwidth is required, use a private resource group.
- JMeter test tasks can use only private resource groups.

Suggestions on Using Nodes

- If an application is deployed on a node in a cluster, the node cannot be selected to create a private resource group. Do not run any applications or perform other functions on nodes used for test resource groups. Otherwise, applications may run abnormally.
- If you want to perform pressure tests on external services, bind an EIP to each execution node. If you want to debug external services, bind EIPs to both the debugging node and execution node. The test bandwidth is limited by the EIPs' bandwidth.
- Create at least two empty nodes. One is the execution node/executor (a target machine that a pressure test will be performed on and provides performance data during testing). The other serves as a debugging node to debug the execution node. Create nodes of the required specifications based on the number of concurrent users for a pressure test. For details about the recommended node specifications, see [Table 7-1](#) and [Table 7-2](#). These specifications are for reference only. Resource specification requirements for a pressure test are influenced by think time, protocol type, the size and number of requests and responses, response time, and result verification. Adjust the specifications as needed.
- In a PerfTest test project, one execution node with 8 vCPUs and 16 GB memory supports 10,000 concurrent users. In a JMeter test project, one

execution node with 8 vCPUs and 16 GB memory supports 2,000 concurrent users.

Table 7-1 Recommended node specifications for PerfTest projects

Concurrent Users	Specifications	Quantity
0–5,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 4 vCPUs 8 GB	1
5,001–10,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	1
10,001–20,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	2
20,001–30,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	3
30,001–40,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	4
40,001–50,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	5
More than 50,001	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	<i>n</i>

Table 7-2 Recommended node specifications for JMeter projects

Concurrent Users	Specifications	Quantity
0–1,000	Debugging node: 4 vCPUs 8 GB	1

Concurrent Users	Specifications	Quantity
	Execution node: 4 vCPUs 8 GB	1
1,001–2,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	1
2,001–4,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	2
4,001–6,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	3
6,001–8,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	4
8,001–10,000	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	5
More than 10,001	Debugging node: 4 vCPUs 8 GB	1
	Execution node: 8 vCPUs 16 GB	<i>n</i>

Quota Limits

Pay attention to the quota limits listed in [Table 7-3](#).

Table 7-3 Quota limits

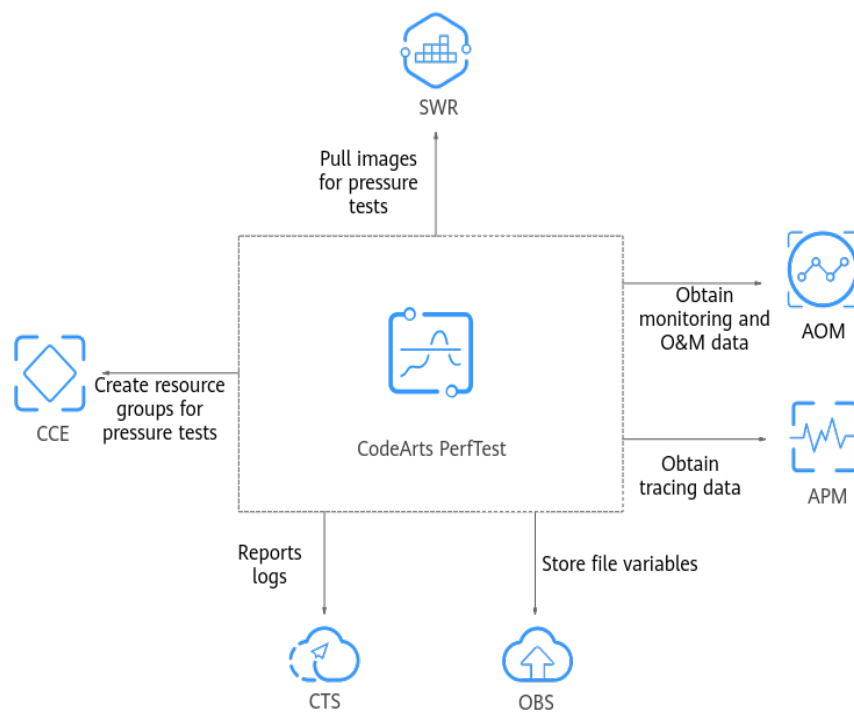
Parameter	Description	Default Value
Maximum number of concurrent users in a task	Restriction on the number of concurrent users in a task	1,000,000

Parameter	Description	Default Value
Quota for the number of instance resource groups	Restriction on the number of instance resource groups	5
Quota for the number of project transactions	Restriction on the number of transactions in a project	100
Quota for the number of transaction elements	Restriction on the number of elements in a transaction	40
Quota for the number of projects	Restriction on the number of tenant projects	100
Quota for the number of tasks	Restriction on the number of tasks in a project	200
Quota for the number of concurrent users in a shared resource group	Restriction on the number of concurrent users in a shared resource group	1,000
Quota for the number of running tasks in a shared resource group	Restriction on the number of running tasks in a shared resource group	2
Quota for the running duration of a task in a shared resource group	Restriction on the running duration of a task in a shared resource group	3,600
Quota for the number of file variables	Restriction on the number of file variables	100

8 Related Services

CodeArts PerfTest works with other cloud services to provide you with performance tests. **Figure 8-1** shows the relationship between them.

Figure 8-1 Related services



CCE

CCE provides highly scalable, high-performance, enterprise-class Kubernetes clusters and supports Docker containers. With CCE, you can easily deploy, manage, and scale containerized applications on Huawei Cloud.

CodeArts PerfTest uses CCE to create nodes that serve as the debugging and execution nodes in CodeArts PerfTest pressure test resource groups.

AOM

AOM is a one-stop, multi-dimensional O&M management platform for cloud applications. It monitors applications and related cloud resources in real time. AOM collects and associates resource metrics, logs, and events to analyze application health statuses, as well as provides flexible alarm reporting and data visualization. This helps you detect faults in a timely manner and monitor running statuses of applications, resources, and services in real time.

AOM monitors and centrally manages servers, storage devices, networks, web containers, and applications hosted in Docker and Kubernetes, effectively preventing problems, facilitating fault locating, and reducing O&M costs. In addition, AOM provides unified APIs for interconnecting self-developed monitoring or reporting systems. Unlike traditional monitoring systems, AOM monitors services by applications, meeting your requirements for high efficiency and fast iteration. It provides effective IT support for your services, and protects and optimizes your IT assets, enabling you to achieve strategic goals.

CodeArts PerfTest uses the resource monitoring provided by AOM, which is useful for intelligent analysis.

CTS

Cloud Trace Service (CTS) is a log audit service for cloud security. It allows you to collect, store, and query operation records of cloud resources for security analysis, compliance auditing, resource tracking, and fault locating.

CTS can record operations associated with CodeArts PerfTest for future query, audit, and backtracking.

APM

APM monitors and manages the performance of cloud applications in real time. APM provides performance analysis of distributed applications, helping O&M personnel quickly locate and resolve faults and performance bottlenecks.

APM is a cloud application diagnosis service with powerful analytic tools, supporting applications based on multiple Java frameworks. It displays the application statuses, call processes, and user operations through topologies, tracing, and transactions, allowing you to quickly locate performance bottlenecks.

APM and CodeArts PerfTest work together. APM provides call chain tracing and resource monitoring for pressure test applications deployed on APM. On the **Call Chain** page of APM, you can view the call chain information of applications. Red indicates abnormal, yellow indicates alarm, and green indicates normal.

OBS

Object Storage Service (OBS) is a secure, reliable, and cost-effective cloud storage service. With OBS, you can easily create, modify, and delete buckets, as well as upload, download, and delete objects.

The file variables of CodeArts PerfTest can be stored in OBS.

SWR

SoftWare Repository for Container (SWR) provides easy, secure, and reliable management over container images throughout their lifecycles, facilitating the deployment of containerized services.

For CodeArts PerfTest, SWR is used to obtain the images used for pressure tests in clusters.

9 Basic Concepts

Execution Node

An execution node is the target machine that a pressure test will be performed on and can provide performance data during testing.

Debugging Node

A debugging node is used to debug execution nodes.

Test Resources

Test resources refer to private resource groups.

CodeArts PerfTest Resources

CodeArts PerfTest resources refer to items such as test projects, directories, cases, and tasks.

Test Project

Test projects are classified into PerfTest projects and JMeter projects. PerfTest projects provide project management capabilities, allowing you to share and reuse the contents of transactions, pressure test tasks, and test reports within a test project, and create different test projects for different test programs. JMeter projects are used to import JMeter scripts to CodeArts PerfTest.

Transaction

A transaction is a user-defined operation model that consists of HTTP/HTTPS/TCP/UDP/WebSocket packet, think time, response extraction, checkpoint, and HLS/RTMP/HTTP-FLV/MQTT packet.

Packet

Packets are data blocks transmitted between applications such as HTTP. These data blocks start with text metadata that describes the packet content and meaning. The text metadata is followed by optional data. Packets are transmitted among clients, servers, and agents.

Think Time

To better simulate user behavior, insert a waiting time between different operations. For example, when a user receives data from the server, the user may wait several seconds before viewing data and providing responses. This period of time is called think time.

Response Extraction

If a transaction contains multiple packets, the output of the previous packet, which is extracted by a regular expression or JSON, is used as the input of the next packet.

Checkpoint

Checkpoints are where you define the verification information to determine whether the contents returned by the server are correct.

Test Task

A test task initiates a performance test based on a defined test model.

Test Report

When a test task is complete, a test report will be generated to present the test results.

Number of Concurrent Users

It refers to the number of users performing operations on a system at the same time. In CodeArts PerfTest, it is the number of virtual users you define when you configure a test phase.

RPS

Requests per second (RPS) indicates the number of requests per second. Average RPS = Total number of requests in a statistical period/Statistical period.

VUM

Virtual user minute (VUM) indicates the number of resources consumed by a task. The calculation formula is $VUM = VU \times M$, in which VU indicates the number of concurrent virtual users and M indicates the pressure test duration, in minutes.

Bandwidth

Bandwidth records the real-time bandwidth usage during the running of the pressure test task. Uplink bandwidth refers to the speed at which the CodeArts PerfTest execution node sends out data. Downlink bandwidth refers to the speed at which the CodeArts PerfTest execution node receives data.

Response Time

Response time indicates the duration from the time when a client sends a request to the time when the client receives a response from the server.

Response Timeout

If the corresponding TCP connection does not return the response data within the set response timeout (5 seconds by default), the transaction request is considered a response timeout. Possible causes are: the tested server is busy, in crashes, or the network bandwidth is fully occupied.

Verification Failure

The response packet content and response code returned from the server do not meet the expectation (the default expected response code of HTTP/HTTPS is 200), such as code 404 or 502. A possible cause is that the tested service cannot be processed normally in scenarios with a large number of concurrent users. For example, a database bottleneck occurs in the distributed system or the backend application returns an error.

Resolution Failure

All response packets are received, but some packets are lost. As a result, the entire transaction response is incomplete. In this case, network packet loss may be the cause.