ServiceStage

User Guide

Issue	01
Date	2025-08-26

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2025. 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

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 Instruction 1
1.1 Before You Start 1
1.2 ServiceStage Product Editions 1
1.3 Logging In to ServiceStage 2
1.4 Changing the ServiceStage Product Edition 7
1.5 Buying a ServiceStage Resource Package 7
1.6 ServiceStage Usage Process 9
2 Using IAM to Grant Access to ServiceStage 12
2.1 Creating a User and Granting ServiceStage Permissions 12
2.2 Creating a Custom Policy 13
2.3 Granting Permissions to ServiceStage-Dependent Services 14
3 Environment Management 16
3.1 Environment Overview 16
3.2 Creating an Environment 17
3.2.1 Creating a VM Environment in Resource Management Mode 18
3.2.2 Creating a Kubernetes Environment in Resource Management Mode 20
3.2.3 Creating a VM + Kubernetes Environment in Resource Management Mode 23
3.3 Managing Basic Resources in the Environment 25
3.3.1 Managing Basic Resources 25
3.3.2 Managing Cluster Resources 46
3.3.2.1 Binding a Cluster 46
3.3.2.2 Managing Namespaces 55
3.3.2.3 Managing Configuration Items 60
3.3.2.4 Managing Secrets 65
3.3.2.5 Managing Plug-ins 69
3.3.2.5.1 Managing Sermant Injector 69
3.3.2.5.2 Managing Metrics Server 73
3.3.2.6 Unbinding a Cluster 76
3.3.3 Managing VM Agents 76
3.3.4 Removing a Basic Resource 82
3.4 Managing Environment Variables of an Environment 82
3.5 Editing an Environment 90

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

3.6 View Environment Details....92
3.7 Deleting an Environment....93
4 Application Management....95
4.1 Application Overview....95
4.2 Creating an Application....96
4.3 Managing Environment Variables of an Application....97
4.4 Editing an Application....102
4.5 Viewing Application Overview....103
4.6 Deleting an Application....104
5 Component Management....106
5.1 Component Overview....106
5.2 Creating and Deploying a Component....114
5.2.1 Creating a Component Based on a Container Using Manual Configurations....114
5.2.2 Creating a Component Based on a Container Using YAML Configurations....135
5.2.3 Creating a Component Based on a VM....152
5.2.4 Creating a Component by Importing a CCE Workload....160
5.2.5 Creating a Component Using a Component Template....167
5.3 Upgrading a Single Component....172
5.3.1 Upgrading a Component in Single-batch Release Mode....172
5.3.2 Upgrading a Component in Rolling Release Mode....179
5.3.3 Upgrading a Component in Dark Launch Mode....187
5.3.4 Upgrading a Component Using a Component Template....199
5.4 Upgrading Components in Batches....201
5.5 Redeploying a Component....205
5.5.1 Redeploying a Component in Single-batch Release Mode....205
5.5.2 Redeploying a Component in Rolling Release Mode....211
5.5.3 Redeploying a Component in Dark Launch Mode....217
5.5.4 Redeploying a Component Using a Component Template....225
5.5.5 Redeploying a Job Component....227
5.6 Rolling Back a Component....234
5.7 Cloning Components in Batches....235
5.8 Synchronizing Component Configurations in Batches....247
5.9 Managing Component Details....247
5.9.1 Managing Component Labels....247
5.9.2 Changing the Component Description....249
5.9.3 Managing Component Instances....250
5.9.4 Configuring the Component Access Mode....253
5.9.5 Changing the Component Access Domain Name....254
5.9.6 Starting or Stopping a Component....255
5.9.7 Manually Synchronizing the Component Status....257
5.10 Configuring a Scaling Policy of a Component Instance....257
5.10.1 Component Instance Scaling Policy Overview....257

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.10.2 Configuring a Manual Scaling Policy....258
5.10.3 Managing HPA Policies....259
5.11 Component O&M....265
5.11.1 Viewing Component Running Metrics....266
5.11.2 Customizing Component Running Metrics....267
5.11.3 Managing Component Logs....268
5.11.3.1 Managing Component AOM Logs....268
5.11.3.2 Managing Component LTS Logs....269
5.11.3.2.1 LTS Log Overview....269
5.11.3.2.2 Configuring an LTS Log Group....269
5.11.3.2.3 Searching for Running Logs....271
5.11.3.2.4 Quickly Querying Logs....272
5.11.3.2.5 Using Visualization to Analyze Logs....273
5.11.3.2.6 Viewing Real-Time Logs....274
5.11.3.2.7 Unbinding an LTS Log Group....274
5.11.4 Viewing Container Logs....275
5.11.5 Configuring Alarm Thresholds for Resource Monitoring....276
5.11.6 Viewing Component Running Events....279
5.12 Managing Cloud Service Settings of a Component....280
5.12.1 Binding a Microservice Engine....280
5.12.2 Binding a Distributed Cache....285
5.12.3 Binding a Cloud Database....286
5.12.4 Binding a GaussDB Instance....287
5.12.5 Binding a DMS for RocketMQ Instance....288
5.12.6 Binding a CSS Instance....289
5.13 Managing Container Settings of a Container-Deployed Component....290
5.13.1 Adding an Environment Variable of a Component....290
5.13.2 Configuring the Startup Command and Lifecycle of a Component....292
5.13.3 Configuring Data Storage....295
5.13.4 Configuring a Log Policy of an Application....308
5.13.5 Configuring Health Check....313
5.14 Managing Application Settings of a Container-Deployed Component....315
5.15 Managing Advanced Settings of a Container-Deployed Component....318
5.15.1 Configuring a Component Upgrade Policy....318
5.15.2 Configuring a Scheduling Policy of a Component Instance....320
5.15.3 Configuring a Tolerance Policy of a Component Instance....326
5.15.4 Configuring Performance Management of a Container-Deployed Component....328
5.15.5 Configuring Custom Monitoring of a Component....329
5.15.6 Configuring Component Job Settings....330
5.16 Managing VM-Deployed Component Configurations....331
5.16.1 Configuring VM Settings....331
5.16.2 Managing Application Settings of a VM-Deployed Component....335

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.16.3 Configuring Application Performance Management of a VM-Deployed Component....338
5.16.4 Configuring Log Settings of a VM-Deployed Component....338
5.17 Viewing Component Details....340
5.18 Viewing the Component Running Environment....342
5.19 Deleting a Component....342
6 Configuration Management....344
6.1 Configuration Overview....344
6.2 Managing Groups....346
6.3 Managing Configuration Files....347
6.3.1 Creating a Configuration File....347
6.3.2 Importing a Configuration File....350
6.3.3 Exporting a Configuration File....351
6.3.4 Editing a Configuration File....352
6.3.5 Viewing Version Details of a Configuration File....355
6.3.6 Comparing Configuration Files....355
6.3.7 Rolling Back a Configuration File....356
6.3.8 Pushing a Configuration File Version to an Associated Component....357
6.3.9 Deleting a Configuration File....358
7 Release Management....359
7.1 New Release Management....359
7.1.1 Overview....359
7.1.2 Creating a Region Release Task....360
7.1.2.1 Creating a Region Release Task Using Manual Configurations....360
7.1.2.2 Creating a Region Release Task Using a Release Task Template....370
7.1.3 Managing Region Release Tasks....375
7.1.3.1 Releasing a Region Release Task....375
7.1.3.2 Pausing the Release of a Region Release Task....376
7.1.3.3 Continuing to Release a Region Release Task....377
7.1.3.4 Retrying a Region Release Task....379
7.1.3.5 Rolling Back a Region Release Task....380
7.1.3.6 Editing a Region Release Task Created Using Manual Configurations....381
7.1.3.7 Editing a Region Release Task Created Using a Release Task Template....394
7.1.3.8 Cloning a Region Release Task Created Using Manual Configurations....397
7.1.3.9 Cloning a Region Release Task Created Using a Release Task Template....399
7.1.3.10 Configuring a Triggering Policy of a Region Release Task....402
7.1.3.11 Viewing the Running Records of a Region Release Task....403
7.1.3.12 Viewing Region Release Task Details....404
7.1.3.13 Deleting a Region Release Task....405
7.2 Old Release Management....406
7.2.1 Overview....406
7.2.2 Creating a Release Task....407
7.2.2.1 Creating a Batch Deployment Release Task....407

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

7.2.2.2 Creating a Batch Upgrade Release Task....419
7.2.2.3 Creating a Batch Clone Release Task....423
7.2.3 Releasing a Release Task....436
7.2.4 Suspending the Release of a Release Task....437
7.2.5 Editing a Release Task....438
7.2.5.1 Editing a Batch Deployment Release Task....438
7.2.5.2 Editing a Batch Upgrade Release Task....450
7.2.5.3 Editing a Batch Clone Release Task....455
7.2.6 Cloning a Release Task....468
7.2.6.1 Cloning a Batch Deployment Release Task....468
7.2.6.2 Cloning a Batch Upgrade Release Task....480
7.2.6.3 Cloning a Batch Clone Release Task....485
7.2.7 Rolling Back a Released Batch Upgrade Release Task....498
7.2.8 Viewing a Release Task....499
7.2.9 Deleting a Release Task....501

8 System Settings....503
8.1 Tech Stack Management....503
8.1.1 Adding a Custom Technology Stack Version....503
8.1.2 Enabling and Disabling a Technology Stack Version....507
8.1.3 Enabling and Disabling a Technology Stack....508
8.1.4 Editing a Custom Technology Stack Version....509
8.1.5 Deleting a Custom Technology Stack Version....512
8.2 Directory Settings....512

9 Deployment Source Management....517
9.1 Managing the Software Center....517
9.1.1 Managing Software Packages....517
9.1.2 Packaging Specifications of Software Packages....522
9.2 Managing Image Repositories....524
9.2.1 Uploading an Image....524
9.2.2 Managing Images....525
9.3 Managing Organizations....528

10 Continuous Delivery....531
10.1 Continuous Delivery Overview....531
10.2 Viewing Build Jobs....532
10.3 Creating a Source Code Job....533
10.4 Creating a Package Job....536
10.5 Maintaining Build Jobs....538
10.5.1 Editing a Source Code Job....538
10.5.2 Editing a Package Job....539
10.5.3 Starting a Build Job....541
10.5.4 Configuring Branch or Tag Build....541

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

10.5.5 Deleting a Build Job....542
10.6 Managing Pipelines....543
10.6.1 Pipeline Overview....543
10.6.2 Creating a Pipeline....543
10.6.3 Configuring a Pipeline Triggering Policy....548
10.6.4 Editing a Pipeline....549
10.6.5 Cloning a Pipeline....552
10.6.6 Starting and Stopping a Pipeline....556
10.6.7 Approving a Pipeline....557
10.6.8 Viewing Pipeline Details....558
10.6.9 Viewing Pipeline Running Records....559
10.6.10 Deleting a Pipeline....559
10.7 Managing Repository Authorization....560
11 Microservice Engine....563
11.1 Cloud Service Engine Overview....563
11.2 Creating a Microservice Engine....563
11.3 Managing Microservice Engines....566
11.3.1 Viewing Microservice Engine Information....566
11.3.2 Obtaining the Service Center Address of a Microservice Engine....567
11.3.3 Obtaining the Configuration Center Address of a Microservice Engine....568
11.3.4 Viewing the Instance Quota of a Microservice Engine....569
11.3.5 Viewing the Configuration Item Quota of a Microservice Engine....569
11.3.6 Configuring Backup and Restoration of a Microservice Engine....570
11.3.7 Managing Public Network Access for a Microservice Engine....572
11.3.7.1 Binding an EIP....572
11.3.7.2 Unbinding an EIP....572
11.3.8 Viewing Microservice Engine Operation Logs....573
11.3.9 Upgrading a Microservice Engine Version....573
11.3.10 Deleting a Microservice Engine....574
11.3.11 Managing Security Authentication for a Microservice Engine....575
11.4 Using Microservice Engines....577
11.4.1 Using the Microservice Dashboard....577
11.4.2 Managing Microservices....578
11.4.3 Microservice Governance....590
11.4.4 Configuration Management (Applicable to Engine 2.x)....609
11.4.5 Configuration Management (Applicable to Engine 1.x)....618
11.4.6 System Management....620
11.4.6.1 Overview....621
11.4.6.2 Accounts....622
11.4.6.3 Roles....627
12 Key Operations Recorded by CTS....634
12.1 ServiceStage Operations That Can Be Recorded by CTS....634

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

12.2 Viewing CTS Traces in the Trace List 635

13 Viewing Monitoring Metrics and Alarms 642

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

1 Instruction

1.1 Before You Start

ServiceStage is an application management and O&M platform that lets you deploy, roll out, monitor, and maintain applications all in one place. It supports technology stacks such as Java, PHP, Python, Node.js, Docker, and Tomcat, and supports microservice applications such as Apache ServiceComb Java Chassis (Java chassis) and Spring Cloud, making it easier to migrate enterprise applications to the cloud.

Before using ServiceStage, you need to learn the following basic knowledge and training to understand ServiceStage functions and how to use ServiceStage.

1. Learn about ServiceStage

   Service Overview describes the capabilities, advantages, and application scenarios of ServiceStage.

2. Get started with ServiceStage

   Getting Startedhelps you experience ServiceStage functions.

3. Learn about ServiceStage functions and how to use ServiceStage

   See **Environment Management** to **Microservice Engine**.

4. Use ServiceStage

   Best Practices describes how to use ServiceStage in specific application scenarios.

5. Buy ServiceStage

   Billing describes ServiceStage billing modes.

1.2 ServiceStage Product Editions

Table 1-1 describes the basic and professional ServiceStage editions. For product pricing of each edition, see **Product Pricing Details**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 1-1 ServiceStage editions

Edition	Description
Basic	Provides basic environment and application running management. • Unified management of Huawei Cloud VMs and containers • Environment management • Build • Pipeline release
Professional	Provides more capabilities on top of the basic edition, including HA deployment, configuration management, microservice governance, full-link dark launch, and full-stack observability. • Unified management of Huawei Cloud VMs, containers, and heterogeneous resources • Template-based creation of environments and components • Application and component management • Fine-grained permission isolation of resource and application deployment between tenants • Continuous integration/deployment • Microservice governance (exclusive ServiceComb instances rewarded) • HA release (single-batch release, rolling release, and dark launch, as well as HA features such as graceful startup and shutdown of microservices and intra-city dual-AZ deployment) • Full-link dark launch

1.3 Logging In to ServiceStage

Before using ServiceStage, log in to the ServiceStage console.

Prerequisites

- You have **registered a Huawei account and enabled Huawei Cloud services**.
- The login account has the permission to use ServiceStage. For details, see **Using IAM to Grant Access to ServiceStage**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Logging In to ServiceStage

Step 1 Log in to the **management console**.

Step 2 Click and select a region.

Step 3 Click in the upper left corner, and click **ServiceStage**.

- If you log in for the first time, click **Authorize** on the displayed service authorization page to authorize ServiceStage to use the services on which it depends. Then, the **ServiceStage** console is displayed.
- If this is not your first login, the **ServiceStage** console is displayed directly.

For details, see **Table 1-2**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 1-2 ServiceStage console

Module	Description
Overview	The **Overview** page provides a dashboard, including ServiceStage edition selection, total number of environments, applications, and components, monitoring information, alarms, and documentation. • Edition selection: ServiceStage provides the basic and professional editions in pay-per-use billing mode. You can select an edition as required. For details, see **Changing the ServiceStage Product Edition**. • Environments: displays the number of created environments. You can click **Environments** to go to the **Environment Management** page and view environment details. • Applications: displays the number of created applications. You can click **Applications** to go to the **Application Management** page and view application details. • Components: displays the number of deployed components. You can click **Components** to go to the **Component Management** page and view component details. • Customize Monitoring: Move the cursor to **Customize Monitoring** in the upper right corner and select the applications and environment to be displayed on the **Overview** page. A maximum of four monitoring information records can be displayed. Where, – Applications: displays the name of each application, number of deployed components (including available and unavailable components), and CPU and memory usage of the application. – Environments: displays the name of each environment, CPU and memory usage in the environment, number of components deployed in the environment, resource health, and instance health of deployed components. Click **CPU usage** or **Memory usage** on an environment card to enable or disable the information display. • Remove the monitoring information that does not need to be displayed: – Click in the upper left corner of a card to be removed and click **Remove**. – Move the cursor to **Customize Monitoring** in the upper right corner and deselect the monitoring information that does not need to be displayed. • Alarms: Click **Learn More** in the **Alarms** area to go to the AOM console and view ServiceStage alarm details. • Documentation: Click **Learn More** in the **Documentation** area to view ServiceStage documents.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Module	Description
Environment Management	An environment is a collection of compute, network, and middleware resources used for deploying and running a component. The **Environment Management** page allows you to create, edit, and delete environments, and configure resources (manage and remove resources). Created environments are displayed in a list.
Application Management	An application is a service system with functions and consists of one or more components. The **Application Management** page allows you to create, edit, and delete applications. Created applications and the number of components created under them are displayed in a list, and entries for creating components under applications are available.
Component Management	A component is a service feature implementation of an application. It is carried by code or software packages and can be independently deployed and run. The **Component Management** page displays components of all applications in a list, and provides the component details page as well as the entries for component creation and O&M.
Configuration Management	In ServiceStage, a configuration is a file. You can fill the environment and application system variables (such as the IP address, port number, database address, and application name associated with the environment) in a configuration file. When the component is associated with the configuration for deployment, the system variables are automatically replaced with the actual values. This implements multi-environment use with one-time configuration through file mounting. The **Configuration Management** page allows you to create, edit, and delete configurations. Created configurations are displayed in a list.
Release Management	Release tasks of different types provide functions for releasing applications: single-component release, batch operations (release, upgrade, and clone), and dependency-based orchestration. This feature enables you to flexibly deploy components in batches, improving service efficiency and user experience. This page displays created release tasks and provides entries for creating and managing release tasks.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Module	Description
System Settings	• Provides an entry for customizing technology stack versions to provide multiple runtime systems. It also provides entries for starting, stopping, editing, and deleting technology stack versions. • Provides an entry for configuring and managing directories to manage the startup users, running directories, and log directories of VM-deployed components. You can view the default directory rules of the system or customize directory rules.
Deployment Source Management	Provides functions such as organization management, software repository, and image repository. • Organization management is used to isolate images and assign access permissions (read, write, and manage) to different users. • Image repositories are used to store and manage Docker images. • Software repositories are used to store, manage, and deploy software packages.
Continuous Delivery	Provides functions such as viewing build projects, releasing build projects, and authorizing repositories. • Build The software package or image package can be generated with a few clicks in a build job. In this way, the entire process of source code pull, compilation, packaging, and archiving is automatically implemented. • Pipeline One-click deployment can be achieved through pipeline. In this way, the entire process of source code pull, compilation, packaging, archiving, and deployment is automatically implemented. This unifies the integration environment and standardizes the delivery process. • Repository Authorization You can create repository authorization so that build projects and application components can use the authorization information to access the software repository.
Cloud Service Engine	Provides operation entries for instance management, dashboard usage, microservice catalog management, microservice governance, configuration management, and system management of engine instances.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

1.4 Changing the ServiceStage Product Edition

You can change the ServiceStage edition based on service requirements. For details about ServiceStage editions, see **ServiceStage Product Editions**.

After the edition is changed, the new edition's capabilities will be provided and you will be billed based on the usage of the new edition. For product pricing of each edition, see **Product Pricing Details**.

If you change from profession edition to basic edition, some functions provided by the professional edition will be lost.

Changing the ServiceStage Product Edition

Step 1 **Log in to ServiceStage**.

Step 2 On the right of the **Overview** page, click **Change** under **Current**

Step 3 Specify **Target**.

Step 4 Click **Next** and view and confirm the **Product Info** of the target version.

Step 5 Select **I have read and agreed to Legal Statement and Privacy Statement**.

Step 6 Click **Confirm**.

----End

1.5 Buying a ServiceStage Resource Package

Resource packages are suitable for predictable resource usage periods and are more cost-effective than pay-per-use billing mode. You can select a proper resource package and required duration based on the number of required instances.

A purchased package cannot be unsubscribed. After it expires, you can continue using ServiceStage and your data is still secure.

Only the account administrator can perform this operation. For details about the account and IAM user, see **Basic Concepts**.

Buying a ServiceStage Resource Package

Step 1 **Log in to ServiceStage**.

Step 2 In the **Current** area, check whether the desired resource package is of the current product edition.

- If yes, go to **Step 4**.
- If no, perform **Changing the ServiceStage Product Edition** and then go to **Step 4**.

Step 3 In the **Current** area, use either of the following methods to go to the **Buy Resource Package** page:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Click **Buy Resource Package**.
- Click **View** next to **Purchased Resource Packages**. On the **Purchased Resource Packages** page that is displayed, click **Buy More Resource Packages**.

Step 4 Select the type of the desired resource package based on the current product edition and service requirements.

Current Product Edition	Package Types
Basic	**Entry-level Resource Package**: supports up to 20 application instances, and provides the AOM professional edition for free.
Professional	• **Basic Package**: supports up to 100 application instances, and provides one exclusive microservice engine (HA; 100 instances) and AOM professional edition for free. • **Medium Package**: supports up to 200 application instances, and provides one exclusive microservice engine (HA; 200 instances) and AOM enterprise edition for free. • **Advanced Package**: supports up to 500 application instances, and provides two exclusive microservice engines (HA; total instances of the two engines <= 500) and AOM enterprise edition for free. • **Golden Package**: supports up to 1000 application instances, and provides three exclusive microservice engines (HA; total instances of the three engines <= 1000) and AOM enterprise edition for free. • **Platinum Package**: supports up to 2000 application instances, and provides four exclusive microservice engines (HA; total instances of the four engines <= 2000) and AOM enterprise edition for free. • **Top Package**: supports up to 5000 application instances, and provides five exclusive microservice engines (HA; total instances of the five engines <= 5000) and AOM enterprise edition for free.

Step 5 Specify **Required Duration**.

You can select a subscription duration of 1 to 9 months.

Step 6 Click **Next** and confirm the **Product Info** of the desired resource package.

Step 7 Select **I have read and agreed to Legal Statement and Privacy Statement**.

Step 8 Click **Pay Now**, confirm the order, and select a payment method.

Step 9 Click **Pay**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

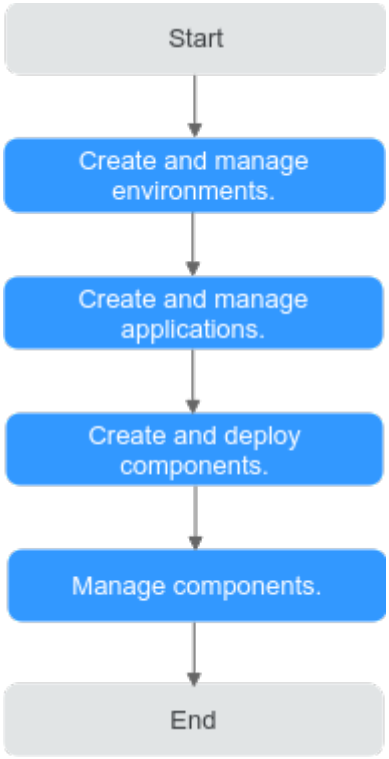
Return to the ServiceStage **Overview** page. In the **Current** area, click **View** next to **Purchased Resource Packages** to view details about the purchased resource package and renew the resource package that is about to expire.

----End

1.6 ServiceStage Usage Process

As shown in **Figure 1-1**, you can use ServiceStage to create running environments, create applications, create and deploy components, and manage components.

Figure 1-1 ServiceStage usage process



1. Create and manage environments.

   An environment is a collection of basic compute (such as cluster and ECS), network (such as ELB and EIP), and middleware (such as DCS and RDS) resources, used for component deployment and running. ServiceStage combines multiple basic resources into an environment, including development, test, pre-production, and production environments. Managing resources, and creating and deploying components by environment simplify O&M management.

   Before using ServiceStage to create and deploy a component, create an environment and manage resources for the environment. For details, see **Environment Management**.

2. Create and manage applications.

   An application is a service system with functions and consists of one or more components.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

A component must belong to an application. Before using ServiceStage to create and deploy a component, create an application. For details, see **Application Management**.

3. Create and deploy components.

   A component is a service feature implementation of an application. It is carried by code or software packages and can be independently deployed and run in an environment.

   Use ServiceStage to create and deploy a component. For details, see **Creating and Deploying a Component**.

4. Manage components.

   Manage components after the components are created and deployed.

   - To upgrade a single component, see **Upgrading a Single Component**.
   - To upgrade components in batches, see **Upgrading Components in Batches**.
   - To redeploy a component using the historical version configuration as the template, see **Redeploying a Component**.
   - To roll back a component, see **Rolling Back a Component**.
   - To manage component running metrics and logs, see **Component O&M**.
   - To manage component instance scaling, see **Configuring a Scaling Policy of a Component Instance**.

You can also use other capabilities provided by ServiceStage based on service requirements.

- **Configuration Management**: By creating a unified configuration, you can fill the system variables (such as the IP address, port number, database address, and application name associated with the environment) of the component environment and application in the configuration to generate a configuration file. When a component is associated with a configuration file for deployment, the system variables are automatically replaced with the actual value. This implements multi-environment use with one-time configuration through file mounting.
- **Tech Stack Management**: Based on the built-in technology stacks, ServiceStage provides technology stack management to support technology stack version customization and multiple runtime systems.
- **Deployment Source Management**: provides functions such as organization management, software repository, and image repository. Organization management is used to isolate images and assign access permissions (read, write, and manage) to different users. Image repositories are used to store and manage Docker images. Software repositories are used to store, manage, and deploy software packages.
- **Continuous Delivery**: provides functions such as viewing build projects, releasing build projects, and authorizing repositories. The software package or image package can be generated with a few clicks in a build job. In this way, the entire process of source code pull, compilation, packaging, and archiving is automatically implemented. One-click deployment can be achieved through pipeline. In this way, the entire process of source code pull, compilation, packaging, archiving, and deployment is automatically implemented. This unifies the integration environment and standardizes the delivery process. You can use repository authorization so that build projects and application

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

components can use the authorization information to access the software repository.

- **Microservice Engine**: Cloud Service Engine is a one-stop management platform for microservice solutions. After microservice components are connected to CSE, you can use service registry, service governance, and configuration management of CSE. In this way, you can focus on service development and improve product delivery efficiency and quality. You can **Binding a Microservice Engine** when setting **Cloud Service Settings** for the component.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

2 Using IAM to Grant Access to ServiceStage

2.1 Creating a User and Granting ServiceStage Permissions

You can use **Identity and Access Management (IAM)** for fine-grained permissions control for your ServiceStage. With IAM, you can:

- Create IAM users for personnel based on your enterprise's organizational structure. Each IAM user has their own identity credentials for accessing ServiceStage resources
- Grant only the permissions required for users to perform a specific task.
- Entrust a Huawei Cloud account or a cloud service to perform efficient O&M on your ServiceStage resources.

If your Huawei Cloud account does not require individual IAM users, you may skip over this section.

This section describes the procedure for granting user permissions. **Figure 2-1** shows the process flow.

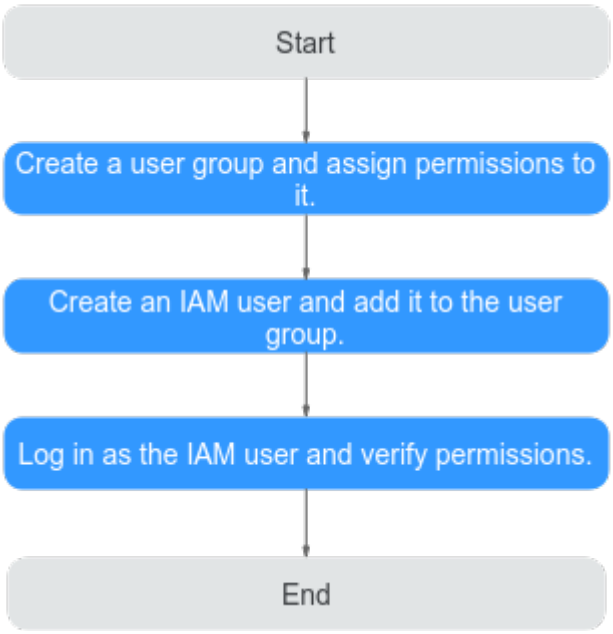
Prerequisites

Before granting permissions to user groups, learn about system-defined permissions in **System-defined permissions for ServiceStage** for ServiceStage. To grant permissions for other services, learn about all **system-defined permissions** supported by IAM.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Process Flow

Figure 2-1 Process for granting ServiceStage permissions



1. **create a user group and grant it permissions** (**ServiceStage ReadOnlyAccess** as an example).

   Create a user group on the IAM console and assign the **ServiceStage ReadOnlyAccess** permissions to the group.

2. **Create an IAM user and add it to the created user group**.

   Create a user on the IAM console and add it to the user group created in **1**.

3. **Log in as the IAM user** and verify permissions.

   Perform the following operations:

   a. Choose **Service List** > **ServiceStage**.

   b. Choose **Application Management** and click **Create Application**.

      If a message appears indicating that you have insufficient permissions to perform the operation, the **ServiceStage ReadOnlyAccess** policy is in effect.

2.2 Creating a Custom Policy

You can create custom policies to supplement the system-defined policies of ServiceStage. For details about actions supported in custom policies, see **Permissions and Supported Actions**.

To create a custom policy, choose either visual editor or JSON.

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Create a JSON policy or edit an existing one.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

For details, see **Creating a Custom Policy**. The following section contains examples of common ServiceStage custom policies.

Example Custom Policies

This procedure creates a policy that an IAM user is prohibited to create and delete a microservice engine.

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "cse:*:*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "cse:engine:create",
                "cse:engine:delete"
            ],
            "Effect": "Deny"
        }
    ]
}
```

A policy with only "Deny" permissions must be used together with other policies. If the permissions granted to an IAM user contain both "Allow" and "Deny", the "Deny" permissions take precedence over the "Allow" permissions.

After authorization, you can verify the permissions as the IAM user using the console or REST APIs.

The following uses the custom policy as an example to describe how to log in to the ServiceStage console to verify that a user is not allowed to create microservice engines.

1. Log in to Huawei Cloud as an IAM user.
   - Tenant name: Name of the Huawei Cloud account used to create the IAM user
   - IAM username and password: Username and password specified during the IAM user creation using the tenant name
2. On the **Cloud Service Engines** page, create a microservice engine. If error 403 is returned, the permissions are correct and have taken effect.

2.3 Granting Permissions to ServiceStage-Dependent Services

Granting Cluster Namespace Permissions

You can grant only common operation permissions on cluster resources to the ServiceStage user group using IAM, excluding the namespace permissions of the clusters with Kubernetes RBAC authentication enabled. Therefore, grant the namespace permissions to the clusters separately.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

For details, see **Namespace Permissions**.

Granting CTS Permissions

After the permissions are granted for ServiceStage using IAM, they do not take effect for the CTS service on which ServiceStage depends. Therefore, grant the CTS permissions separately.

For details, see **Permissions Management**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

3 Environment Management

3.1 Environment Overview

An environment is a collection of basic compute (such as cluster and ECS), network (such as ELB and EIP), and middleware (such as DCS and RDS) resources, used for component deployment and running. ServiceStage combines multiple basic resources into an environment, including development, test, pre-production, and production environments. Managing resources, and creating and deploying components by environment simplify O&M management.

For details about components, see **Component Overview**.

Environment Type Description

Table 3-1 lists the environments supported by ServiceStage. For details, see **Component Deployment Modes**.

Table 3-1 ServiceStage environment types

Environment Type	Description
VM	Applicable to VM-based deployment. Components are deployed on ECSs or custom servers using software packages.
Kubernetes	Applicable to container-based deployment. Components are deployed using container images and scheduled by Kubernetes. ServiceStage allows you to create HA and non-HA Kubernetes environments. • HA environment provides intra-city active-active HA capabilities for applications. This environment is applicable for deploying applications that require intra-city HA. Two clusters in different AZs need to be bound. • Non-HA environment is applicable for deploying applications that do not require intra-city HA. Only one cluster needs to be bound.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Environment Type	Description
VM + Kubernetes	The environment type is **VM + Kubernetes** if clusters, ECSs, custom servers, and AS groups are managed in the environment at the same time. This is applicable to creating and deploying components based on VMs or containers in the same environment.

Environment Management Process

Figure 3-1 Environment management process

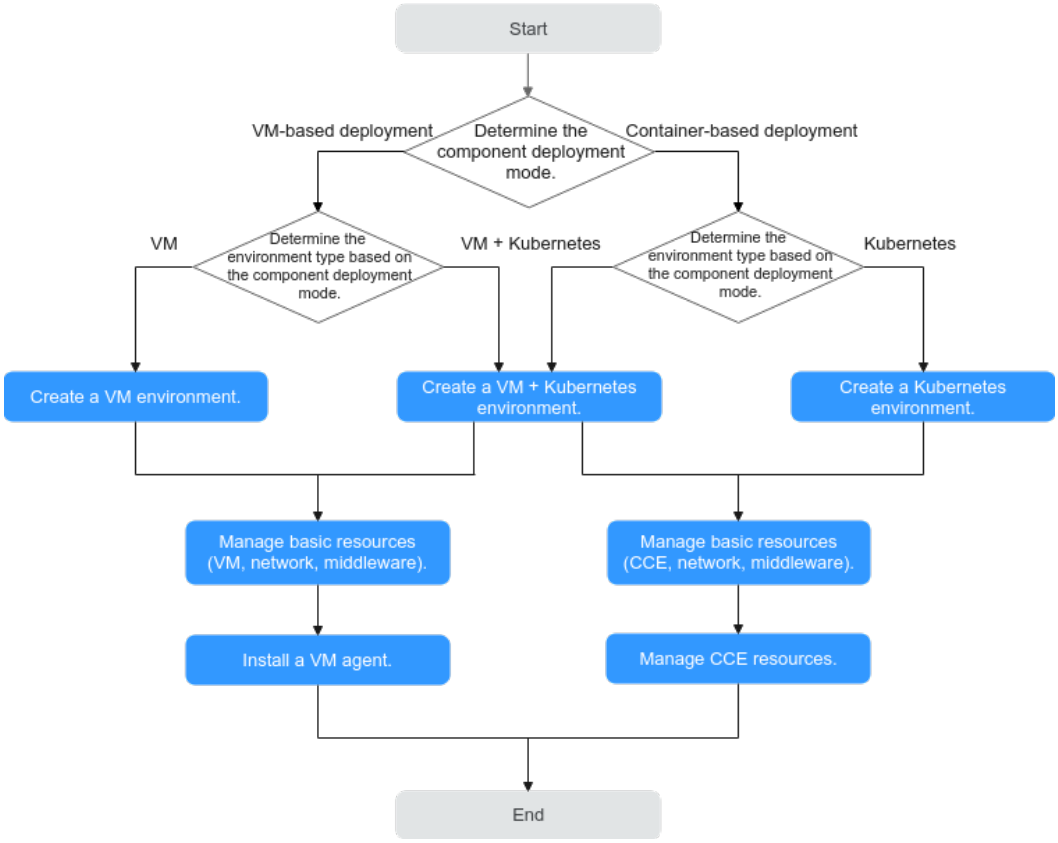


Figure 3-1 shows the environment management process. You need to determine the type of the environment to be created based on the component deployment mode, create the environment, and manage the basic resources of the environment.

3.2 Creating an Environment

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

3.2.1 Creating a VM Environment in Resource Management Mode

A maximum of 2000 environments can be created in a project.

You can manage the created compute resources (ECSs, AS groups, and custom servers), network resources, and middleware in the environment for creating and deploying components on VMs.

Creating a VM Environment in Resource Management Mode

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management** > **Create Environment** and configure the basic environment information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Environment	Environment name, which must be unique in the same enterprise project. Enter 2 to 64 case-sensitive characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
*Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**.
*Environment Type	Select **VM**. Applicable to VM-based deployment. Components are deployed on ECSs or custom servers using software packages.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*VPC	VPC where the environment resources are located. After the environment is created, the VPC cannot be modified during **Editing an Environment**. Select the VPC where the environment resource is located based on your service requirements. • To use a created VPC, select a VPC created under the current account from the drop-down list. • To use a shared VPC, select a VPC that another account shares with the current account from the drop-down list. VPC owners can share the subnets in a VPC with one or multiple accounts through **Resource Access Manager (RAM)**. Through VPC sharing, you can easily configure and manage multiple accounts' resources at low costs. For more information about VPC and subnet sharing, see **VPC Sharing**. • To use a new VPC, click **Create a VPC**. For details, see **Creating a VPC with a Subnet**.
Environment Tag	Tags are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use tags to classify them based on usage, owner, or environment. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment creation may fail. Contact your administrator to learn more about tag policies. Each environment can have a maximum of 20 tags. 1. Click **Add Tag**. The **Add Tag** dialog box is displayed. 2. Click **Add Tag**. 3. Enter the key and value of the tag. – The key contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-). – The value contains 0 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 4. Click **OK**.
Description	Environment description. Enter up to 128 characters.
*Configuration Mode	Click **Resource management** to manage created basic resources in the environment.

Step 3 Click **Create Now**.

On the **Overview** page, you can perform environment management operations such as basic resource management by referring to **Follow-up Operations**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Follow-up Operations

- **Managing Basic Resources**: After an environment is created, other basic resources need to be managed in the environment to form an environment.
- **Managing VM Agents**: After a VM environment is created, install the agent on the VM before deploying components on the VM. After the host node is managed, the management node can communicate with the host node.
- **Managing Environment Variables of an Environment**: Environment variables of an environment describe environment information and environment resource information, such as middleware credentials, connection information, and environment functions. Customers can use these variables to shield environment or resource differences, and switch components to different environments or run them using different cloud service resources without modification.
- **Editing an Environment**: Modify the name, enterprise project, and description of an environment.
- **View Environment Details**: View environment details at any time.

3.2.2 Creating a Kubernetes Environment in Resource Management Mode

A maximum of 2000 environments can be created in a project.

You can manage the created compute resources (clusters), network resources, and middleware in the environment for creating and deploying components in containers.

Creating a Kubernetes Environment in Resource Management Mode

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management** > **Create Environment** and configure the basic environment information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Environment	Environment name, which must be unique in the same enterprise project. Enter 2 to 64 case-sensitive characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**.
*Environment Type	Select **Kubernetes**. It is applicable to container-based deployment (CCE). Components are deployed using container images and scheduled by Kubernetes.
*HA Environment	• **Yes**: Create an HA environment of the Kubernetes type to provide intra-city active-active HA capabilities for applications. This environment is applicable for deploying applications that require intra-city HA. • **No**: Create a non-HA environment of the Kubernetes type to deploy applications that do not require intra-city HA.
*AZ	Select two AZs for the environment when **High Available** is set to **OK**.
*VPC	VPC where the environment resources are located. After the environment is created, the VPC cannot be modified during **Editing an Environment**. Select the VPC where the environment resource is located based on your service requirements. • To use a created VPC, select a VPC created under the current account from the drop-down list. • To use a shared VPC, select a VPC that another account shares with the current account from the drop-down list. VPC owners can share the subnets in a VPC with one or multiple accounts through **Resource Access Manager (RAM)**. Through VPC sharing, you can easily configure and manage multiple accounts' resources at low costs. For more information about VPC and subnet sharing, see **VPC Sharing**. • To use a new VPC, click **Create a VPC**. For details, see **Creating a VPC with a Subnet**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Environment Tag	Tags are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use tags to classify them based on usage, owner, or environment. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment creation may fail. Contact your administrator to learn more about tag policies. Each environment can have a maximum of 20 tags. 1. Click **Add Tag**. The **Add Tag** dialog box is displayed. 2. Click **Add Tag**. 3. Enter the key and value of the tag. – The key contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-). – The value contains 0 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 4. Click **OK**.
Description	Environment description. Enter up to 128 characters.
*Configuration Mode	Click **Resource management** to manage created basic resources in the environment.

Step 3 Click **Create Now**.

On the **Overview** page, you can perform environment management operations such as basic resource management by referring to **Follow-up Operations**.

----End

Follow-up Operations

- **Managing Cluster Resources**: After an environment is created, bind a cluster to the environment and manage the cluster before using the environment to deploy components.
- **Managing Basic Resources**: After an environment is created, other basic resources (excluding clusters) need to be managed in the environment to form an environment.
- **Managing Environment Variables of an Environment**: Environment variables of an environment describe environment information and environment resource information, such as middleware credentials, connection information, and environment functions. Customers can use these variables to shield environment or resource differences, and switch components to different environments or run them using different cloud service resources without modification.
- **Editing an Environment**: Modify the name, enterprise project, and description of an environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **View Environment Details**: View environment details at any time.

3.2.3 Creating a VM + Kubernetes Environment in Resource Management Mode

A maximum of 2000 environments can be created in a project.

You can manage the created compute resources (clusters, ECSs, AS groups, and custom servers), network resources, and middleware for creating and deploying components based on containers and VMs at the same time in the same environment.

Creating a VM + Kubernetes Environment in Resource Management Mode

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management** > **Create Environment** and configure the basic environment information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Environment	Environment name, which must be unique in the same enterprise project. Enter 2 to 64 case-sensitive characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
*Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**
*Environment Type	Select **VM + Kubernetes**. Applicable to creating and deploying components based on VMs or containers in the same environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*VPC	VPC where the environment resources are located. After the environment is created, the VPC cannot be modified during **Editing an Environment**. Select the VPC where the environment resource is located based on your service requirements. • To use a created VPC, select a VPC created under the current account from the drop-down list. • To use a shared VPC, select a VPC that another account shares with the current account from the drop-down list. VPC owners can share the subnets in a VPC with one or multiple accounts through **Resource Access Manager (RAM)**. Through VPC sharing, you can easily configure and manage multiple accounts' resources at low costs. For more information about VPC and subnet sharing, see **VPC Sharing**. • To use a new VPC, click **Create a VPC**. For details, see **Creating a VPC with a Subnet**.
Environment Tag	Tags are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use tags to classify them based on usage, owner, or environment. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment creation may fail. Contact your administrator to learn more about tag policies. Each environment can have a maximum of 20 tags. 1. Click **Add Tag**. The **Add Tag** dialog box is displayed. 2. Click **Add Tag**. 3. Enter the key and value of the tag. – The key contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-). – The value contains 0 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 4. Click **OK**.
Description	Environment description. Enter up to 128 characters.
*Configuration Mode	Click **Resource management** to manage created basic resources in the environment.

Step 3 Click **Create Now**.

On the **Overview** page, you can perform environment management operations such as basic resource management by referring to **Follow-up Operations**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Follow-up Operations

- **Managing Cluster Resources**: After an environment is created, bind a cluster to the environment for management before creating and deploying components in containers in this environment.
- **Managing Basic Resources**: After an environment is created, other basic resources (excluding clusters) need to be managed in the environment to form an environment.
- **Managing VM Agents**: After an environment is created, install the agent on the VMs managed by the environment before creating and deploying components on VMs in this environment. After the host nodes are managed, the management nodes can communicate with the host nodes.
- **Managing Environment Variables of an Environment**: Environment variables of an environment describe environment information and environment resource information, such as middleware credentials, connection information, and environment functions. You can use these variables to shield environment or resource differences, and switch components to different environments or run them using different cloud service resources without modification.
- **Editing an Environment**: Modify the name, enterprise project, and description of an environment.
- **View Environment Details**: View environment details at any time.

3.3 Managing Basic Resources in the Environment

3.3.1 Managing Basic Resources

After an environment is created, you can manage required basic resources and combine them into an environment for deploying and running components.

Prerequisites

- You have created an environment. For details, see **Creating an Environment**.
- You have created basic resources based on the environment type and site requirements by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
Compute	Clusters	Used to deploy components in container mode. Components are deployed using container images and scheduled by Kubernetes.	Clusters can be bound to a Kubernetes or VM + Kubernetes environment.	See **Binding a Cluster**.	See **CCE Billing**.
	ECSs	Used to deploy and run components based on a VM.	ECSs can be managed in a VM or VM + Kubernetes environment. The ECS must be in the same VPC as the environment and cannot be managed by other environments.	See **Purchasing an ECS in Custom Config Mode**.	See **ECS Billing**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
	Custom servers	You can manage custom servers on ServiceStage for deploying and running components based on a VM.	Custom servers can be managed in a VM or VM + Kubernetes environment. You have obtained the name, unique ID, and IP address of the custom server, and the custom server cannot be managed by other environments.	-	-

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
	AS Groups	Used to deploy and run components based on a VM.	AS groups can be managed in a VM or VM + Kubernetes environment. The AS group must be in the same VPC as the environment and cannot be managed by other environments.	See **Creating an AS Group**.	See **AS Pricing Details**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
Networking	Load Balancers	Used to access services provided by a component through the domain name using the load balancer bound with an EIP, or to upgrade the container-deployed components through ELB dark launch.	-	• Use the domain name to access an application. – See **Creating a Shared Load Balancer**. – Create an application load balancer by referring to **Creating a Dedicated Load Balancer**. • **ELB dark launch** Create an application load balancer by referring to **Creating a Dedicated Load Balancer**.	See **ELB Billing**.
	VPC EIPs	Used to access services provided by a component.	-	See **Assigning an EIP**.	See **EIP Billing**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
	VPCs	Provides a logically isolated, configurable, and manageable virtual network environment, improving cloud resource security and simplifying network deployment.	-	See **Creating a VPC with a Subnet**.	See **VPC Billing**.
	Subnet	A subnet is a CIDR block in a VPC, and is on the Layer 3 network. You can create multiple subnets in a VPC to manage cloud resources.	-	See **Creating a Subnet for an Existing VPC**.	See **VPC Billing**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
	ACL	Provides security protection for associated subnets.	-	See **Creating a Network ACL**.	See **VPC Billing**.
Middleware	DCS Instances	Used to read environment variables to obtain distributed cache information during application running.	-	See **Buying a DCS Redis Instance**.	See **DCS Billing**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
	ServiceComb Engines	Microservice engine (see **Cloud Service Engine Overview**), which is used to connect microservices running in the environment to the engine to implement microservice registry and discovery, service governance, and configuration management.	In an HA environment of the Kubernetes type, each AZ can manage only one ServiceComb engine.	See **Creating a Microservice Engine**.	• If you have purchased a ServiceStage resource package, you can create a ServiceComb engine of the corresponding version for free. For details, see **Buying a ServiceStage Resource Package**. • If you use ServiceStage in pay-

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
					per-use billing mode or create additional ServiceComb engines, see **ServiceComb Engine Billing**.
	Registry/ Configuration Center	Connects microservices running in the environment to implement microservice registry and discovery, and configuration management.	-	See **Creating a Registry/ Configuration Center**.	See .-

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
	RDS Instances	Used for persistent storage of application data.	-	See **Buying an RDS for MySQL DB Instance**.	See **RDS Billing**.
	Application Gateways	Connects microservices running in the environment to implement full-link rate limiting and API management.	In an HA environment of the Kubernetes type, each AZ can manage only one application gateway.	See .-	-
	GaussDB	Used for persistent storage of application data.	-	See **Buying a GaussDB Instance**.	See **GaussDB Billing**.
	RocketMQ	Used for application production message storage and consumption subscription.	-	See **Buying a RocketMQ Instance**.	See **RocketMQ Billing**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Description	Restrictions	Creation Method	Billing
	CSS	Uses the multi-condition search, statistics, and report capabilities of structured and unstructured text provided by the Elasticsearch cluster of Cloud Search Service (CSS) to analyze component logs in all scenarios.	-	See **Creating an Elasticsearch Cluster**.	See **CSS Billing**.

Managing Basic Resources

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Manage the required basic resources in the environment by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Operation
Compute	Clusters	See **Binding a Cluster**. Clusters (CCE and custom Kubernetes clusters) can be bound to a Kubernetes or VM + Kubernetes environment. To manage clusters in the environment, see **Managing Cluster Resources**.
	ECSs	1. Choose **Compute** > **ECSs**. 2. Click **Manage Resource**. 3. Select the ECSs to be managed. 4. Click **OK**. – ECSs can be managed in a VM or VM + Kubernetes environment. – ECSs that have been managed by other environments cannot be managed again. – If **Agent Status** of the managed ECSs indicates that the agent is missing and to install it first, install the agent. For details, see **Installing a VM Agent**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Operation
	Custom servers	1. Choose **Compute** > **Custom servers**. 2. Set **Server Name**, **ID**, and **IP** of the target custom server. – Click **Add Server** to add more custom servers to the environment. – Click **Remove** to remove a custom server from the list. 3. Click **OK**. – Custom servers can be managed in a VM or VM + Kubernetes environment. – For custom servers that are initially managed and do not have a VM agent installed (see **Installing a VM Agent**), you need to customize the **Server Name** and **ID**. Where, **Server Name** contains 1 to 128 characters. Use letters, digits, underscores (_), hyphens (-), and dots (.). **ID** contains 1 to 64 characters. Use letters, digits, underscores (_), and hyphens (-). Keep the custom **ID** properly for future operations such as re-management. – To manage a custom server with a VM agent installed (see **Installing a VM Agent**), set **ID** to the value you have defined during the first management. **Server Name** can be customized. **Server Name** contains 1 to 128 characters. Use letters, digits, underscores (_), hyphens (-), and dots (.). – The custom server to be managed must be able to access the environment where ServiceStage is deployed through network and cannot be managed by other environments.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Operation
	AS Groups	1. Choose **Compute** > **AS Groups**. 2. Click **Manage Resource**. 3. Select the AS groups to be managed. 4. Click **OK**. – AS groups can be managed in a VM or VM + Kubernetes environment. – AS groups that have been managed by other environments cannot be managed again. – If **Agent Status** of the ECSs of the managed AS groups indicates that the agent is missing and to install it first, install the agent. For details, see **Installing a VM Agent**.
Networking	Load Balancers	1. Choose **Networking** > **Load Balancers**. 2. Click **Manage Resource**. 3. Select the load balancers to be managed. 4. Click **OK**.
	VPC EIPs	1. Choose **Networking** > **VPC EIPs**. 2. Click **Manage Resource**. 3. Select the EIPs to be managed. 4. Click **OK**.
	VPCs	1. Choose **Network** > **VPCs**. 2. Click **Manage Resource**. 3. Select the VPCs to be managed. 4. Click **OK**.
	Subnet	1. Choose **Network** > **Subnet**. 2. Click **Manage Resource**. 3. Select the VPC subnets to be managed. 4. Click **OK**.
	ACL	1. Choose **Network** > **ACL**. 2. Click **Manage Resource**. 3. Select the ACLs to be managed. 4. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Operation
Middleware	DCS Instances	1. Choose **Middleware** > **DCS Instances**. 2. Click **Manage Resource**. 3. Select the DCS instances to be managed. 4. Click **Next**. 5. (Optional) Set **Redis Login Password**. If the selected DCS instances are configured with the password access mode, you need to set the login password. You can also skip this operation and configure the parameters by referring to **Follow-up Operations** after the DCS instances are successfully managed. 6. Click **OK**. After the DCS instances are successfully managed in the environment, ServiceStage writes the built-in and custom environment variables in the environment by referring to **Table 3-19** for use.
	ServiceComb Engines	1. Choose **Middleware** > **ServiceComb Engines**. 2. Click **Manage Resource**. 3. Select the ServiceComb engines to be managed. 4. Click **OK**. After the ServiceComb engines are successfully managed in the environment, ServiceStage writes the built-in environment variables in the environment by referring to **Table 3-19** for use.
	Registry/ Configuration Center	1. Choose **Middleware** > **Registry/Configuration Center**. 2. Click **Manage Resource**. 3. Select the registry/configuration centers to be managed. 4. Click **OK**. After the registry/configuration centers are successfully managed in the environment, ServiceStage writes the built-in environment variables in the environment by referring to **Table 3-19** for use.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Operation
	RDS Instances	1. Choose **Middleware** > **RDS Instances**. 2. Click **Manage Resource**. 3. Select the RDS instances to be managed. 4. Click **Next**. 5. (Optional) Set **Database Name**, **Database User Name**, and **Database Login Password**. You can also skip this operation and configure the parameters by referring to **Follow-up Operations** after the RDS instances are successfully managed. 6. Click **OK**. After the RDS instances are successfully managed in the environment, ServiceStage writes the built-in and custom environment variables in the environment by referring to **Table 3-19** for use.
	Application Gateways	1. Choose **Middleware** > **Application Gateways**. 2. Click **Manage Resource**. 3. Before managing the application gateways in the HA environment of the Kubernetes type, select the AZ where these application gateways are located. 4. Select the application gateways to be managed. 5. Click **OK**.
	GaussDB	1. Choose **Middleware** > **GaussDB**. 2. Click **Manage Resource**. 3. Select the GaussDB instances to be managed. 4. Click **Next**. 5. (Optional) Set **Database Name**, **Database User Name**, and **Database Login Password**. You can also skip this operation and configure the parameters by referring to **Follow-up Operations** after the GaussDB instances are successfully managed. 6. Click **OK**. After the GaussDB instances are successfully managed in the environment, ServiceStage writes the built-in and custom environment variables in the environment by referring to **Table 3-19** for use.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Resource Name	Operation
	RocketMQ	1. Choose **Middleware** > **RocketMQ**. 2. Click **Manage Resource**. 3. Select the RocketMQ instances to be managed. 4. Click **Next**. 5. (Optional) Set **Username** and **Password**. You can also skip this operation and configure the parameters by referring to **Follow-up Operations** after the RocketMQ instances are successfully managed. 6. Click **OK**. After the RocketMQ instances are successfully managed in the environment, ServiceStage writes the built-in and custom environment variables in the environment by referring to **Table 3-19** for use.
	CSS	1. Choose **Middleware** > **CSS**. 2. Click **Manage Resource**. 3. Select the Elasticsearch cluster to be managed. 4. Click **Next**. 5. (Optional) Set **Username** and **Password**. You can also skip this operation and configure the parameters by referring to **Follow-up Operations** after the Elasticsearch clusters are successfully managed. 6. Click **OK**. After the Elasticsearch clusters are successfully managed in the environment, ServiceStage writes the built-in and custom environment variables in the environment by referring to **Table 3-19** for use.

Step 5 Click **OK**.

After the basic resources are managed, click **Overview** and select the basic resources (for example, VPC) managed in **Step 4**. You can see that **Status** is **Available** and **Resource Mode** is **Manage**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Follow-up Operations

You can perform maintenance operations on DCS, RDS, GaussDB, RocketMQ, and CSS instances managed in the environment by referring to the following table.

Resource Type	Maintenance Operation	Description
DCS	Setting the connection password	A password has been set for the managed DCS instances. You need to enter the password. 1. On the **Overview** page, choose **Distributed Cache Service**. 2. Click **Set Password** in the **Operation** column of the target DCS instance. 3. Click **Edit** in the **Operation** column. 4. (Optional) Modify **Name**. The default value is **DISTRIBUTED_CACHE_PASSWORD**. You can change the value as required. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.). 5. Enter the value of **Variable/Variable Reference**, that is, the password for accessing the DCS instance. 6. Click **Confirm** in the **Operation** column.
	Setting environment variables of an environment	• To add an environment variable, see **Adding an Environment Variable**. • To edit an environment variable, see **Editing an Environment Variable**. • To delete an environment variable, see **Deleting an Environment Variable**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Maintenance Operation	Description
RDS	Setting the account and password	Set the name, username, and password of the database that can be accessed by the managed RDS instance. 1. On the **Overview** page, choose **Relational Database Service**. 2. Click **Set Account and Password** in the **Operation** column of the target RDS instance. 3. Set the name of the cloud database to be accessed. a. Click **Edit** in the **Operation** column of the default variable name **RELATIONAL_DATABASE_DB_NAME**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column. 4. Set the username for accessing the database in Relational Database Service. a. Click **Edit** in the **Operation** column of the default variable name **RELATIONAL_DATABASE_DB_USER**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column. 5. Set the password of the username for accessing the database in Relational Database Service. a. Click **Edit** in the **Operation** column of the default variable name **RELATIONAL_DATABASE_PASSWORD**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column.
	Setting environment variables of an environment	• To add an environment variable, see **Adding an Environment Variable**. • To edit an environment variable, see **Editing an Environment Variable**. • To delete an environment variable, see **Deleting an Environment Variable**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Maintenance Operation	Description
RocketMQ	Setting the account and password	Set the username and password for accessing the managed RocketMQ instance. 1. On the **Overview** page, choose **RocketMQ**. 2. Click **Set Account and Password** in the **Operation** column of the target RocketMQ instance. 3. Set the username for accessing the database in Relational Database Service. a. Click **Edit** in the **Operation** column of the default variable name **DISTRIBUTED_MESSAGE_ROCKETMQ_USER**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column. 4. Set the password of the access user. a. Click **Edit** in the **Operation** column of the default variable name **DISTRIBUTED_MESSAGE_ROCKETMQ_PASSWORD**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column.
	Setting environment variables of an environment	• To add an environment variable, see **Adding an Environment Variable**. • To edit an environment variable, see **Editing an Environment Variable**. • To delete an environment variable, see **Deleting an Environment Variable**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Maintenance Operation	Description
GaussDB	Setting the account and password	Set the name, username, and password of the database that can be accessed by the managed GaussDB instance. 1. On the **Overview** page, choose **GaussDB**. 2. Click **Set Account and Password** in the **Operation** column of the target RocketMQ instance. 3. Set the name of the cloud database to be accessed. a. Click **Edit** in the **Operation** column of the default variable name **PAAS_GAUSSDB_DB_NAME**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column. 4. Set the username for accessing the database in Relational Database Service. a. Click **Edit** in the **Operation** column of the default variable name **PAAS_GAUSSDB_DB_USER**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column. 5. Set the password of the username for accessing the database in Relational Database Service. a. Click **Edit** in the **Operation** column of the default variable name **PAAS_GAUSSDB_PASSWORD**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column.
	Setting environment variables of an environment	• To add an environment variable, see **Adding an Environment Variable**. • To edit an environment variable, see **Editing an Environment Variable**. • To delete an environment variable, see **Deleting an Environment Variable**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Maintenance Operation	Description
CSS	Setting the account and password	Set the username and password for accessing the managed CSS Elasticsearch cluster. 1. On the **Overview** page, choose **CSS**. 2. Click **Set Account and Password** in the **Operation** column of the target Elasticsearch cluster. 3. Set the access user name. a. Click **Edit** in the **Operation** column of the default variable name **PAAS_ELASTIC_SEARCH_USER**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column. 4. Set the password of the username for accessing the database in Relational Database Service. a. Click **Edit** in the **Operation** column of the default variable name **PAAS_ELASTIC_SEARCH_PASSWORD**. b. Modify **Name**. c. Enter **Variable/Variable Reference**. d. Click **Confirm** in the **Operation** column.
	Setting environment variables of an environment	• To add an environment variable, see **Adding an Environment Variable**. • To edit an environment variable, see **Editing an Environment Variable**. • To delete an environment variable, see **Deleting an Environment Variable**.

3.3.2 Managing Cluster Resources

3.3.2.1 Binding a Cluster

Before creating a component based on a container in the Kubernetes or VM + Kubernetes environment, bind a cluster to the environment.

- The VM + Kubernetes environment and Kubernetes non-HA environment support CCE and custom Kubernetes clusters.
- The Kubernetes HA environment supports CCE clusters.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Restrictions

- The CCE cluster must be a **CCE Turbo Cluster** or **CCE Standard Cluster**. For details, see **CCE Cluster Types**.

  If you create a CCE Turbo cluster, set SNAT rules for the cluster so that the cluster can access the public network through the NAT gateway to pull source code. Otherwise, when the source code is used to deploy a component in container-based deployment or a source code-build task is executed, the source code fails to be obtained because the public network cannot be accessed. As a result, the component deployment or build fails. For details about how to set SNAT rules for the cluster, see **Accessing the Internet from a Container**.
- If IPv6 is enabled for the VPC where the target environment is located, select a CCE cluster with IPv6 enabled. Otherwise, the Java chassis microservice registered on the exclusive microservice engine with security authentication enabled in the VPC fails to register the discovery address using IPv6. For details, see **What Should I Do If the Service Registration Fails After IPv6 Is Enabled for the Exclusive Microservice Engine with Security Authentication Enabled?**

Prerequisites

- You have created an environment. For details, see:
  - **Creating a Kubernetes Environment in Resource Management Mode**
  - **Creating a VM + Kubernetes Environment in Resource Management Mode**
- If you want to bind a CCE cluster, a CCE cluster in the **Running** state has been created, and the CCE cluster and environment must be in the same VPC.
  - For details about how to create a CCE cluster, see **Buying a CCE Standard/Turbo Cluster**. Create a CCE Turbo or CCE Standard cluster based on your service requirements.
    - For a Kubernetes HA environment, create a CCE cluster for each AZ.
    - For a Kubernetes non-HA environment or a VM + Kubernetes environment, create a CCE cluster.
  - For details about how to add a node to the CCE cluster, see **Creating a Node**.
    - CCE clusters support **Elastic Cloud Server (VM)** and **BMS** nodes.
    - For details about the OSs supported by CCE cluster nodes, see **Node OS**.
    - CCE cluster nodes support Docker and Containerd container engines. For details about relationship between node OSs and container engines, see **Container Engine**.
  - If you want to bind a custom Kubernetes cluster, a kubeconfig file in YAML format has been created. For details, see **kubeconfig File Description**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Binding a CCE Cluster

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

Step 5 Click **Bind now**.

- For a Kubernetes environment, bind a cluster to the environment based on whether the environment is an HA environment. For details, see **Table 3-2**.
- For a VM + Kubernetes environment, CCE clusters and custom Kubernetes clusters can be bound. For details about how to bind a cluster to a non-HA environment, see **Table 3-2**.

Table 3-2 Binding a cluster

HA Environment or Not	How to Bind
Yes	Only CCE clusters can be bound. In an HA environment, the AZ where the master node of the cluster to be bound is located and the AZ where the nodes running workloads in the cluster are located must be the same as the AZ selected when the cluster is bound. 1. Select an AZ from the **AZ** drop-down list. 2. Select a CCE cluster that can be bound in the AZ from the **Cluster** drop-down list. 3. Click **OK**. 4. Click **Add Cluster**. 5. Select another AZ from the **AZ** drop-down list. 6. Select a CCE cluster that can be bound in the AZ from the **Cluster** drop-down list. 7. Click **OK**. After the CCE cluster is bound, the cluster-created namespace **default**, existing namespaces, and new user-created namespaces are automatically bound. For details, see **Managing Namespaces**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

HA Environment or Not	How to Bind
No	You can bind a CCE cluster or a custom Kubernetes cluster. 1. Set **Cluster Source**. ▪ Select **CCE cluster** and go to **Step 5.b**. ▪ Select **Custom Kubernetes cluster** and go to **Step 5.d**. 2. Select the cluster to be bound from the **Cluster** drop-down list. 3. (Optional) Select the namespace to be bound to the CCE cluster from the **Namespace** drop-down list, and go to **Step 5.f**. ▪ If no namespace is selected, after the CCE cluster is bound, the cluster-created namespace **default**, existing namespaces, and new user-created namespaces are automatically bound. ▪ If a namespace is selected, the selected namespace is bound after the CCE cluster is bound. For details, see **Managing Namespaces**. 4. Enter a cluster name. Enter 2 to 64 case-sensitive characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). 5. Click **Import YAML File** to import the kubeconfig file in YAML format. 6. Click **OK**.

If you have performed **Installing Sermant Injector** for the cluster, manage the ServiceComb engine or registry/configuration center associated with the installed Sermant Injector plug-in in the environment. Otherwise, the Sermant Injector plug-in will be unavailable in this environment.

For details, see **Managing Basic Resources**.

- If no CCE cluster is created, go to the CCE console as prompted, create a CCE cluster by referring to **Prerequisites**, and bind the CCE cluster.

----End

Follow-up Operations

- On the **Overview** page, click the **Node List** tab to view details about each node in the CCE cluster.
- On the **Overview** page, click the **Namespace** tab to manage cluster namespaces by referring to **Managing Namespaces**.
- On the **Overview** page, click the **ConfigMaps** tab to manage cluster configuration items by referring to **Managing Configuration Items**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- On the **Overview** page, click the **Secrets** tab to manage cluster secrets by referring to **Managing Secrets**.
- On the **Overview** page, click **View Resource Details** to view cluster details on the CCE console.
- On the **Overview** page, you can edit the custom Kubernetes cluster bound to a Kubernetes non-HA environment or a VM+Kubernetes environment by referring to **Editing a Bound Custom Kubernetes Cluster**.
- On the **Plug-in Management** page, you can manage cluster plug-ins by referring to **Managing Plug-ins**.
- If the cluster bound to the environment is no longer used, unbind the cluster by referring to **Unbinding a Cluster**.

kubeconfig File Description

Only use the kubeconfig file from reliable sources to prevent malicious code execution or file exposure.

It is a common way to manage a native Kubernetes cluster using the kubeconfig file. The kubeconfig file contains the configuration information required for the client to communicate with the Kubernetes cluster, including the cluster information, user authentication information, and context information.

Example kubeconfig file structure:

```
apiVersion: v1
kind: Config
preferences: {}
current-context: dev-context
clusters:
- name: my-cluster
  cluster:
    server: https://api.example.com:6443
    certificate-authority-data: LS0tLS1CRUd...  # Base64-encoded CA certificate
users:
- name: admin-user
  user:
    client-certificate-data: LS0tLS1CRUd...    # Base64-encoded client certificate
    client-key-data: LS0tLS1CRUd...            # Base64-encoded client private key
contexts:
- name: dev-context
  context:
    cluster: my-cluster
    user: admin-user
    namespace: default
```

For details, see **Table 3-3**.

Table 3-3 Parameters in the kubeconfig file

Parameter	Mandatory	Type	Description	Example Value
apiVersion	Yes	String	API version of the kubeconfig file.	v1
kind	Yes	String	Resource type of the kubeconfig file. The value is fixed at **Config**.	Config

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Mandatory	Type	Description	Example Value
preferences	No	Object	User preference settings (such as the output format and color).	{}
current-context	No	String	Name of the current context.	dev-context
clusters	Yes	Array of objects	Cluster list, which defines the cluster address and authentication information. For details, see **Table 3-4**.	-
users	Yes	Array of objects	User list, which defines the user authentication mode (such as certificate, token, username and password, and plug-in). For details, see **Table 3-6**.	-
contexts	Yes	Array	Context list, which associates clusters, users, and default namespaces. For details, see **Table 3-9**.	-

Table 3-4 clusters

Parameter	Mandatory	Type	Detail	Example Value
name	Yes	String	Unique cluster name.	my-cluster
cluster	Yes	Array	Cluster connection information. For details, see **Table 3-5**.	-

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 3-5 cluster

Parameter	Mandatory	Type	Detail	Example Value
server	Yes	String	API server address of the cluster, which must contain the protocol and port number and can be accessed from the public network.	https://api.example.com:6443
certificate-authority	No	String	Path of the CA certificate file. Either this parameter or **certificate-authority-data** must be specified.	etc/kubernetes/pki/ca.crt
certificate-authority-data	No	String	Base64-encoded CA certificate data. Either this parameter or **certificate-authority** must be specified.	LS0tLS1CRUd...
insecure-skip-tls-verify	No	Boolean	Whether to skip TLS verification. This parameter is used only in the test environment.	false

Table 3-6 users

Parameter	Mandatory	Type	Detail	Example Value
name	Yes	String	Unique user name (customized).	admin-user
user	No	Array of objects	Defines the credential and mode for user identity authentication. Multiple authentication mechanisms (such as certificates, tokens, and dynamic plug-ins) are supported. For details, see **Table 3-7**.	-

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 3-7 user

Parameter	Mandatory	Type	Detail	Example Value
client-certificate	No	String	Path of the client certificate file. Either this parameter or **client-certificate-data** must be specified.	/path/to/client.crt
client-certificate-data	No	String	Base64-encoded client certificate data. Either this parameter or **client-certificate** must be specified.	LS0tLS1CRUd...
client-key	No	String	Path of the client private key file. Either this parameter or **client-key-data** must be specified.	/path/to/client.key
client-key-data	No	String	Base64-encoded client private key data. Either this parameter or **client-key** must be specified.	LS0tLS1CRUd...
token	No	String	Static token (bearer token authentication)	eyJhbGciOiJSUzI1NiIsIm...
username	No	String	Basic authentication username.	admin
password	No	String	Basic authentication password.	secret
exec	No	Object	Dynamic authentication plug-in configuration. For details, see **Table 3-8**.	-

Table 3-8 exec

Parameter	Mandatory	Type	Detail	Example Value
apiVersion	Yes	String	API version of the exec plug-in.	client.authentication.k8s.io/v1beta1

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Mandatory	Type	Detail	Example Value
command	Yes	String	Path of the executable command.	/usr/bin/aws-iam-authenticator
args	No	Array of String	Command parameter.	["token", "-i", "my-cluster"]
env	No	Object	Environment variable list.	{name: "AWS_PROFILE", value: "prod"}

Table 3-9 contexts

Parameter	Mandatory	Type	Detail	Example Value
name	Yes	String	Unique name of the context (customized).	dev-context
context	Yes	Array	Binds a cluster, user, and namespace to define a specific operating environment. You can switch between different contexts to quickly switch between different clusters, users, or namespaces. For details, see **Table 3-10**.	-

Table 3-10 context

Parameter	Mandatory	Type	Detail	Example Value
cluster	Yes	String	Name of the associated cluster. The value must be the same as **name** in **Table 3-4**.	my-cluster
user	Yes	String	Name of the associated user. The value must be the same as **name** in **Table 3-6**.	admin-user

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Mandatory	Type	Detail	Example Value
namespace	No	String	Default namespace. If this parameter is not specified, **default** is used.	default

Editing a Bound Custom Kubernetes Cluster

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

Step 5 Click **Edit Cluster**.

Step 6 Change the **Cluster Name**.

Enter 2 to 64 case-sensitive characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).

Step 7 Click **Import YAML File** to import the kubeconfig file in YAML format.

Step 8 Click **OK**.

----End

3.3.2.2 Managing Namespaces

A namespace is a collection of resources and objects. Multiple namespaces can be created in a single cluster with data isolated from each other. This enables namespaces to share the services of the same cluster without affecting each other. For example, you can deploy workloads in a development environment into one namespace, and deploy workloads in a testing environment into another namespace.

Table 3-11 describes the namespace types.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 3-11 Namespace types

Type	Description
Created by a cluster by default	When a cluster is started, the **default**, **kube-public**, **kube-system**, and **kube-node-lease** namespaces are created by default. • **default**: All objects for which no namespace is specified are allocated to this namespace. • **kube-public**: Resources in this namespace can be accessed by all users (including unauthenticated users) to deploy public add-ons and container charts. • **kube-system**: All resources created by Kubernetes are in this namespace. • **kube-node-lease**: Each node has an associated Lease object in this namespace. The object is periodically updated by the node.
Created by a user	You can create namespaces as required. For example, you can create one namespace for a development environment, one for a joint debugging environment, and one for a testing environment. You can also create one namespace for login services and one for game services.

This section describes how to perform **Creating a Namespace**, **Binding a Namespace**, **Managing Namespace Resource Quotas**, **Unbinding a Namespace**, and **Deleting a Namespace** on clusters bound to a Kubernetes or VM + Kubernetes environment.

Restrictions

CCE cluster 1.11.7-r2 and later support Kubernetes RBAC authorization and namespace permission configuration. You can regulate users' or user groups' access to Kubernetes resources in a single namespace based on their Kubernetes RBAC roles. For details, see **Namespace Permissions**.

If a CCE cluster 1.11.7-r2 or later is bound in the environment and namespace permissions are configured, you must have the administrator permission (cluster-admin) for all namespaces in the cluster or the development permission (admin) for a single namespace to be operated to manage namespaces.

Creating a Namespace

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

Step 5 Go to the **Namespace** page.

- For an HA environment, click the cluster that has been bound in the environment and click the **Namespace** tab.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For a non-HA environment, click the **Namespace** tab.

Step 6 Click **Create Namespace** and set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Namespace	Unique namespace name. Enter 1 to 63 characters. Start with a lowercase letter and end with a lowercase letter or digit. Only use lowercase letters, digits, and hyphens (-).
Description	Namespace description. Enter up to 200 characters.

Step 7 Click **OK**.

- In an HA environment, the new namespace is automatically bound to the CCE cluster.
- In a non-HA environment, if a namespace is selected when **Binding a Cluster**, the new namespace can be bound to a CCE cluster only after **Binding a Namespace** is performed.
- In a non-HA environment, if no namespace is selected when **Binding a Cluster**, the new namespace is automatically bound to the CCE cluster.

----End

Binding a Namespace

Only the cluster-created namespace **default** and user-created namespaces can be bound.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

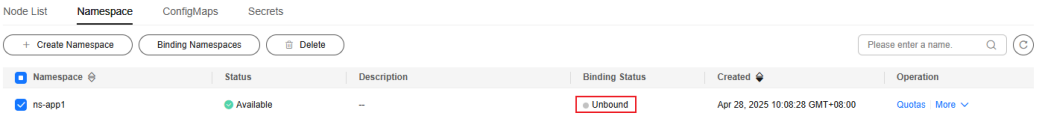
Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

Step 5 Go to the **Namespace** page.

- For an HA environment, click the cluster that has been bound in the environment and click the **Namespace** tab.
- For a non-HA environment, click the **Namespace** tab.

Step 6 Select the namespace whose **Binding Status** is **Unbound**.

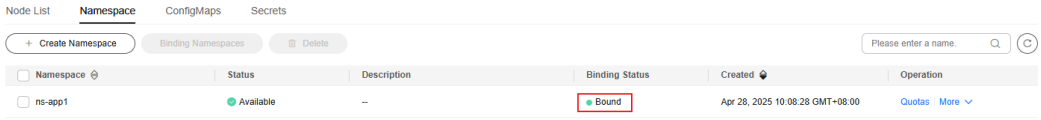


Step 7 Click **Binding Namespaces**.

Step 8 Click **OK**.

After the namespace is bound, the status changes from **Unbound** to **Bound**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.



----End

Managing Namespace Resource Quotas

By default, pods running in a CCE cluster can use the CPUs and memory of a node without restrictions. This means that pods in a namespace may exhaust all resources of the cluster.

Kubernetes provides namespaces for you to group workloads in a cluster. By setting resource quotas for each namespace, you can prevent resource exhaustion and ensure cluster reliability. You can configure quotas for resources such as CPU, memory, and the number of pods in a namespace. For more information, see **Resource Quotas**.

Only the cluster-created namespace **default** and user-created namespaces support namespace resource quota management.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

Step 5 Go to the **Namespace** page.

- For an HA environment, click the cluster that has been bound in the environment and click the **Namespace** tab.
- For a non-HA environment, click the **Namespace** tab.

Step 6 Click **Resource Quota Manager** in the **Operation** column of the target namespace.

In the displayed **Resource Quota Manager** dialog box, view the resource types, total resource quotas, and accumulated quota usage in the namespace.

Step 7 Click **Edit Quota** and set the total quota of each resource type.

- If the usage of a resource type is not limited, leave it blank.
- If the usage of a resource type is limited, enter an expected integer ranging from 1 to 9,007,199,254,740,992.
  - Accumulated quota usage includes the default resources created by the cluster, such as the Kubernetes service (which can be viewed using kubectl) created under the **default** namespace. Therefore, you are advised to set a resource quota slightly greater than expected.
  - If the total CPU or memory quota in a namespace is limited, you must set the maximum and minimum number of CPU cores and memory (GiB) that can be used by the component when setting resources for the container-deployed component of this namespace in **Creating and**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Deploying a Component and **Upgrading a Single Component**. Otherwise, this operation will fail.

- If the total quota of other resource types in a namespace is limited and the remaining usage of this resource type does not meet requirements, the container-deployed components of this namespace will fail to be deployed.

Step 8 Click **OK**.

On the **Resource Quota Manager** page, view the total resource quota and accumulated quota usage of the resource type whose quota has been reset.

----End

Unbinding a Namespace

After a namespace is unbound, it cannot be used when **Creating and Deploying a Component** based on a container in the the environment where the namespace is located.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

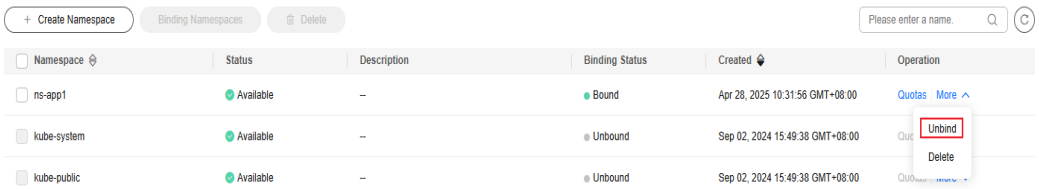
Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

Step 5 Go to the **Namespace** page.

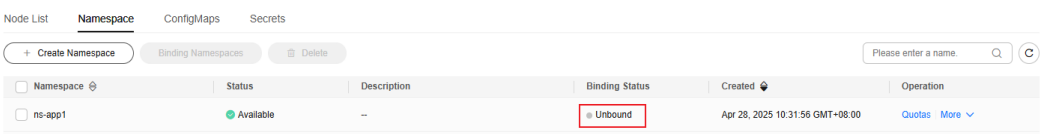
- For an HA environment, click the cluster that has been bound in the environment and click the **Namespace** tab.
- For a non-HA environment, click the **Namespace** tab.

Step 6 Select the namespace whose **Binding Status** is **Bound**, and click **More**> **Unbind** in the **Operation** column.



Step 7 Click **Save**.

After the namespace is unbound, the status changes from **Bound** to **Unbound**.



----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Deleting a Namespace

- Deleting a namespace will delete all resources (such as workloads and configuration items) in the namespace. As a result, **Upgrading a Single Component**, **Upgrading Components in Batches**, or **Rolling Back a Component** may fail for components running in the namespace.
- The namespaces (for example, **default**) created by the cluster by default cannot be deleted.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

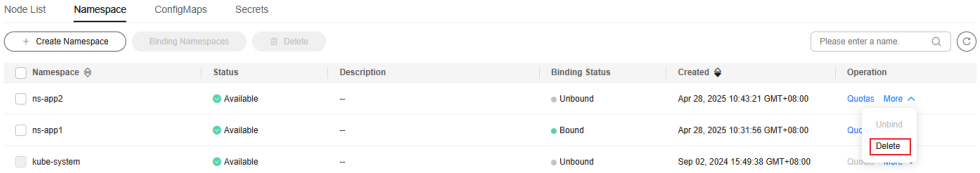
Step 5 Go to the **Namespace** page.

- For an HA environment, click the cluster that has been bound in the environment and click the **Namespace** tab.
- For a non-HA environment, click the **Namespace** tab.

Step 6 Delete the namespace.

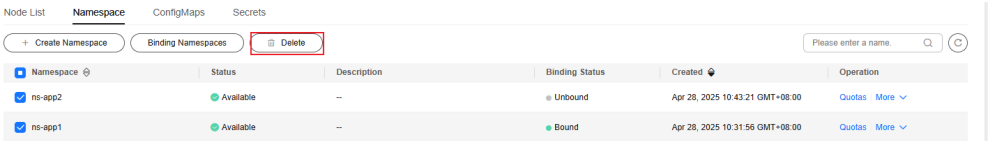
- To delete a single namespace, locate the target namespace and click **Delete** in the **Operation** column.

  Figure 3-2 Deleting a single namespace

- To delete namespaces in batches, select the target namespaces and click **Delete** above the namespaces.

  Figure 3-3 Deleting namespaces in batches

Step 7 In the displayed dialog box, enter **DELETE** and click **OK**.

----End

3.3.2.3 Managing Configuration Items

Configuration items (ConfigMaps) are user-defined resources that store application configurations. They can be used as files or environment variables in applications.

Configuration items allow you to decouple configuration files from images to enhance the portability of applications.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Benefits of configuration items:

- Manage configurations for different environments and services.
- Deploy an application in different environments using the same configuration. You can maintain configuration files in multiple versions, which makes it easy to update and roll back applications.
- Quickly import configurations in the form of files to containers.

This section describes how to **create**, **view, update, and delete configuration items** for the cluster bound to a Kubernetes or VM + Kubernetes environment.

Restrictions

Configuration items must be associated with namespaces. CCE cluster 1.11.7-r2 and later support Kubernetes RBAC authorization and namespace permission configuration. You can regulate users' or user groups' access to Kubernetes resources in a single namespace based on their Kubernetes RBAC roles. For details, see **Namespace Permissions**.

If a CCE cluster 1.11.7-r2 or later is bound in the environment and namespace permissions are configured, you must have the administrator permission (cluster-admin) for all namespaces in the cluster or the development permission (admin) for a single namespace to be operated to manage namespaces.

Prerequisites

A cluster has been bound to the environment. For details, see **Binding a Cluster**.

Creating a Configuration Item

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

Step 5 Go to the **ConfigMap** page.

- For a non-HA environment, click the **ConfigMap** tab.
- For an HA environment, click the bound cluster and click the **ConfigMap** tab.

Step 6 Click **Create Configuration Item**.

ServiceStage allows you to create configuration items in **Visualization** or **YAML** mode.

- Method 1: Visualization

  Configure the configuration item by referring to **Table 3-12**. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 3-12 Parameters for creating a configuration item in visualization mode

Parameter	Description
*Configuration Name	Configuration item name, which must be unique in a namespace. Enter 4 to 24 characters. Start with a lowercase letter and end with a lowercase letter or digit. Only use lowercase letters, digits, or hyphens (-).
*Cluster	Select the cluster bound to the environment.
*Namespace	Namespace to which the configuration item belongs. – Select an existing namespace: Select the cluster-created namespace **default** or a user-created namespace. For details about namespace types, see **Managing Namespaces**. – Use a new namespace: Click **Create Namespace** and create a namespace by referring to **Creating a Namespace**.
Description	Configuration item description. Enter 0 to 255 characters.
Configuration Data	Configuration data to be used in applications or used to store configuration data. **Key** indicates a file name, and **Value** indicates the content in the file. 1. Click **Add Data**. 2. Enter the key and value. ▪ The key contains 1 to 63 characters. Only use digits, letters, dots (.), hyphens (-), and underscores (_). ▪ The value contains 0 to 1,048,576 characters.
Configuration Labels	Labels are attached to objects such as applications, nodes, and services in key-value pairs. Labels define the identifiable attributes of these objects and are used to manage and select the objects. 1. Click **Add Label**. 2. Enter the key and value. ▪ The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, dots (.), hyphens (-), underscores (_), and slashes (/). ▪ The value contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, dots (.), hyphens (-), and underscores (_).

- Method 2: YAML

  To create a configuration item by uploading a file, ensure that a configuration item resource file in YAML format has been created and the file size is less

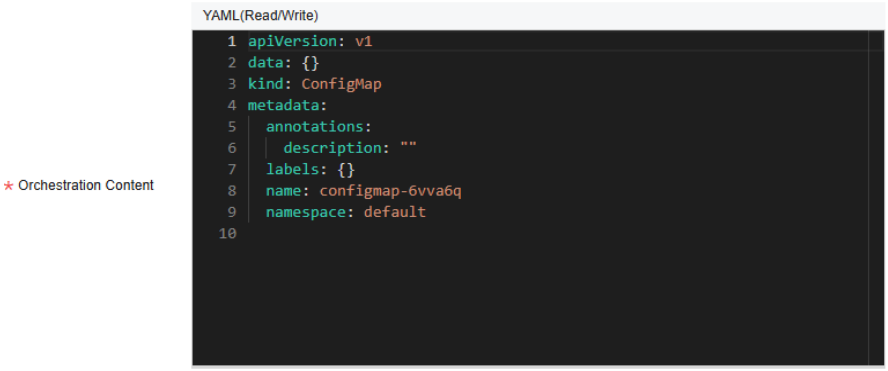
 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

than 1 MB. For details, see **Requirements for the Configuration Item Resource File**.

a. Select a cluster from the **Cluster** drop-down list.

b. Use either of the following methods to set the configuration item resource file:

- Click **Upload File**, select the configuration item resource file created locally, click **Open**, and wait until the file is uploaded.
- Write or modify the ConfigMap resource file in **Orchestration Content**.



Step 7 Click **Create ConfigMap**.

After the configuration item is created, it is displayed in the configuration item list.

----End

Follow-up Operations

After a configuration item is created, you can search for, view, update, and delete the configuration item by referring to **Table 3-13**.

- After a configuration item is deleted, the component to which the configuration item is mounted or referenced in environment variables may fail to be started after being upgraded.
- System configuration items can only be viewed.

Table 3-13 Configuration item management operations

Operation	Description
Searching for a configuration item	1. Select the namespace to which the configuration item belongs from the namespace drop-down list. 2. Enter a configuration item name in the search box.
Viewing a configuration item	Click **Show YAML** in the **Operation** column of the target configuration item to view the content of the YAML file of the configuration item.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Editing a configuration item	To update **data**, **labels**, and **description** in a configuration item in YAML mode, perform the following operations: 1. Click **Edit YAML** in the **Operation** column of the target configuration item. For details about the configuration item YAML file, see **Requirements for the Configuration Item Resource File**. You can use either of the following methods to edit a YAML file: • Edit the YAML file of the configuration item to be operated in the editing area. • Click **Import YAML File** to import the edited YAML file of the target component. 2. Click **Save**.
Modifying a configuration item	1. Click **More** > **Update** in the **Operation** column of the target configuration item. 2. Modify **data**, **labels**, and **description** of the configuration item according to the **Table 3-12**. 3. Click **Update Configuration Item**.
Deleting a configuration item	• Single deletion 1. Click **More** > **Delete** in the **Operation** column of the target configuration item. 2. In the displayed dialog box, click **OK**. • Batch deletion 1. Select the configuration items to be deleted. 2. Click **Delete Configuration Item**. 3. In the displayed dialog box, click **OK**.

Requirements for the Configuration Item Resource File

A configuration item resource file should be in YAML format, and the file size cannot exceed 1 MB.

Example:

```
apiVersion: v1
data: {}
kind: ConfigMap
metadata:
  annotations:
    description: ''
  labels: {}
  name: configmap-ww8qkl
  namespace: cse
```

For details about key parameters, see **Table 3-14**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 3-14 Key parameters of a configuration item

Parameter	Description
apiVersion	Fixed value: **v1**.
kind	Fixed value: **ConfigMap**.
metadata.name	Configuration item name, which can be customized. The configuration item name must be unique in a namespace. Enter 4 to 24 characters. Start with a lowercase letter and end with a lowercase letter or digit. Only use lowercase letters, digits, or hyphens (-).
data	Configuration item data in key-value pair format. • The key contains 1 to 63 characters. Only use digits, letters, dots (.), hyphens (-), and underscores (_). • The value contains 0 to 1,048,576 characters.

3.3.2.4 Managing Secrets

Secrets are user-defined resources that store authentication and sensitive information such as application keys. They can be used as files or environment variables in applications.

This section describes how to **create**, **view, update, and delete secrets** for the cluster bound to a Kubernetes or VM + Kubernetes environment.

Prerequisites

A cluster has been bound to the environment. For details, see **Binding a Cluster**.

Creating a Secret

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Clusters** from **Compute**.

Step 5 Go to the **Secret** page.

- For a non-HA environment, click the **Secret** tab.
- For an HA environment, click the bound CCE cluster and click the **Secret** tab.

Step 6 Click **Create**.

ServiceStage allows you to create secrets in **Visualization** or **YAML** mode.

- Method 1: Visualization Configure the parameters by referring to **Table 3-15**. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 3-15 Parameters for creating a secret in visualization mode

Parameter	Description
*Name	Secret name, which must be unique in the same namespace. Enter 4 to 24 characters. Start with a lowercase letter and end with a lowercase letter or digit. Only use lowercase letters, digits, or hyphens (-).
*Cluster	Select the cluster bound to the environment.
*Namespace	Namespace to which the secret belongs. – Select an existing namespace: Select the cluster-created namespace **default** or a user-created namespace. For details about namespace types, see **Managing Namespaces**. – Use a new namespace: Click **Create Namespace** and create a namespace by referring to **Creating a Namespace**.
Description	Secret description. Enter 0 to 255 characters.
*Secret Type	Select the type of the secret to be created based on service requirements. – **Opaque**: general secret type. If the secret type is not explicitly set in the secret configuration file, the default secret type is **Opaque**. – **kubernetes.io/dockerconfigjson**: a secret that stores the authentication information required for pulling images from a private repository. – **IngressTLS**: a secret that stores the certificate required by ingresses (layer-7 load balancing services). – **Other**: Enter a secret type that is none of the above. Enter 4 to 24 characters.
*Repository Address	This parameter is valid only when **Secret Type** is set to **kubernetes.io/dockerconfigjson**. Enter the image repository address.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Secret Data	Value of the **data** field in the application secret file. – If the secret type is **Opaque**, enter the key and value. The key contains 1 to 63 characters. Only use digits, letters, dots (.), hyphens (-), and underscores (_). The value contains 1 to 1,048,576 characters after Base64 encoding. For details, see **Base64 Encoding**. Click **Add Data** to add secret data. – If the secret type is **kubernetes.io/dockerconfigjson**, enter the username and password. – If the secret type is **IngressTLS**, click **Upload File** and upload the certificate file and private key file. The certificate file must be less than 1 MB and in .crt or .cer format. The private key file must be less than 1 MB and in .key or .pem format. – If the secret type is **Other**, enter the key and value. The key contains 1 to 63 characters. Only use digits, letters, dots (.), hyphens (-), and underscores (_). The value contains 1 to 1,048,576 characters after Base64 encoding. For details, see **Base64 Encoding**. Click **Add Data** to add secret data.
Secret Label	Labels that you want to attach to various objects (such as applications, nodes, and services) in the form of key-value pairs. Labels define the identifiable attributes of these objects and are used to manage and select the objects. 1. Click **Add**. 2. Enter the key and value. The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, dots (.), hyphens (-), underscores (_), and slashes (/). The value contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, dots (.), hyphens (-), and underscores (_).

- Method 2: YAML

  To create a secret by uploading a file, ensure that a secret resource file in YAML format has been created and the file size is less than 1 MB. For details, see **Secret Resource File Configuration Example**.

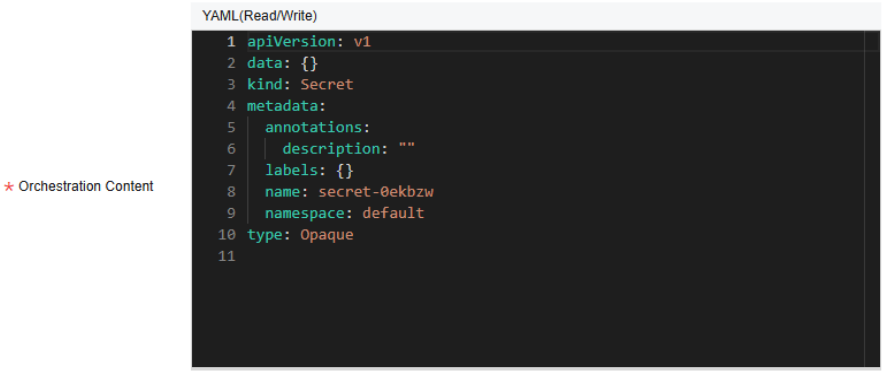
  a. Select a cluster from the **Cluster** drop-down list.

  b. Use either of the following methods to set the configuration item resource file:

     - Click **Upload File**, select the created secret file, and click **Open**. Wait until the file is uploaded.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Write or modify the secret resource file in **Orchestration content**.



Step 7 Click **Create Secret**.

The new secret is displayed in the secret list.

----End

Follow-up Operations

After a secret is created, you can search for, view, update, and delete the secret by referring to **Table 3-16**.

- After a secret is deleted, the component to which the secret is mounted or referenced in environment variables may fail to be started after being upgraded.
- The secret list contains system secrets, which can only be viewed and cannot be modified or deleted.

Table 3-16 Secret management operations

Operation	Description
Searching for a secret	1. Select the namespace to which the secret belongs from the namespace drop-down list. 2. Enter a secret name in the search box.
Viewing a secret	Click **Show YAML** in the **Operation** column of the target secret to view the content of the YAML file of the secret.
Updating a secret	1. Click **Update** in the **Operation** column of the target secret. 2. Modify the information according to **Table 3-15**. 3. Click **Modify Secret**.
Deleting a secret	1. Click **Delete** in the **Operation** column of the target secret. 2. In the displayed dialog box, click **OK**.
Deleting secrets in batches	1. Select the secrets to be deleted. 2. Click **Delete Key**. 3. In the displayed dialog box, click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

3.3.2.5 Managing Plug-ins

3.3.2.5.1 Managing Sermant Injector

Sermant Agent is automatically mounted to container-deployed Spring Cloud applications through the Sermant Injector plug-in. These applications are connected to the ServiceComb engine or registry/configuration center with security authentication disabled through Sermant Agent. For details about Sermant Agent, see **Sermant-agent User Manual**.

Compared with **Spring Cloud Huawei**, Sermant Agent allows you to access the engine and use functions such as application registry and discovery, full-link dark launch, graceful startup and shutdown, and label routing without modifying application code. However, microservice governance is not supported.

Use either Sermant Agent or **Spring Cloud Huawei** to connect Spring Cloud applications to the exclusive ServiceComb engine or registry/configuration center. Do not use them at the same time to avoid conflicts.

Before using Sermant Agent to connect a Spring Cloud application to an exclusive ServiceComb engine or registry/configuration center, **Installing Sermant Injector**.

Prerequisites

- A CCE cluster later than 1.15 has been bound in the environment. For details, see **Binding a Cluster**.
- If the bound CCE cluster has the **cse** namespace, the namespace cannot contain the configuration item sermant-injector, key sermant-injector-secret, workload sermant-injector, or service sermant-injector. Otherwise, the plug-in installation will fail.
  - For details, see **Managing Namespaces**.
  - For details about configuration items, see **Managing Configuration Items**.
  - For details about secrets, see **Managing Secrets**.
  - For details about workloads, see **Overview**.
  - For details about services, see **Overview**.
- An exclusive ServiceComb engine 2.4.0 or later or registry/configuration center with security authentication disabled has been managed in the environment. For details, see **Managing Basic Resources**.

Installing Sermant Injector

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Sermant Injector** under **Addons**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For a non-HA environment, click **install** and then go to **Step 6**.
- For an HA environment, select the AZ where the CCE cluster on which the Sermant Injector plug-in is to be installed is located, click **install**, and go to **Step 6**.

Step 6 Set parameters by referring to the following table.

Parameter	Description
CCE Cluster	By default, a bound CCE cluster in the current environment is selected.
Server Type	Select the same CPU architecture as the master node of the selected CCE cluster. • x86_64 • AArch64
Associated Microservice Engine	Select the exclusive ServiceComb engine or registry/configuration center for installing the plug-in.

Step 7 Click **OK**.

- After the Sermant Injector plug-in is successfully installed, the content listed in the following table is created.

Name	Description
Namespace	Create a namespace named cse in the CCE cluster selected in **Step 6**. **NOTE** If the CCE cluster selected in **Step 6** already has a namespace named cse, the namespace will be reused when the Sermant Injector plug-in is installed.
Configuration item	In namespace **cse** of the CCE cluster selected in **Step 6**, create a configuration item named sermant-injector.
Key	In namespace **cse** of the CCE cluster selected in **Step 6**, create a key named sermant-injector-secret.
Workload	In namespace **cse** of the CCE cluster selected in **Step 6**, create a workload named sermant-injector.
Service	In namespace **cse** of the CCE cluster selected in **Step 6**, create a service named sermant-injector.

- If the Sermant Injector plug-in fails to be installed, try again by referring to **Reinstalling Sermant Injector**.
- If a CCE cluster with the Sermant Injector plug-in installed is bound to another environment, manage the exclusive ServiceComb engine or registry/configuration center associated with the Sermant Injector plug-in in that environment. Otherwise, the Sermant Injector plug-in will be unavailable in that environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

For details about how to manage exclusive ServiceComb engines, see **Managing Basic Resources**.

----End

Upgrading Sermant Injector

Upgrade the Sermant Injector plug-in to the latest version.

Sermant Agent cannot be mounted during the Sermant Injector load upgrade. As a result, the service instances restarted during this period may fail to be connected to the ServiceComb engine or registry/configuration center.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Sermant Injector** under **Addons**.

- For a non-HA environment, click **Upgrade**.
- For an HA environment, click the AZ tab of the CCE cluster where the Sermant Injector plug-in is to be upgraded, and click **Upgrade**.

Step 6 In the displayed dialog box, click **OK**.

----End

Restarting the Sermant Injector Load

Restart the Sermant Injector load.

Sermant Agent cannot be mounted during the Sermant Injector load restart. As a result, the service instances restarted during this period may fail to be connected to the ServiceComb engine or registry/configuration center.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Sermant Injector** under **Addons**.

- For a non-HA environment, click **Restart**.
- For an HA environment, click the AZ tab of the CCE cluster where the Sermant Injector plug-in is to be restarted, and click **Restart**.

Step 6 In the displayed dialog box, click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Reinstalling Sermant Injector

Reinstall Sermant Injector if it fails to be installed.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Sermant Injector** under **Addons**.

- For a non-HA environment, click **Redeploy**.
- For an HA environment, click the AZ tab of the CCE cluster where the Sermant Injector plug-in fails to be installed, and click **Redeploy**.

----End

Deleting Sermant Injector

After the installed Sermant Injector plug-in is deleted, Sermant Agent cannot be used to connect microservice instances to the ServiceComb engine or registry/configuration center.

If the Sermant Injector plug-in is deleted, the content listed in the following table is deleted.

Name	Description
Configuration item	Configuration item sermant-injector created in namespace **cse** of the CCE cluster selected during **Installing Sermant Injector**. For details about configuration items, see **Managing Configuration Items**.
Key	Secret sermant-injector-secret created in namespace **cse** of the CCE cluster selected during **Installing Sermant Injector**. For details about secrets, see **Managing Secrets**.
Workload	Workload sermant-injector created in namespace **cse** of the CCE cluster selected during **Installing Sermant Injector**. For details about workloads, see .
Service	Service sermant-injector created in namespace **cse** of the CCE cluster selected during **Installing Sermant Injector**. For details about services, see .

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Sermant Injector** under **Addons**.

- For a non-HA environment, click **Delete**.
- For an HA environment, click the AZ tab of the CCE cluster where the Sermant Injector plug-in is to be deleted, and click **Delete**.

Step 6 Click **OK**.

----End

3.3.2.5.2 Managing Metrics Server

Metrics Server is an aggregator for monitoring data of core cluster resources. For details about the official projects and documents of the Metrics Server community, see **https://github.com/kubernetes-sigs/metrics-server**.

After installing Metrics Server, you can create HPA policies. For details, see **Managing HPA Policies**.

Prerequisites

A CCE cluster later than 1.15 has been bound in the environment. For details, see **Binding a Cluster**.

Installing Metrics Server

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Metrics Server** under **Addons**.

- For a non-HA environment, click **install** and then go to **Step 6**.
- For an HA environment, select the AZ where the cluster in which the Metrics Server plug-in is to be installed is located, and go to **Step 6**.

Step 6 Set parameters by referring to the following table.

Parameter	Description
CCE Cluster	By default, a managed cluster in the current environment is selected.
Server Type	Select the same CPU architecture as the master node of the selected cluster. • x86_64 • AArch64

Step 7 Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- After the Metrics Server plug-in is successfully installed, the content listed in the following table is created.

Name	Description
Workloads	In namespace **kube-system** of the cluster selected in **Step 6**, create a workload named **metrics-server**.
Services	In namespace **kube-system** of the cluster selected in **Step 6**, create a service named **metrics-server**.

- If the Metrics Server plug-in fails to be installed, try again by referring to **Reinstalling Metrics Server**.

----End

Upgrading Metrics Server

Upgrade the Metrics Server plug-in to the latest version.

Auto scaling is unavailable during the Metrics Server plug-in upgrade. If the traffic is abnormal during this period, the component service may be abnormal.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Metrics Server** under **Addons**.

- For a non-HA environment, click **Upgrade**.
- For an HA environment, click the AZ tab of the cluster where the Metrics Server plug-in is to be upgraded, and click **Upgrade**.

Step 6 In the displayed dialog box, click **OK**.

----End

Restarting the Metrics Server Load Process

Restart the Metrics Server load process.

Auto scaling is unavailable during the Metrics Server load startup. If the traffic is abnormal during this period, the component service may be abnormal.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Plug-in Management**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 5 Select **Metrics Server** under **Addons**.

- For a non-HA environment, click **Restart**.
- For an HA environment, click the AZ tab of the cluster where the Metrics Server plug-in is to be restarted, and click **Restart**.

Step 6 In the displayed dialog box, click **OK**.

----End

Reinstalling Metrics Server

Reinstall Metrics Server if it fails to be installed.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Metrics Server** under **Addons**.

- For a non-HA environment, click **Redeploy**.
- For an HA environment, click the AZ tab of the cluster where the Metrics Server plug-in fails to be installed, and click **Redeploy**.

----End

Deleting Metrics Server

If the Metrics Server plug-in is deleted, the content listed in the following table is deleted.

Name	Description
Workloads	metrics-server workload created in the kube-system namespace of the cluster selected when **Installing Metrics Server**. For details about workloads, see **Overview**.
Services	metrics-server service created in the kube-system namespace of the cluster selected when **Installing Metrics Server**. For details about services, see **Overview**.

After the Metrics Server plug-in is deleted, the **HPA policy** cannot be used for component instance auto scaling.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 Choose **Plug-in Management**.

Step 5 Select **Metrics Server** under **Addons**.

- For a non-HA environment, click **Delete**.
- For an HA environment, click the AZ tab of the cluster where the Metrics Server plug-in is to be deleted, and click **Delete**.

Step 6 Click **OK**.

----End

3.3.2.6 Unbinding a Cluster

If a cluster that has been bound to a Kubernetes or VM + Kubernetes environment is no longer used, you can unbind it from the environment.

After a CCE cluster is unbound, the unbound cluster resources cannot be used when **Creating and Deploying a Component** based on a container in this environment.

Prerequisites

- A cluster has been bound. For details, see **Binding a Cluster**.
- If the Sermant Injector plug-in has been installed in the cluster resources to be removed, delete the plug-in first. For details, see **Deleting Sermant Injector**.

Unbinding a Cluster

Step 1 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 2 Click the target environment. The **Overview** page is displayed.

Step 3 Choose **Clusters** from **Compute**.

Step 4 Click **Unbind** to remove the cluster from the environment.

Unbinding a CCE cluster resource whose **Resource Mode** is **Manage** will not delete the resource instance. To delete it, see **Deleting a Cluster**.

----End

3.3.3 Managing VM Agents

To deploy a component to a VM (ECS or custom server), you need to install the VM agent. After the host node is managed, the backend can communicate with the host node.

VM Agent Status Description

For details about the VM agent status and description, see **Table 3-17**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 3-17 VM agent status description

Agent Status	Description
Agent uninstalled	The VM agent is not installed on the ECS node. You need to install the VM agent.
Online	The VM agent has been installed on the ECS node and is running properly.
Agent offline	The VM agent has been installed on the ECS node, but is offline and cannot work properly. For details about how to handle agent offline, see **What Should I Do If the VM Agent Is Offline?**
Upgrading	The VM agent has been installed on the ECS node and is being upgraded.
Upgrade failed	The VM agent has been installed on the ECS node and fails to be upgraded.
Agent restarting	The VM agent has been installed on the ECS node and is restarting.
Agent unhealthy	The VM agent has been installed on the VM node and is unhealthy. The node cannot access the ServiceStage domain name. To rectify the fault, see **What Should I Do If the VM Agent Is Subhealthy?**

Supported OSs and Versions

The VM agent supports multiple OSs. You need to create an image by referring to **Table 3-18**, use the created image to create an ECS, and install the VM agent.

Table 3-18 OSs and versions supported by the VM agent

OS	Version	Description
EulerOS	• 2.2 64bit • 2.3 64bit • 2.5 64bit • 2.8 64bit	• For Linux x86_64 servers, all the listed OSs and versions are supported. • For Linux Arm servers, all the listed OSs and versions except CentOS 7.3 and earlier are supported.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

OS	Version	Description
CentOS	• 6.5 64bit • 6.8 64bit • 6.9 64bit • 6.10 64bit • 7.2 64bit • 7.3 64bit • 7.4 64bit • 7.5 64bit • 7.6 64bit • 7.7 64bit • 7.8 64bit • 7.9 64bit	
Fedora	• 29 64bit • 30 64bit	
openEuler	20.03 64bit	

Installing a VM Agent

This section describes how to install a VM agent for a single VM (ECS or custom server) managed in the VM or VM + Kubernetes environment. Before performing this operation, ensure that:

- VM resources have been managed in the environment, and no VM agent is installed. For details, see **Managing Basic Resources**.
- Python 2.x or later has been installed on the VM.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target VM environment. The **Overview** page is displayed.

Step 4 Choose **ECSs** or **Custom servers** from **Compute**.

Step 5 In the managed resource list, locate the VM where the agent is to be installed and click **Install Agent** in the **Agent Status** column.

Step 6 Select an **Authorization Model** to authorize the agent to use your authentication information for obtaining the deployment, upgrade, start, and stop tasks of an application and execute the task.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Authorization Model	Description
Agency	Only managed ECSs support this function. 1. Click and select an agency. For details about how to create an agency, see **Creating an Agency**. When creating an agency, you need to delegate the op_svc_ecs account to manage resources or ECS cloud service to access other cloud resources, and select the Tenant Administrator policy in the corresponding region. 2. Click .
AK/SK	For security purposes, obtain and use the AK/SK with the ServiceStage Development permission. The account to which the AK/SK belongs and the account for installing the VM agent must belong to the same user. For details about how to obtain the AK/SK, see **Access Keys**. Set **AK/AK/SK Filling**. • Manual In the VM Agent installation command, {AK} and {SK} are placeholders and do not contain the actual AK and SK values. After copying the installation command and logging in to the VM to install the agent, manually enter the AK and SK values as prompted, which is more confidential. • Quick entry AK and SK are directly filled in the VM agent installation command. After copying the installation command, log in to the VM and enter the installation command to install the agent. Enter the obtained AK and SK in the text boxes.

Step 7 Click **Copy Command**. The agent installation command is automatically generated in the lower part of the window.

- Example command for the **Agency** model:
```
export AGENT_INSTALL_URL=https://${Region_Name}-servicestage-vmapp.obs.${Region_Name}.${Domain_Name}/vmapp/agent/agent-install.sh;if [ -f `which curl` ];then curl -# -O -k ${AGENT_INSTALL_URL};else wget --no-check-certificate ${AGENT_INSTALL_URL};fi;bash agent-install.sh ${Project_ID} ${Version} ${Region_Name} ${Flag}
```
- In AK/SK authentication mode, run the following commands to install the agent on an ECS:
```
export AGENT_INSTALL_URL=https://${Region_Name}-servicestage-vmapp.obs.${Region_Name}.${Domain_Name}/vmapp/agent/agent-install.sh;if [ -f `which curl` ];then curl -# -O -k ${AGENT_INSTALL_URL};else wget --no-check-certificate ${AGENT_INSTALL_URL};fi;bash agent-install.sh ${AK}${SK} ${Project_ID} ${Version} ${Region_Name} ${Flag}
```
- In AK/SK authentication mode, run the following commands to install the agent on a custom server:
```
export AGENT_INSTALL_URL=https://${Region_Name}-servicestage-vmapp.obs.${Region_Name}.${Domain_Name}/vmapp/agent/agent-install.sh;if [ -f `which curl` ];then curl -# -O -k ${AGENT_INSTALL_URL};else wget --no-check-certificate ${AGENT_INSTALL_URL};fi;bash agent-
```

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

```
install.sh ${AK}${SK} ${Project_ID} ${Version} ${Region_Name} ${Flag} ${Domain_Name} ${Server_ID}
${Server_IP} ${Server_Name}
```

The parameters are described as follows:

Parameter	Description
AGENT_INSTALL_URL	Agent installation address.
{AK}, {SK}	Access key. • If **Manual** is selected in **Step 6**, {AK} and {SK} are access key placeholders and do not contain the actual AK and SK values. • If **Quick entry** is selected in **Step 6**, {AK} and {SK} are the actual access key values.
${Region_Name}	Region name. For example, ap-southeast-1.
${Domain_Name}	Global domain name.
${Project_ID}	Project ID. For details about how to obtain a project ID, see **API Credentials**.
${Version}	Version number. The default value is **latest**, indicating that the latest version is automatically downloaded.
${Flag}	Whether to automatically add the application access port. • **true**: yes • **false**: no
${Server_ID}	Unique ID of the custom server.
${Server_IP}	IP address of the custom server.
${Server_Name}	Name of the custom server.

Step 8 Log in to the VM and run the installation command.

Select **Manual** in **Step 6**, enter the installation command, press **Enter**, and enter the obtained AK and SK as prompted.

If the message "Install agent success!" is displayed, the installation was successful.

- After the VM agent is installed, the following directories are generated on the VM where the agent is installed:
  - **/opt/application**: directory for storing services and configuration files of VM-deployed components. Do not add, delete, or modify the directory. Otherwise, the installed VM agent may be offline.
  - **/opt/servicestage-agent**: directory for storing configuration files of the VM agent. Do not add, delete, or modify the directory. Otherwise, the installed VM agent may be offline.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

  - **/var/log/servicestage-agent**: log directory of the installed VM agent, which stores the agent log **agent.log** and monitoring script log **servicestage-agent-watchdog.log**. Do not delete or modify the directory. Otherwise, the installed VM agent may be offline.
  - **/var/log/application**: directory for storing component logs. Do not delete or modify the directory. Otherwise, the installed VM agent may be offline.
  - **/opt/ssa/packages**: temporary directory for storing service packages of VM-deployed components. Do not delete or modify the directory. Otherwise, the installed VM agent may be offline.
- If the VM agent fails to be installed, see **What Should I Do If I Don't See the VM Agent After Installing It?**

----End

Upgrading a VM Agent

If the installed VM agent is in the **Agent online** state and a later version is available, you can upgrade the agent by referring to this section.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target VM environment. The **Overview** page is displayed.

Step 4 Choose **ECSs** or **Custom servers** from **Compute**.

Step 5 In the managed resource list, select the target resource and click **Upgrade Agent**.

Step 6 Click **OK**.

After **Agent Status** changes from **Agent upgrading** to **Agent online**, the VM agent has been upgraded.

----End

Restarting a VM Agent

If the installed VM agent is in the **Agent online** state, you can restart the agent based on service requirements.

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target VM environment. The **Overview** page is displayed.

Step 4 Choose **ECSs** or **Custom servers** from **Compute**.

Step 5 In the managed resource list, select the target resource and click **Restart Agent**.

Step 6 Click **OK**.

After **Agent Status** changes from **Agent restarting** to **Agent online**, the VM agent has been restarted.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

3.3.4 Removing a Basic Resource

You can remove basic resources managed in the environment that are no longer used.

After the basic resources managed in an environment are removed, they cannot be used when this environment is used to deploy components.

Removing a resource will not delete the resource instance. Log in to the resource console and delete the resource.

Prerequisites

The resources to be removed have been managed in the environment. For details, see **Managing Basic Resources**.

Removing a Basic Resource

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose the resource from **Compute**, **Networking**, or **Middleware**.

Step 5 In the managed resource list, perform the following operations:

- To remove resources in batches, select the target resources and click **Remove Resource**.
- To remove a single resource, click **Remove** in the **Operation** column of the target resource.

The ServiceComb engines managed in the HA environment of the Kubernetes type can only be removed one by one.

For details about how to remove cluster resources bound in the Kubernetes or VM + Kubernetes environments, see **Unbinding a Cluster**.

----End

3.4 Managing Environment Variables of an Environment

Environment variables of an environment describe environment information and environment resource information, such as middleware credentials, connection information, and environment functions. Customers can use these variables to shield environment or resource differences, and switch components to different environments or run them using different cloud service resources without modification.

If the resources (DCS, RDS, ServiceComb, Nacos, and CSS) listed in **Table 3-19** exist in the environment, ServiceStage writes the corresponding built-in and custom environment variables in the environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 3-19 Built-in and custom environment variables of an environment

Resource Type	Name	Description	Variable Source	Variable/Variable Reference Example
DCS	DISTRIBUTED_CACHE_TYPE	Distributed cache type.	Built-in	Redis
	DISTRIBUTED_CACHE_NAME	Name of a DCS instance.	Built-in	dcs-test
	DISTRIBUTED_CACHE_VERSION	Distributed cache version.	Built-in	5.0
	DISTRIBUTED_CACHE_CLUSTER	Distributed cache cluster type.	Built-in	false
	DISTRIBUTED_CACHE_HOST	Distributed cache address.	Built-in	192.168.0.136
	DISTRIBUTED_CACHE_PORT	Distributed cache port.	Built-in	6379
	DISTRIBUTED_SESSION_CLUSTER	Distributed session cluster type.	Built-in	false
	DISTRIBUTED_SESSION_HOST	Distributed session address.	Built-in	192.168.0.136
	DISTRIBUTED_SESSION_VERSION	Distributed session version.	Built-in	5.0
	DISTRIBUTED_SESSION_TYPE	Distributed session type.	Built-in	Redis
	DISTRIBUTED_CACHE_PASSWORD	Distributed cache password.	Custom	********
RDS	RELATIONAL_DATABASE_DB_TYPE	Cloud database type.	Built-in	MySQL

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Name	Description	Variable Source	Variable/Variable Reference Example
	RELATIONAL_DATABASE_HOST	Cloud database address.	Built-in	172.16.0.185
	RELATIONAL_DATABASE_CONNECTION_TYPE	Cloud database connection type.	Built-in	JNDI
	RELATIONAL_DATABASE_PORT	Cloud database port.	Built-in	3306
	RELATIONAL_DATABASE_NAME	Cloud database name.	Built-in	rds-pro
	RELATIONAL_DATABASE_VERSION	Cloud database version.	Built-in	8.0
	RELATIONAL_DATABASE_JNDI_NAME	Cloud database JNDI name.	Custom	jndi-u1
	RELATIONAL_DATABASE_DB_USER	Cloud database account.	Custom	rds-u1
	RELATIONAL_DATABASE_PASSWORD	Cloud database password.	Custom	********
ServiceComb	PAAS_CSE_CC_ENDPOINT	Address of the ServiceComb engine configuration center.	Built-in	https://192.168.0.31:30110,https://192.168.0.36:30110
	PAAS_CSE_SC_ENDPOINT	Service center address of the ServiceComb engine.	Built-in	https://192.168.0.31:30100,https://192.168.0.36:30100

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Name	Description	Variable Source	Variable/Variable Reference Example
	SERVICECOMB_SERVICE_PARAMETERS	ServiceComb engine service configuration.	Built-in	cas-instance-id:c3bee2c1-5174-4ed4-b4e0-2cfdd5b47357
	PAAS_CSE_ENGINE_MGR_ENDPOINT	Address of the ServiceComb engine management node.	Built-in	Endpoint of the ServiceComb engine. Obtain the value from **Regions and Endpoints**.
	SERVICECOMB_SERVICE_PROPS	ServiceComb engine service configuration.	Built-in	cas-instance-id:c3bee2c1-5174-4ed4-b4e0-2cfdd5b47357
	SERVICECOMB_ENGINE_NAME	ServiceComb engine name.	Built-in	cse-izq6rg
	SERVICECOMB_INSTANCE_PROPS	ServiceComb engine instance configuration.	Built-in	affinity-tag:base,cas-instance-id:c3bee2c1-5174-4ed4-b4e0-2cfdd5b47357,cas_lane_tag:base
	PAAS_CSE_DS_ENDPOINT	Dashboard center address of the ServiceComb engine.	Built-in	https://192.168.0.31:30109,https://192.168.0.36:30109
	DYNAMIC_CONFIG_DYNAMICCONFIGTYPE	Type of the ServiceComb engine configuration center.	Built-in	KIE
	DYNAMIC_CONFIG_SERVERADDRESS	Configuration center address of the ServiceComb engine.	Built-in	https://192.168.0.31:30110,https://192.168.0.36:30110

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Name	Description	Variable Source	Variable/Variable Reference Example
Nacos	REGISTER_SERVICE_REGISTERTYPE	Nacos engine service registry and discovery type.	Built-in	NACOS
	REGISTER_SERVICE_ADDRESS	Service center address of the Nacos engine.	Built-in	https://192.168.0.31:30100,https://192.168.0.36:30100
	SPRING_CLOUD_NACOS_CONFIG_SERVERADDR	Service center address of the Nacos engine.	Built-in	https://192.168.0.31:30100,https://192.168.0.36:30100
	SPRING_CLOUD_NACOS_DISCOVERY_SERVERADDR	Service center address of the Nacos engine.	Built-in	https://192.168.0.31:30100,https://192.168.0.36:30100
	EUREKA_CLIENT_SERVICEURL_DEFAULTZONE	Eureka server registry address.	Built-in	https://192.168.1.30:8761/eureka/
	DYNAMIC_CONFIG_DYNAMICCONFIGTYPE	Type of the Nacos engine configuration center.	Built-in	NACOS
	DYNAMIC_CONFIG_SERVERADDRESS	Configuration center address of the Nacos engine.	Built-in	https://192.168.0.31:30100,https://192.168.0.36:30100
GaussDB	PAAS_GAUSSDB_VERSION	Version.	Built-in	8.200.0
	PAAS_GAUSSDB_NAME	Name.	Built-in	gauss-ss
	PAAS_GAUSSDB_ADDRESS	Connection address.	Built-in	192.168.0.178:8000,192.168.0.47:8000,192.168.0.166:8000

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Name	Description	Variable Source	Variable/Variable Reference Example
	PAAS_GAUSSDB_DB_NAME	Database name.	Custom	cas
	PAAS_GAUSSDB_DB_USER	Username.	Custom	root
	PAAS_GAUSSDB_PASSWORD	Password.	Custom	******
RocketMQ	DISTRIBUTED_MESSAGE_ROCKETMQ_VERSION	Version.	Built-in	5.x
	DISTRIBUTED_MESSAGE_ROCKETMQ_NAME	Name.	Built-in	rocketmq-ss
	DISTRIBUTED_MESSAGE_ROCKETMQ_ADDRESS	Connection address.	Built-in	192.168.0.84:9093,192.168.0.212:9093,192.168.0.174:9093
	DISTRIBUTED_MESSAGE_ROCKETMQ_USER	Username.	Custom	root
	DISTRIBUTED_MESSAGE_ROCKETMQ_PASSWORD	Password.	Custom	******
CSS	PAAS_ELASTIC_SEARCH_VERSION	Version.	Built-in	7.10.2
	PAAS_ELASTIC_SEARCH_NAME	Name.	Built-in	css_es
	PAAS_ELASTIC_SEARCH_ADDRESS	Connection address.	Built-in	21.108.0.218:9200,21.108.0.88:9200,21.108.0.231:9200

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Resource Type	Name	Description	Variable Source	Variable/Variable Reference Example
	PAAS_ELASTIC_SEARCH_HTTPS_ENABLE	Whether to enable HTTPS.	Built-in	true
	PAAS_ELASTIC_SEARCH_USER	Username.	Custom	admin
	PAAS_ELASTIC_SEARCH_PASSWORD	Password.	Custom	******

You can also add, edit, and delete custom environment variables of an environment.

Priority of environment variables of the component, application, and environment levels: component > application > environment.

- For details about environment variables of an application, see **Managing Environment Variables of an Application**.
- For details about environment variables of a component, see **Adding an Environment Variable of a Component** (container-deployed components) or **Adding an Environment Variable of a Component** (VM-deployed components).

Adding an Environment Variable

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Choose **Environment Variables** > **Add Environmental Variable** and set variable parameters by referring to the following table.

Exercise caution when inputting sensitive information in configuring environment variables, or encrypt sensitive information to avoid information leakage.

Parameter	Description
Name	Name of an environment variable. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.).

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Variable/ Variable Reference	Value of a variable name.

Step 5 Click **Confirm** in the **Operation** column of the new environment variable.

Type of the new environment variable is **Custom**, and **Associated Object** is the current environment. After environment variables of an environment are changed, you can:

- Make the changed environment variables take effect for a specified component deployed in the environment by **Upgrading a Component in Single-batch Release Mode**.
- Make the changed environment variables take effect for multiple or all specified components deployed in the environment by **Upgrading Components in Batches**.

----End

Editing an Environment Variable

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Click **Environment Variables**.

Step 5 Click **Edit** in the **Operation** column of the custom environment variable to be edited and set parameters by referring to the following table.

Exercise caution when inputting sensitive information in configuring environment variables, or encrypt sensitive information to avoid information leakage.

Parameter	Description
Name	Name of an environment variable. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.).
Variable/ Variable Reference	Value of a variable name.

Step 6 Click **Confirm** in the **Operation** column of the environment variable.

After environment variables of an environment are changed, you can:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Make the changed environment variables take effect for a specified component deployed in the environment by **Upgrading a Component in Single-batch Release Mode**.
- Make the changed environment variables take effect for multiple or all specified components deployed in the environment by **Upgrading Components in Batches**.

----End

Deleting an Environment Variable

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**.

Step 3 Click the target environment. The **Overview** page is displayed.

Step 4 Click **Environment Variables** to delete a custom environment variable.

- Single deletion: Click **Delete** in the **Operation** column of the target custom environment variable.
- Batch deletion: Select the target custom environment variables and click **Bulk Delete**.

Step 5 Click **OK**.

Delete environment variables as required to prevent component startup failures caused by missing environment variables when **Upgrading a Single Component** or **Upgrading Components in Batches**.

After environment variables of an environment are changed, you can:

- Make the changed environment variables take effect for a specified component deployed in the environment by **Upgrading a Component in Single-batch Release Mode**.
- Make the changed environment variables take effect for multiple or all specified components deployed in the environment by **Upgrading Components in Batches**.

----End

3.5 Editing an Environment

Modify the name, enterprise project, and description of an environment.

Prerequisites

You have created an environment. For details, see **Creating an Environment**.

Editing an Environment

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Use either of the following methods to go to the **Edit Environment** page:

- Directly edit: Select the target environment and click **Edit** in the **Operation** column.
- Edit after viewing environment details: Click the target environment. On the displayed **Overview** page, click **Edit**.

Step 4 Edit the environment information by referring to the following table.

Parameter	Description
Environment	Environment name, which must be unique in the same enterprise project. Enter 2 to 64 case-sensitive characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after you **enable the enterprise project function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**.
Environment Tag	Tags are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use tags to classify them based on usage, owner, or environment. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment editing may fail. Contact your administrator to learn more about tag policies. Each environment can have a maximum of 20 tags. • Delete a tag Click ⊗ next to the tag to be deleted. • Add a tag 1. Click **Add Tag**. The **Add Tag** dialog box is displayed. 2. Click **Add Tag**. 3. Enter the key and value of the tag. The key contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-). The value contains 1 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 4. Click **OK**.
Description	Environment description. Enter up to 128 characters.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 5 Click **Save**.

----End

3.6 View Environment Details

View environment details at any time.

Prerequisites

You have created an environment. For details, see **Creating an Environment**.

Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Select an environment to view the following information:

Environment Information	Description
Environment	Environment name.
Enterprise Project	Enterprise project to which the environment belongs. Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**.
Environment Type	Environment type. • **VM**: applicable to VM-based deployment. Components are deployed on ECSs or custom servers using software packages. • **Kubernetes**: applicable to container-based deployment (CCE). Components are deployed using container images and scheduled by Kubernetes. • **VM + Kubernetes**: applicable to the environments that can manage clusters, ECSs, custom servers, and AS groups at the same time. You can create and deploy components based on a VM and container at the same time.
Created	Time when the environment is created. The format is yyyy-mm-dd hh:mm:ss GMT+8:00. Example: **2024-12-30 21:23:12 GMT +08:00**.
Creator	Account of the environment creator.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Environment Information	Description
Environment Status	Status of managed and deployed resources in the environment. To manually update the environment status, click the environment. On the **Overview** page, click **update** next to **Environment Status**. • **Unknown**: The resource status is not updated or fails to be updated after the environment is created. • **Refreshing**: The resource status in the environment is being refreshed. • **Partially Normal**: Some resources in the environment are abnormal. For example, the VM agent is not installed on the environment VM, the VM agent is offline, the ACL in the environment is not bound to a subnet, or the EIP in the environment is not bound to other resources in the environment. • **Normal**: The resources in the environment are normal.

----End

3.7 Deleting an Environment

You can delete an environment that is no longer used.

Prerequisites

All components deployed in the environment to be deleted have been deleted. For details, see **Deleting a Component**.

Deleting an Environment

Step 1 Log in to ServiceStage.

Step 2 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 3 Use either of the following methods to delete an environment:

- Directly delete: Select the target environment and click **Delete** in the **Operation** column.
- Delete after viewing environment details: Click the target environment. On the displayed **Overview** page, click **Delete**.

Step 4 Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Deleting an environment will not delete the basic resource instances managed in the environment. To delete them, log in to the resource console and delete the resource.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

4 Application Management

4.1 Application Overview

An application is a service system with functions and consists of one or more components.

For details about components, see **Component Overview**.

For example, the weather forecast is an application that contains the weather and forecast components. ServiceStage organizes multiple components by application, and supports dark launch and quick cloning of applications in different environments.

Figure 4-1 Application management process

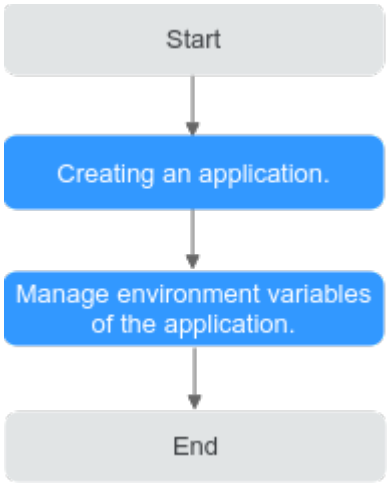


Figure 4-1 shows the application management process. A component must belong to an application. Before using ServiceStage to create and deploy a component, create an application. If you need to use environment variables instead of code to specify parameter settings during component deployment, you can add environment variables of an application.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

4.2 Creating an Application

Create an application before **Creating and Deploying a Component**.

ServiceStage allows a single user to create a maximum of 7500 applications under the same project.

Creating an Application

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management** > **Create Application** and configure the application. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Name	Unique application name. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
*Enterprise Project	Select an enterprise project. Enterprise projects let you manage cloud resources and users by project. It is available after you **enable the enterprise project function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Tags	Tags are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use tags to classify them based on usage, owner, or environment. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, application creation may fail. Contact your administrator to learn more about tag policies. Each application can have a maximum of 20 tags. 1. Enter the key and value of the tag. – The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). – The value contains 0 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 2. Click **OK**.
Description	Application description. Enter up to 128 characters.

Step 3 Click **OK**.

The created application is displayed in the application list.

----End

4.3 Managing Environment Variables of an Application

Environment variables of an application are parameters set in the system or user applications. You can obtain the values of environment variables by calling APIs. During deployment, parameters are specified through environment variables instead of in the code, which makes the deployment flexible. Environment variables added to an application are global environment variables and take effect for all application components.

To add environment variables for a specific component:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For container-deployed components, see **Adding an Environment Variable of a Component**.
- For VM-deployed components, see **Adding an Environment Variable of a Component**.

For details about environment variables of an environment, see **Managing Environment Variables of an Environment**.

Before using environment variables of an application, add environment variables for the application. You can add environment variables of an application by **Manually Adding an Environment Variable** or **Importing Environment Variables**, or **Editing an Environment Variable** and **Deleting an Environment Variable** based on service requirements.

Manually Adding an Environment Variable

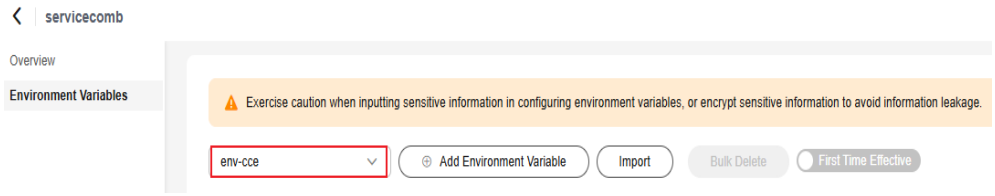
Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Click the target application. The **Overview** page is displayed.

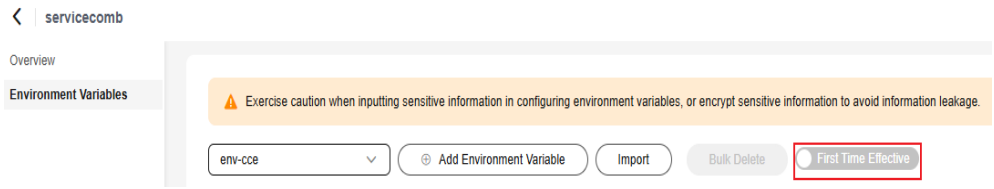
Step 4 In the left navigation pane, click **Environment Variables**.

Step 5 Select a created environment from the drop-down list.

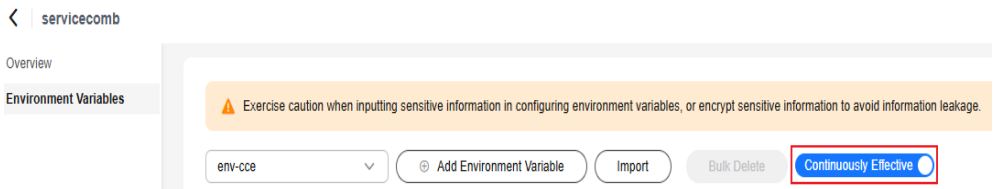


Step 6 Set the policy for the environment variables to take effect.

- **First Time Effective**: The environment variables take effect only when the component is created for the first time. Subsequent modifications to the environment variables will not be synchronized to the existing components of the application.



- **Continuously Effective**: The modification of environment variables will be synchronized to the existing components of the application through component upgrade.



Step 7 Click **Add Environment Variable** and set **Name** and **Variable/Variable Reference**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Exercise caution when inputting sensitive information in configuring environment variables, or encrypt sensitive information to avoid information leakage.

- **Name**: application environment variable name, which must be unique in the same application environment. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.).
- **Variable/Variable Reference**: application environment variable value.

For example, if you set **Name** to **User** and **Variable/Variable Reference** to **admin**, when the program code reads the **User** environment variable, **admin** is obtained. Or, you can start a subprocess or read a file as user **admin**. The actual execution effect depends on the code.

Step 8 Click **Submit**.

After an environment variable with **Continuously Effective** selected changes, you can:

- Make the changed environment variables take effect for a specified component of the application by **Upgrading a Single Component**.
- Make the changed environment variables take effect for multiple or all components of the application by **Upgrading Components in Batches**.

----End

Importing Environment Variables

The file must be a key-value pair mapping file in JSON or YAML format. A maximum of 200 environment variables can be imported at a time. Example file content and format:

```
{"key1": "value1", "key2": "value2"}
```

- **key1** and **key2** are the environment variable names, which must be unique in the same application environment. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.).
- **value1** and **value2** are the environment variable values.

Exercise caution when inputting sensitive information in configuring environment variables, or encrypt sensitive information to avoid information leakage.

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Click the target application. The **Overview** page is displayed.

Step 4 In the left navigation pane, click **Environment Variables**.

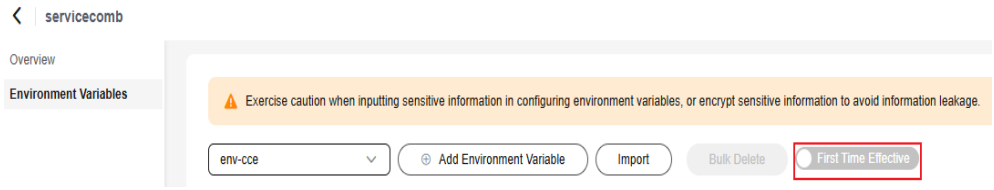
Step 5 Select a created environment from the **Environment** drop-down list.

Step 6 Set the policy for the environment variables to take effect.

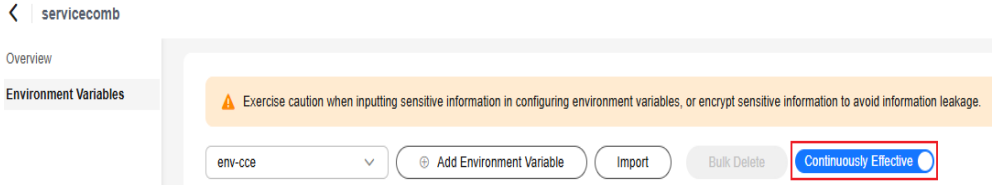
- **First Time Effective**: The environment variables take effect only when the component is created for the first time. Subsequent modifications to the

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

environment variables will not be synchronized to the existing components of the application.



- **Continuously Effective**: The modification of environment variables will be synchronized to the existing components of the application through component upgrade.



Step 7 Click **Import** and select the created environment variable file.

Step 8 Click **Submit**.

After an environment variable with **Continuously Effective** selected changes, you can:

- Make the changed environment variables take effect for a specified component of the application by **Upgrading a Single Component**.
- Make the changed environment variables take effect for multiple or all components of the application by **Upgrading Components in Batches**.

----End

Editing an Environment Variable

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

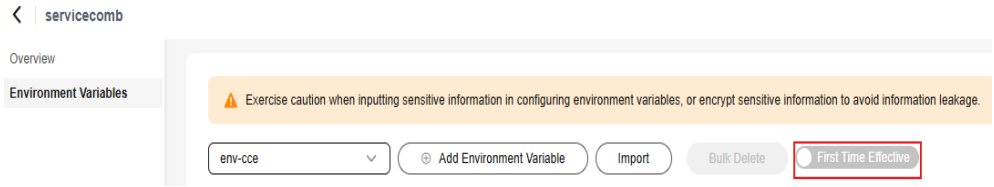
Step 3 Click the target application. The **Overview** page is displayed.

Step 4 In the left navigation pane, click **Environment Variables**.

Step 5 Select a created environment from the **Environment** drop-down list.

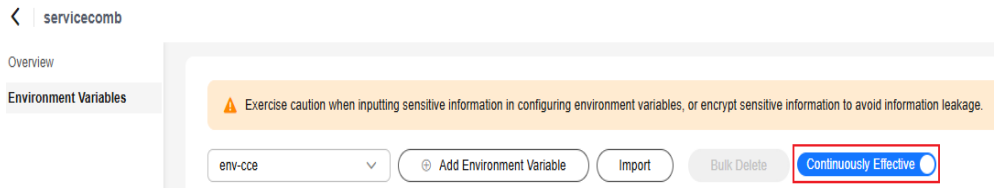
Step 6 Set the policy for the environment variables to take effect.

- **First Time Effective**: The environment variables take effect only when the component is created for the first time. Subsequent modifications to the environment variables will not be synchronized to the existing components of the application.



 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Continuously Effective**: The modification of environment variables will be synchronized to the existing components of the application through component upgrade.



Step 7 Select the target variable and click **Edit** in the **Operation** column.

Step 8 Reset **Name** and **Variable/Variable Reference**.

Exercise caution when inputting sensitive information in configuring environment variables, or encrypt sensitive information to avoid information leakage.

- **Name**: environment variable name, which must be unique in the same environment. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.).
- **Variable/Variable Reference**: environment variable value.

Step 9 Click **Submit**.

After an environment variable with **Continuously Effective** selected changes, you can:

- Make the changed environment variables take effect for a specified component of the application by **Upgrading a Single Component**.
- Make the changed environment variables take effect for multiple or all components of the application by **Upgrading Components in Batches**.

----End

Deleting an Environment Variable

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

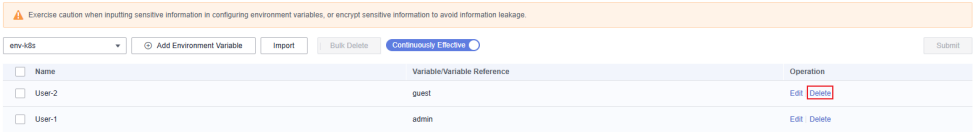
Step 3 Click the target application. The **Overview** page is displayed.

Step 4 In the left navigation pane, click **Environment Variables**.

Step 5 Select a created environment from the **Environment** drop-down list.

- Single deletion: Select the target variable and click **Delete** in the **Operation** column.

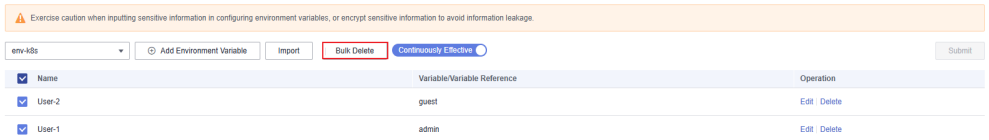
Figure 4-2 Single deletion



- Batch deletion: Select the target variables and click **Bulk Delete**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 4-3 Batch deletion



Step 6 In the displayed dialog box, click **OK**.

After an environment variable with **Continuously Effective** selected changes, you can:

- Make the changed environment variables take effect for a specified component of the application by **Upgrading a Single Component**.
- Make the changed environment variables take effect for multiple or all components of the application by **Upgrading Components in Batches**.

----End

4.4 Editing an Application

You can modify the application parameters as required.

Prerequisites

You have created an application. For details, see **Creating an Application**.

Editing an Application

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Use either of the following methods to go to the application editing page:

- Directly edit: Select the target application and click **Edit** in the **Operation** column.
- Edit after viewing application details: On the **Application Management** page, click the target application to go to the **Component Management** page. After **Viewing Application Overview**, click **Edit** in the upper right corner.

Step 4 Configure the application again by referring to the following table.

Parameter	Description
Name	Unique application name. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Enterprise Project	Select an enterprise project. Enterprise projects let you manage cloud resources and users by project. It is available after you **enable the enterprise project function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**.
*Tags	Tags are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use tags to classify them based on usage, owner, or environment. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, application editing may fail. Contact your administrator to learn more about tag policies. Each application can have a maximum of 20 tags. • Add a tag: Enter the key and value of the tag, and click **OK**. – The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). – The value contains 0 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. • Delete a tag: Click ⊗ next to the tag to be deleted.
Description	Application description. Enter up to 128 characters.

Step 5 Click **OK**.

----End

4.5 Viewing Application Overview

After the application is created, you can go to the **Overview** page to view the application details.

Prerequisites

You have created an application. For details, see **Creating an Application**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Viewing Application Overview

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Click the target application. On the displayed **Overview** page, view the following application details.

Parameter	Description
ID	Application ID.
Enterprise Project	Enterprise project to which the application belongs.
Created	Time when the application is created.
Creator	Account for creating the application.
Description	Application description entered during application creation.
Component List	If a component has been created and deployed under the application, you can view the component details in the component list. For details, see **Viewing Component Details**.

----End

4.6 Deleting an Application

You can delete an application that is no longer used.

Prerequisites

No component is contained in the application or all components in the application have been deleted. For details, see **Deleting a Component**.

Deleting an Application

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Use either of the following methods to delete an application:

- Directly delete: Select the target application and click **Delete** in the **Operation** column.
- Delete after viewing application details: Click the target application to go to the **Component Management** page. After **Viewing Application Overview**, click **Delete** in the upper right corner.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 Click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5 Component Management

5.1 Component Overview

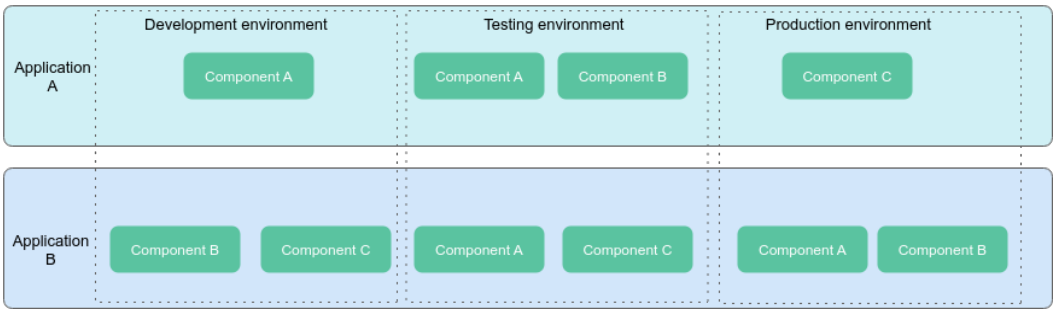
Introduction

A component is a service feature implementation of an application. It is carried by code or software packages and can be independently deployed and run in an environment.

For details about applications and environments, see **Application Overview** and **Environment Overview**.

Figure 5-1 shows the relationships between components, applications, and environments.

Figure 5-1 Relationships



After creating an application and environment on ServiceStage, you can set the component technology stack and component source (package type and storage mode) based on service requirements to create and deploy components on the UI.

ServiceStage provides the capabilities of automatically creating and upgrading components using component templates. When creating a component using a component template, you can create configurations and Kubernetes resources. For details, see **Creating a Component Using a Component Template**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Technology Stack

A technology stack includes the operating system, framework, and runtime system for component running. It consists of attributes such as the stack name, type, status, and version. The version complies with the **semantic versioning specifications**.

ServiceStage provides and manages the stack lifecycle. You only need to focus on service development to improve application hosting experience.

The lifecycle phases of the technology stack are defined as follows:

- Preview: The beta version is released.
- General Availability (GA): The official version is released.
- End of Life (EOL): The lifecycle ends.

The technology stack status is defined as follows:

- Preview: The stack is in the Preview phase.
- Supported: The stack is in the GA phase.
- Deprecated: The stack is in the GA phase but the EOL announcement has been released, or the stack is not recommended by ServiceStage.

Table 5-1 lists the open-source technology stacks provided by ServiceStage. If you encounter any problem when using the open-source technology stack, visit the official forum and community of the technology stack and contact technical support.

If these open-source technology stacks cannot meet your service requirements, you can use a custom technology stack. For details, see **Adding a Custom Technology Stack Version**.

ServiceStage supports the deployment of Java application components that are packaged as a WAR package and developed based on the Spring Boot framework. Version mapping between JDK and Spring Boot:

- OpenJDK 8 supports Spring Boot 2.0.x to 2.7.x.
- OpenJDK 11 supports Spring Boot 2.2.x to 2.7.x.
- OpenJDK 17 supports Spring Boot 2.5.x and later.
- OpenJDK 21 supports Spring Boot 2.7.x and later.

For details, see **Spring Boot Versions**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-1 Open-source technology stack information

Technology Stack	Type	Status	Release Description	Package Type and Deployment Mode
OpenJDK8	Java	Supported	• **OpenJDK-8u442** • Image OS: EulerOS 2.9.21. Character set en_US.UTF-8 is supported.	• The package type is source code or software package (JAR or WAR), and container-based deployment is supported. For details, see **Component Deployment Modes**. • The package type is software package (JAR, WAR, or package in .zip or .tar.gz format), and VM-based deployment is supported. For details, see **Component Deployment Modes**. For details about how to compress a package, see **How Do I Package a Java or Tomcat Application?**
OpenJDK11	Java	Supported	• **OpenJDK 11.0.26** • Image OS: EulerOS 2.9.21. Character set en_US.UTF-8 is supported.	
OpenJDK17	Java	Supported	• **OpenJDK 17.0.14** • Image OS: EulerOS 2.9.21. Character set en_US.UTF-8 is supported.	
OpenJDK21	Java	Supported	• **OpenJDK 21.0.6** • Image OS: EulerOS 2.9.21. Character set en_US.UTF-8 is supported.	

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Technology Stack	Type	Status	Release Description	Package Type and Deployment Mode
Tomcat8/ OpenJDK8	Tomcat	Supported	• **OpenJDK-8u442** • **Tomcat-8.5.88** • Image OS: EulerOS 2.9.21	• The package type is source code or WAR package, and container-based deployment is supported. For details, see **Component Deployment Modes**. • The package type is WAR package or package in .zip or .tar.gz format, and VM-based deployment is supported. For details, see **Component Deployment Modes**. For details about how to compress a package, see **How Do I Package a Java or Tomcat Application?**
Tomcat9/ OpenJDK8	Tomcat	Supported	• **OpenJDK-8u442** • **Tomcat-9.0.88** • Image OS: EulerOS 2.9.21	

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Technology Stack	Type	Status	Release Description	Package Type and Deployment Mode
Node.js8	Node.js	Supported	• **Nodejs-v8.17.0** • Image OS: EulerOS 2.9.21	• The package type is source code or ZIP package, and container-based deployment is supported. For details, see **Component Deployment Modes**. • The package type is ZIP package and VM-based deployment is supported. For details, see **Component Deployment Modes**.
Node.js14	Node.js	Supported	• **Nodejs-v14.21.3** • Image OS: EulerOS 2.9.21	
Node.js16	Node.js	Supported	• **Nodejs-v16.20.2** • Image OS: EulerOS 2.9.21	
Node.js18	Node.js	Supported	If a component is created and deployed based on a VM using Node.js 18, the GLIBC version of the host OS in the component running environment must be 2.25 or later. Otherwise, the component cannot be started. • **Nodejs-v18.20.4** • Image OS: EulerOS 2.9.21	
Docker	Docker	-	Supported by CCE. For details, see **Kubernetes Release Notes**.	The package type is image package and container-based deployment is supported. For details, see **Component Deployment Modes**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Technology Stack	Type	Status	Release Description	Package Type and Deployment Mode
Python3	Python	-	• **Python-3.12.4** • Image OS: EulerOS 2.9.21	The package type is source code or ZIP package, and container-based deployment is supported. For details, see **Component Deployment Modes**.
Php7	Php	-	• **Php-7.4.33** • Image OS: EulerOS 2.9.21	The package type is source code or ZIP package, and container-based deployment is supported. For details, see **Component Deployment Modes**.

Component Source

As shown in **Table 5-1**, when using ServiceStage to create and deploy a component, you need to select the component source based on the selected deployment mode. You can select source code repository, software package (JAR package, WAR package, ZIP package, or package), or image package. You can also select a package storage mode based on service requirements. For details, see **Table 5-2**.

Table 5-2 Component sources

Package Type	Storage Mode
Source Code Repository	Create authorization by referring to **Creating Repository Authorization** and set the code source.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Package Type	Storage Mode
JAR package	The following upload methods are supported: • Select the corresponding software package from the SWR software repository. Upload the software package to the software repository in advance. For details, see **Uploading a Software Package**. If PKIToken is disabled, the SWR repository is not supported. • Select the corresponding software package from OBS. Upload the software package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**. • Obtain the corresponding software package from the custom file address. You can download the software package from the HTTP/HTTPS custom file address. Upload the software package to the corresponding custom file address in advance.
WAR package	
ZIP package	
Package	
Image package	Containerized applications need to be created based on images. **My Images** (private images), **Open Source Images**, **Shared Images**, and **Third-Party Images** are supported. • If you select **My Images**, upload the image to the image repository in advance. For details, see **Uploading an Image**. • If you select **Third-Party Images**, ensure that you have obtained the address of the third-party image. The format of the image address is as follows: *{IP address of the third-party image repository}:{Port number for accessing the third-party image repository}/{Image storage path}/{Image name}:{Image tag}* Alternatively: *{Image name}:{Image tag}* If the image tag is not specified, the latest version is used by default.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Component Deployment Modes

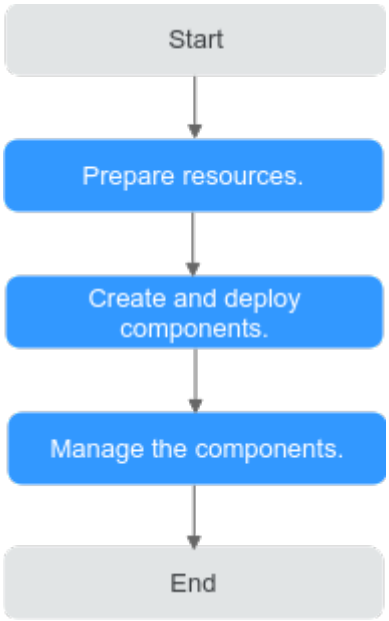
Table 5-3 describes the component deployment modes supported by ServiceStage.

Table 5-3 Deployment modes

Deployment Mode	Description
Container-based deployment	Uses container technologies (such as Docker and Kubernetes) to package applications and their dependencies, ensuring consistent running in any environment. This method effectively simplifies the deployment process and improves the consistency between development, test, and production environments. With CCE and custom Kubernetes clusters, you can easily deploy, manage, and scale containerized applications. ServiceStage allows you to create and deploy components based on a container in a Kubernetes or VM + Kubernetes environment. For details, see **Environment Type Description**.
VM	A VM, or an ECS or custom server, is a basic computing unit that consists of vCPUs, memory, OS, and EVS disks. After creating an ECS, you can use it like using your local computer or physical server to deploy components. ServiceStage allows you to create and deploy components based on a VM in a VM or VM + Kubernetes environment. For details, see **Environment Type Description**.

Component Management Process

Figure 5-2 Component management process



 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

As shown in **Figure 5-2**, component management operations are as follows:

1. Before using component management, you need to prepare the following resources:
   a. Upload the source code or software package of the component to be deployed to the repository based on **Component Source**.
   b. Create the environment to which the component to be deployed belongs based on **Component Deployment Modes**. For details, see **Environment Management**.
   c. Create the application to which the component to be deployed belongs. For details, see **Application Management**.
2. **Creating and Deploying a Component** based on the **Component Deployment Modes**.
3. Manage the created components:
   - To upgrade a single component, see **Upgrading a Single Component**.
   - To upgrade components in batches, see **Upgrading Components in Batches**.
   - To redeploy a component using the historical version configuration as the template, see **Redeploying a Component**.
   - To roll back a component, see **Rolling Back a Component**.
   - To manage component running metrics and logs, see **Component O&M**.
   - To manage component instance scaling, see **Configuring a Scaling Policy of a Component Instance**.

5.2 Creating and Deploying a Component

5.2.1 Creating a Component Based on a Container Using Manual Configurations

ServiceStage supports a maximum of 15,000 components in the same project.

This section describes how to create and deploy a component based on a container using manual configurations. With the guidance of the ServiceStage console, you can quickly create a component, save a component draft, and deploy a component in a container environment.

When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. During component creation, you can click **Save as Draft** to prevent component parameter settings from being lost.

Prerequisites

- An application has been created because components can only be added to applications. For details, see **Creating an Application**.
- An environment supported by container-based deployment has been created and resources have been managed because components need to be deployed in an environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

  - For the environment types supported by container-based deployment, see **Component Deployment Modes**.
  - For details about how to create an environment and manage resources, see **Environment Management**.
- An organization has been created because images generated during component build need to be managed by organization. For details, see **Creating an Organization**.
- A namespace has been bound to the CCE cluster to isolate build data and component instances. For details, see **Managing Namespaces**.
- If you create a component based on a source code repository, repository authorization has been created. For details, see **Creating Repository Authorization**.
- If you create a component based on a software package:
  - Upload the software package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Upload the software package to the OBS bucket. For details, see **Streaming Upload (PUT)**.
  - If the package type is software package and you need to use JFrog (example) as the software package repository, you can download the package from the HTTP/HTTPS custom file address. Upload the software package to the corresponding custom file address in advance.

  If the software package fails to be uploaded, see **What If a Software Package Fails to Be Uploaded?**
- If you create a component based on an image package, prepare the image by referring to **Component Source**.

Creating and Deploying a Component Based on a Container Using Manual Configurations

Step 1 Log in to ServiceStage.

Step 2 Select a component creation mode.

- To create a component from scratch, go to **Step 3**.
- To create a component based on a component draft, go to **Step 4**.

Step 3 To create a component from scratch, perform the following operations:

1. Use any of the following methods to go to the **Create Component** page.
   - Choose **Component Management** > **Create Component** > **Create Now**.
   - On the **Application Management** page, select the application for which you want to create a component, and click **More** > **Create Now** in the **Operation** column.
   - On the **Application Management** page, click the application for which you want to create a component. On the displayed **Overview** page, click **Create Component**.
2. Go to **Step 5**.

Step 4 To create a component based on a component draft, perform the following operations:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

1. Use either of the following methods to go to the **Create Component** page.
   - On the **Application Management** page, click the application where the component draft is located. The **Component Management** page is displayed. In the component list, click the target component whose **Status** is **Draft**.
   - On the **Component Management** page, click the target component whose **Status** is **Draft**.
2. Go to **Step 5**.

Step 5 In the **Basic Information** area, configure the component by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Component Name	Name of a component, which cannot be changed after the component is created and deployed. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments.
*Component Version	Component version number, which can be automatically generated or customized. • Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0.
*Application	Application to which the component belongs.
*Environment	Select **Kubernetes** or **VM + Kubernetes**. For details, see **Environment Overview**.
*Select Cluster	Select a CCE cluster for deploying and running the component when **Environment** is an HA environment.
*Namespace	Namespace of the container where the component instance is located, which is used to isolate the component instance. For details, see **Managing Namespaces**.
*Deployment Mode	Select **Container**. This parameter is mandatory when **Environment** is set to **VM + Kubernetes**. For details, see **Component Deployment Modes**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Workload Type	A workload is an application running on Kubernetes. No matter how many components are there in your workload, you can run it in a group of Kubernetes pods. A workload is an abstract model of a group of pods in Kubernetes. Select a workload type as required. ● **Stateless**: Deployments do not store any data or status while they are running. Typical scenario: Nginx. ● **Stateful**: StatefulSets store data and statuses while they are running. Typical scenario: MySQL. ● **DaemonSet**: A DaemonSet runs a pod on each node in a cluster and ensures that there is only one pod. This works well for certain system-level applications, such as log collection and resource monitoring, since they must run on each node and need only a few pods. A good example is kube-proxy. DaemonSets are closely related to nodes. If a node becomes faulty, the DaemonSet will not create the same pods on other nodes. ● **Job**: It is a one-time task that runs to completion. It can be executed immediately after being created. It is completed after it exits normally (exit 0). A job is a resource object that is used to control batch tasks. Jobs are different from long-term servo tasks (such as Deployments and StatefulSets). The former is started and terminated at specific times, while the latter runs unceasingly unless being terminated. The pods managed by a job automatically exit after the job is completed based on user configurations. Typical scenarios: database migration, batch computing, and CI/CD pipeline tasks.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description

*Workload	The workload name is automatically generated by default, and can also be customized.
	The default workload name consists of the **Component Name** and **Environment** you set, and six random characters generated by the system. The total length cannot exceed 52 characters.
	● If the **Component Name** contains 45 characters or fewer, the default workload name is:
	`All of the component name-All or part of the environment name-Six random characters generated by the system`
	Underscores (_) in the name will be replaced with hyphens (-), and uppercase letters will be converted to lowercase letters.
	For example, if the component name is **Com_calc** and the environment is **env-cce**, the default workload name is:
	`com-calc-env-cce-1uw8g0`
	● If the **Component Name** contains more than 45 characters, the default workload name is:
	`First 45 characters of the component name-Six random characters generated by the system`
	Underscores (_) in the name will be replaced with hyphens (-), and uppercase letters will be converted to lowercase letters.
	For example, if the component name is **C_aaaa

Parameter	Description
*Label	ServiceStage allows you to add labels of the key-value pair type to manage and select components, for example, for **Configuring a Scheduling Policy of a Component Instance**. Each component can have a maximum of 20 labels. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment creation may fail. Contact your administrator to learn more about tag policies. 1. Click **Add Label**. 2. In the **Add Label** dialog box, click **Add Label**. 3. Enter a key and value. The key cannot be the built-in **app**, **casid**, or **version**. – The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). – The value contains 0 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 4. Click **OK**.
Description	Component description. 1. Click and enter up to 128 characters. 2. Click to save the component description.

Step 6 In the **Component Package** area, configure the component package by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Stack	1. Select a component technology stack type based on the component deployment mode. For details, see **Table 5-1**. Only enabled technology stacks in the **Enabling** state can be selected. For details, see **Enabling a Technology Stack**. If a custom Kubernetes cluster is bound to the environment selected in **Environment** in **Step 5**, only the Docker technology stack is supported. 2. Select a technology stack version from the **Name** drop-down list. Only enabled technology stack versions in the **Enabling** state can be selected. For details, see **Enabling a Technology Stack Version**. 3. Set **JVM** to configure the memory parameter size during Java code running. This parameter is available when a Java or Tomcat technology stack is selected. Click **Stack Settings** and enter the JVM parameter. For example: `-Xms256m -Xmx1024m` Multiple parameters are separated by spaces. 4. Set **Tomcat** to configure the parameters such as Tomcat request path and port number. This parameter is available when a Tomcat technology stack is selected. a. Click **Stack Settings** and select **Tomcat**. The **Tomcat** dialog box is displayed. b. Click **Use Sample Code** and edit the template file based on service requirements. Tomcat settings use the default **server.xml** configuration with **/** as the context path and without an application path. To customize an application path, see **How Do I Customize a Tomcat Context Path?** c. Click **OK**.
*Source Code/ Software Package	• If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. • If you select a software package (JAR, WAR, or ZIP), the software package type is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	If the package type is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file download address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • If the package type is image package: – Select **Multi-container** to enable multi-container deployment. Containers share network and storage resources and collaborate with each other. Click **Select Image Package** to select an image package for the container. Select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source or delete the container.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Image Access Credential	An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.
*Container Name	• By default, the container name is the same as the workload name in **Step 5**. You can also customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type selected for **Upload Method** is image package and multi-container deployment is enabled, you need to customize a container name for the new container. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit.

Step 7 In the **Build Job** area, set build parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

This parameter is mandatory when the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP in **Step 6**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The cluster in the build environment must be able to access the Internet. You are advised to select **Use current environment**. If the cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. • **Use independent environment**: Use an independent build environment to build an image. The clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. • **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	• If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. • If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Namespace	Select the namespace of the cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.
Node Label	In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: • If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. • If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.
YAML Mode	Disable it to use the UI configurations to deploy components.

Step 8 Perform the following operations based on whether to continue setting component parameters:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If yes, click **Next** and go to **Step 9**.
- If no, click **Save as Draft**.

  The component is saved as a draft in the component list of the application selected in **Step 5**. When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. You can use the saved component draft to continue creating a component.

Step 9 In the **Resources** area, configure the resources required by the container-deployed component by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Resources	You can customize **CPU** and **Memory** to set their quota, and set the maximum/minimum number of CPU cores and memory size (GiB) that can be used by components. In this way, you can set the number of resources required by each component instance. When resource **Request** is specified for a component instance, the cluster uses this information to determine the node to which the component instance is scheduled. When resource **Limit** is specified for a component instance, the cluster ensures that the resources occupied by the running component instance do not exceed the limit. The cluster will also reserve the requested number of system resources for the component instance. For details about how to configure them, see **Resource Management**. To modify them, select the item to be changed and enter a new value. Unselected parameters indicate no restriction. If the package type selected for **Upload Method** in **Step 6** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.
*Instances	When **Workload Type** in **Step 5** is **Stateless**, **Stateful**, or **Job**, set the number of component instances running in the environment. The value ranges from 1 to 200.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 5-3 Configuring the resources required by the container-deployed component

Step 10 In the **Headless Service Settings** area, set the service name and port number of the component by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

This operation is required when **Workload Type** is **Stateful** in **Step 5**. This operation enables pods in a StatefulSet to access each other and provides a fixed access domain name for each pod.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 11 In the **Access Mode** area, enable **Public Network Access**.

Access modes can be configured if **Workload Type** of the component in **Step 5** is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the selected environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If a component is upgraded and maintained in ELB dark launch mode after being created and deployed, you need to enable public network access for the component. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

Click to enable public access and set the following parameters:

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	▪ **Creating a Shared Load Balancer**. ▪ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

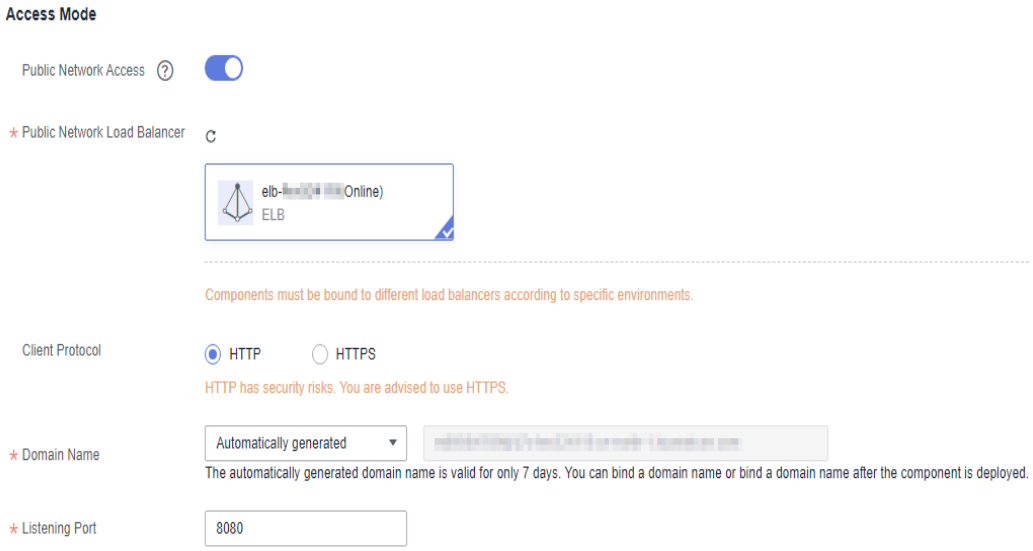
2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.
4. Set **Listening Port**.

   Enter the listening port number of the application process.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 5-4 Configuring public access



Step 12 In the **Local Time** area, set the time zone of the container.

By default, the time zone is the same as that of the region where the container node is located.

Step 13 Set **Cloud Service Settings**.

Cloud Service Settings can be set only if a CCE cluster is bound to the environment selected in **Step 5**.

Operation Type	Operation	Description
Microservice Engine	Binding a microservice engine	**Cloud Service Engine** is a one-stop management platform for microservice solutions. After microservice components are connected to CSE, you can use service registry, service governance, and configuration management of CSE. In this way, you can focus on service development and improve product delivery efficiency and quality. Choose **Cloud Service Settings** > **Microservice Engine** and bind a microservice engine. For details, see **Binding a Microservice Engine**. • If the package type selected for **Upload Method** in **Step 6** is image package and multi-container deployment is enabled, the microservice engine cannot be bound. • To upgrade and maintain components in dark launch mode, bind the ServiceComb engine or registry/configuration center. For details, see **Upgrading a Component in Dark Launch Mode**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation Type	Operation	Description
Distributed Cache Service	Binding a distributed cache	In a distributed system, the distributed cache service provides scalable and reliable user session management. Choose **Cloud Service Settings** > **Distributed Cache Service** and bind a distributed cache. For details, see **Binding a Distributed Cache**.
Relational Database Service	Binding a cloud database	RDS is required for persistent storage of application data. Choose **Cloud Service Settings** > **Relational Database Service** and bind a cloud database. For details, see **Binding a Cloud Database**.
GaussDB	Binding a GaussDB instance	GaussDB is required for persistent storage of application data. Choose **Cloud Service Settings** > **GaussDB** and bind a GaussDB instance. For details, see **Binding a GaussDB Instance**.
Distributed Message Service for RocketMQ	Binding a DMS for RocketMQ instance	DMS for RocketMQ is required for application components to store and consume messages. Choose **Cloud Service Settings** > **Distributed Message Service for RocketMQ** and bind a RocketMQ instance. For details, see **Binding a DMS for RocketMQ Instance**.
Cloud Search Service	Binding a CSS instance	Uses the multi-condition search, statistics, and report capabilities of structured and unstructured text provided by the Elasticsearch cluster of Cloud Search Service (CSS) to analyze component logs in all scenarios. Choose **Cloud Service Settings** > **Cloud Search Service** and bind a CSS instance. For details, see **Binding a CSS Instance**.

Step 14 Set **Container Settings**.

If the package type selected for **Upload Method** in **Step 6** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation Type	Operation	Description
Environment Variable	Configuring environment variables	Environment variables are set in the container running environment and can be modified after component deployment to ensure application flexibility. Environment variables set for a component are local environment variables and take effect only for this component. Choose **Container Settings** > **Environment Variable** and configure environment variables. For details, see **Adding an Environment Variable of a Component**.
Startup Command	Configuring a startup command	A startup command is used to start a container. Choose **Container Settings** > **Startup Command** and configure the startup command. For details, see **Configuring the Startup Command and Lifecycle of a Component**.
Data Storage	Configuring data storage	Container storage is a component that provides storage for applications. Multiple types of storage are supported. A component can use any amount of storage. Cloud storage can be configured if a CCE cluster is bound to the environment selected in **Step 5**. Choose **Container Settings** > **Data Storage** and configure data storage. For details, see **Configuring Data Storage**.
Lifecycle	Configuring the lifecycle	For container-deployed components, ServiceStage provides callback functions for the lifecycle management of applications. For example, if you want an application component to perform a certain operation before stopping, you can register a hook function. Choose **Container Settings** > **Lifecycle** and configure the lifecycle. For details, see **Configuring the Startup Command and Lifecycle of a Component**.
Log Collection	Configuring log collection	For container-deployed components, ServiceStage supports setting of application log policies. You can view related logs on the AOM console. You can configure a log policy during or after component deployment. If no configuration is performed, the system collects standard application output logs by default. Choose **Container Settings** > **Log Collection** and configure log collection. For details, see **Configuring a Log Policy of an Application**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation Type	Operation	Description
Health Check	Configuring health check	Health check periodically checks application health status during running of container-deployed components. Health check can be configured if **Workload Type** of the component in **Step 5** is **Stateless**, **Stateful**, or **DaemonSet**. Choose **Container Settings** > **Health Check** and configure health check. For details, see **Configuring Health Check**.

Step 15 Set **Application Settings**.

Set the component configuration file by referring to **Managing Application Settings of a Container-Deployed Component**. This implements multi-environment component release and upgrade with one-time configuration through file mounting.

If the package type selected for **Upload Method** in **Step 6** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.

Step 16 Set **Advanced Settings**.

Operation Type	Operation	Description
Upgrade Policy	Configuring an upgrade policy	Upgrade policy can be configured if **Workload Type** of the component in **Step 5** is **Stateless**, **Stateful**, or **DaemonSet**. In practice, component upgrade is a common scenario. Components deployed based on a container in ServiceStage support the following upgrade policies: • RollingUpdate: New pods are created gradually, and then old pods are deleted. This policy is used by default when you **create and deploy a component based on a container**, **upgrade a component in rolling release mode**, **upgrade a component in dark launch mode**, and **upgrade components in batches**. • Recreate: Old pods are deleted, and then new pods are created. This policy is used by default when you **upgrade a component in single-batch release mode**. Choose **Advanced Settings** > **Upgrade Policy** and configure the upgrade policy. For details, see **Configuring a Component Upgrade Policy**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation Type	Operation	Description
Scheduling Policy	Configuring the scheduling policy	Scheduling policy can be configured if **Workload Type** of the component in **Step 5** is **Stateless**, **Stateful**, or **DaemonSet**. For container-deployed components, ServiceStage divides the components into minimum deployment instances based on the deployment features of the components. The application scheduler monitors application instance information in real time. When detecting that a new pod needs to be scheduled, the application scheduler calculates all remaining resources (compute, storage, and network resources) in the cluster to obtain the most appropriate scheduling target node. Choose **Advanced Settings** > **Scheduling Policy** and configure the scheduling policy. For details, see **Configuring a Scheduling Policy of a Component Instance**.
Tolerance Policy	Configuring the tolerance policy	Tolerance policy can be configured if **Workload Type** of the component in **Step 5** is **Stateless**, **Stateful**, or **DaemonSet**. For container-deployed components, tolerations allow the scheduler to schedule pods to nodes with target taints. Tolerances work with node taints. Each node allows one or more taints. If no tolerance is configured for a pod, the scheduler will schedule the pod based on node taint policies to prevent the pod from being scheduled to an inappropriate node. Choose **Advanced Settings** > **Tolerance Policy** and configure the tolerance policy. For details, see **Configuring a Tolerance Policy of a Component Instance**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation Type	Operation	Description
Performance Management	Configuring performance management	Performance management can be configured if **Workload Type** of the component in **Step 5** is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the selected environment. The Application Performance Management (APM) service helps you quickly locate problems and identify performance bottlenecks to improve your experience. ServiceStage allows you to configure application performance management during or after component deployment. APM can be configured for components whose technology stack type is Java, Tomcat, or Docker. Choose **Advanced Settings** > **Performance Management** and configure performance management. For details, see **Configuring Performance Management of a Container-Deployed Component**.
Custom Monitoring	Configuring custom monitoring	Custom monitoring can be configured if **Workload Type** of the component in **Step 5** is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the selected environment. ServiceStage allows you to obtain monitoring data based on custom metrics. You can set custom metric monitoring during or after component deployment. Components deployed based on CCE support custom monitoring. Choose **Advanced Settings** > **O&M Policy** and configure custom monitoring. For details, see **Configuring Custom Monitoring of a Component**.
Job	Configuring job settings	Job settings can be configured if **Workload Type** of the component in **Step 5** is **Job**. Configure job settings to complete batch processing tasks efficiently and reliably. You can configure job parameters to implement flexible task management and resource optimization. Choose **Application Settings** > **Job** and configure job settings. For details, see **Configuring Component Job Settings**.

Step 17 Perform subsequent operations based on whether to deploy the component by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Deploy or Not	Procedure
Yes	Click **Create and Deploy**. • On the **View Deployment History** page, view the deployment logs and wait until the component deployment is complete. If "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details. • If the technology stack type of the component selected in **Step 6** is Java, Tomcat, Node.js, Python, or PHP, the system creates a build job of the **System created** type after the component is deployed. For details, see **Viewing Build Jobs**. • When deploying a component whose **Workload Type** is **Stateless**, **Stateful**, or **DaemonSet** in **Step 5**, you can click **Suspend** to stop the deployment if the component configuration is incorrect. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Single Component**.
No	Click **Save as Draft**. The component is saved as a draft in the component list of the application selected in **Step 5**. When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. You can use the saved component draft to continue creating a component.

----End

Follow-up Operations

- **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations** for **Draft** components.
- **Upgrading a Component in Single-batch Release Mode** and **Upgrading a Component in Rolling Release Mode** for **Running**, **Not ready**, **Abnormal**, and **Failed** components.
- **Upgrading a Component in Dark Launch Mode** for **Running**, **Not ready**, and **Abnormal** components.
- **Upgrading Components in Batches** for **Running**, **Not ready**, and **Abnormal** components.
- **Cloning Components in Batches** for **Running**, **Not ready**, and **Abnormal** components.
- **Configuring a Scaling Policy of a Component Instance** for **Running** and **Not ready** components.
- Maintain components by referring to **Component O&M** after the components are created and deployed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For **Running** components with public access enabled, you can **Changing the Component Access Domain Name** to access services provided by the component using domain names.
- For **Running** components, you can **Configuring the Component Access Mode** to access the services provided by the component in a preset mode.

5.2.2 Creating a Component Based on a Container Using YAML Configurations

ServiceStage supports a maximum of 15,000 components in the same project.

This section describes how to create a component, save a component draft, and deploy a component based on a container using YAML configurations so that you can deploy components using the Kubernetes deployment template. This is applicable to users who are familiar with CCE.

When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. During component creation, you can click **Save as Draft** to prevent component parameter settings from being lost.

Prerequisites

- An application has been created because components can only be added to applications. For details, see **Creating an Application**.
- An environment supported by container-based deployment has been created and resources have been managed because components need to be deployed in an environment.
  - For the environment types supported by container-based deployment, see **Component Deployment Modes**.
  - For details about how to create an environment and manage resources, see **Environment Management**.
- An organization has been created because images generated during component build need to be managed by organization. For details, see **Creating an Organization**.
- A namespace has been bound to the CCE cluster to isolate build data and component instances. For details, see **Managing Namespaces**.
- If you create a component based on a source code repository, repository authorization has been created. For details, see **Creating Repository Authorization**.
- If you create a component based on a software package:
  - Upload the software package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Upload the software package to the OBS bucket. For details, see **Streaming Upload (PUT)**.
  - If the package type is software package and you need to use JFrog (example) as the software package repository, you can download the package from the HTTP/HTTPS custom file address. Upload the software package to the corresponding custom file address in advance.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If the software package fails to be uploaded, see **What If a Software Package Fails to Be Uploaded?**

- If you create a component based on an image package, prepare the image by referring to **Component Source**.

Creating and Deploying a Component Based on a Container Using YAML Configurations

Step 1 Log in to ServiceStage.

Step 2 Select a component creation mode.

- To create a component from scratch, go to **Step 3**.
- To create a component based on a component draft, go to **Step 4**.

Step 3 To create a component from scratch, perform the following operations:

1. Use any of the following methods to go to the **Create Component** page.
   - Choose **Component Management** > **Create Component** > **Create Now**.
   - On the **Application Management** page, select the application for which you want to create a component, and click **More** > **Create Now** in the **Operation** column.
   - On the **Application Management** page, click the application for which you want to create a component. On the displayed **Overview** page, click **Create Component** > **Create Now**.
2. Go to **Step 5**.

Step 4 To create a component based on a component draft, perform the following operations:

1. Use either of the following methods to go to the **Create Component** page.
   - On the **Application Management** page, click the application where the component draft is located. The **Component Management** page is displayed. In the component list, click the target component whose **Status** is **Draft**.
   - On the **Component Management** page, click the target component whose **Status** is **Draft**.
2. Go to **Step 5**.

Step 5 In the **Basic Information** area, configure the component by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Component Name	Name of a component, which cannot be changed after the component is created and deployed. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments.
*Component Version	Component version number, which can be automatically generated or customized. • Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0.
*Application	Application to which the component belongs.
*Environment	Select **Kubernetes** or **VM + Kubernetes**. For details, see **Environment Overview**.
*Select Cluster	Select a CCE cluster for deploying and running the component when **Environment** is an HA environment.
*Namespace	Namespace of the container where the component instance is located, which is used to isolate the component instance. For details, see **Managing Namespaces**.
*Deployment Mode	Select **Container**. This parameter is mandatory when **Environment** is set to **VM + Kubernetes**. For details, see **Component Deployment Modes**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Workload Type	A workload is an application running on Kubernetes. No matter how many components are there in your workload, you can run it in a group of Kubernetes pods. A workload is an abstract model of a group of pods in Kubernetes. Select a workload type as required. ● **Stateless**: Deployments do not store any data or status while they are running. Typical scenario: Nginx. ● **Stateful**: StatefulSets store data and statuses while they are running. Typical scenario: MySQL. ● **DaemonSet**: A DaemonSet runs a pod on each node in a cluster and ensures that there is only one pod. This works well for certain system-level applications, such as log collection and resource monitoring, since they must run on each node and need only a few pods. A good example is kube-proxy. DaemonSets are closely related to nodes. If a node becomes faulty, the DaemonSet will not create the same pods on other nodes. ● **Job**: It is a one-time task that runs to completion. It can be executed immediately after being created. It is completed after it exits normally (exit 0). A job is a resource object that is used to control batch tasks. Jobs are different from long-term servo tasks (such as Deployments and StatefulSets). The former is started and terminated at specific times, while the latter runs unceasingly unless being terminated. The pods managed by a job automatically exit after the job is completed based on user configurations. Typical scenarios: database migration, batch computing, and CI/CD pipeline tasks.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description

*Workload	The workload name is automatically generated by default, and can also be customized.
	The default workload name consists of the **Component Name** and **Environment** you set, and six random characters generated by the system. The total length cannot exceed 52 characters.
	● If the **Component Name** contains 45 characters or fewer, the default workload name is:
	`All of the component name-All or part of the environment name-Six random characters generated by the system`
	Underscores (_) in the name will be replaced with hyphens (-), and uppercase letters will be converted to lowercase letters.
	For example, if the component name is **Com_calc** and the environment is **env-cce**, the default workload name is:
	`com-calc-env-cce-1uw8g0`
	● If the **Component Name** contains more than 45 characters, the default workload name is:
	`First 45 characters of the component name-Six random characters generated by the system`
	Underscores (_) in the name will be replaced with hyphens (-), and uppercase letters will be converted to lowercase letters.
	For example, if the component name is **C_aaaa

Parameter	Description
*Label	ServiceStage allows you to add labels of the key-value pair type to manage and select components, for example, for **Configuring a Scheduling Policy of a Component Instance**. Each component can have a maximum of 20 labels. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment creation may fail. Contact your administrator to learn more about tag policies. 1. Click **Add Label**. 2. In the **Add Label** dialog box, click **Add Label**. 3. Enter a key and value. The key cannot be the built-in **app**, **casid**, or **version**. – The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). – The value contains 0 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 4. Click **OK**.
Description	Component description. 1. Click and enter up to 128 characters. 2. Click to save the component description.

Step 6 In the **Component Package** area, configure the component package by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Stack	1. Select a component technology stack type based on the component deployment mode. For details, see **Table 5-1**. Only enabled technology stacks in the **Enabling** state can be selected. For details, see **Enabling a Technology Stack**. If a custom Kubernetes cluster is bound to the environment selected in **Environment** in **Step 5**, only the Docker technology stack is supported. 2. Select a technology stack version from the **Name** drop-down list. Only enabled technology stack versions in the **Enabling** state can be selected. For details, see **Enabling a Technology Stack Version**. 3. Set **JVM** to configure the memory parameter size during Java code running. This parameter is available when a Java or Tomcat technology stack is selected. Click **Stack Settings** and enter the JVM parameter. For example: `-Xms256m -Xmx1024m` Multiple parameters are separated by spaces. 4. Set **Tomcat** to configure the parameters such as Tomcat request path and port number. This parameter is available when a Tomcat technology stack is selected. a. Click **Stack Settings** and select **Tomcat**. The **Tomcat** dialog box is displayed. b. Click **Use Sample Code** and edit the template file based on service requirements. In Tomcat configuration, the default **server.xml** configuration is used. The context path is **/**, and no application path is specified. If you need to customize an application path, customize the Tomcat context path by referring to **How Do I Customize a Tomcat Context Path?** c. Click **OK**.
*Source Code/ Software Package	• If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. • If you select a software package (JAR, WAR, or ZIP), the software package type is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	If the package type is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • If the package type is image package: – Select **Multi-container** to enable multi-container deployment. Containers share network and storage resources and collaborate with each other. Click **Select Image Package** to select an image package for the container. Select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source or delete the container.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Image Access Credential	An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.
*Container Name	• By default, the container name is the same as the workload name in **Step 5**. You can also customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type selected for **Upload Method** is image package and multi-container deployment is enabled, you need to customize a container name for the new container. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit.

Step 7 In the **Build Job** area, set build parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

This parameter is mandatory when the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP in **Step 6**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The cluster in the build environment must be able to access the Internet. You are advised to select **Use current environment**. If the cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. • **Use independent environment**: Use an independent build environment to build an image. The clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. • **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	• If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. • If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Namespace	Select the namespace of the cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.
Node Label	You can use a node label to deliver the build job to a fixed node bound with an EIP. In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: • If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. • If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.
YAML Mode	Enable it by clicking to use the YAML configurations to deploy component.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 8 Perform the following operations based on whether to continue setting component parameters:

- If yes, click **Next** and go to **Step 9**.
- If no, click **Save as Draft**.

  The component is saved as a draft in the component list of the application selected in **Step 5**. When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. You can use the saved component draft to continue creating a component.

Step 9 In the **Access Mode** area, enable **Public Network Access**.

Access modes can be configured if **Workload Type** of the component in **Step 5** is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the selected environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If a component is upgraded and maintained in ELB dark launch mode after being created and deployed, you need to enable public network access for the component. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

Click to enable public access and set the following parameters:

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources, refer to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

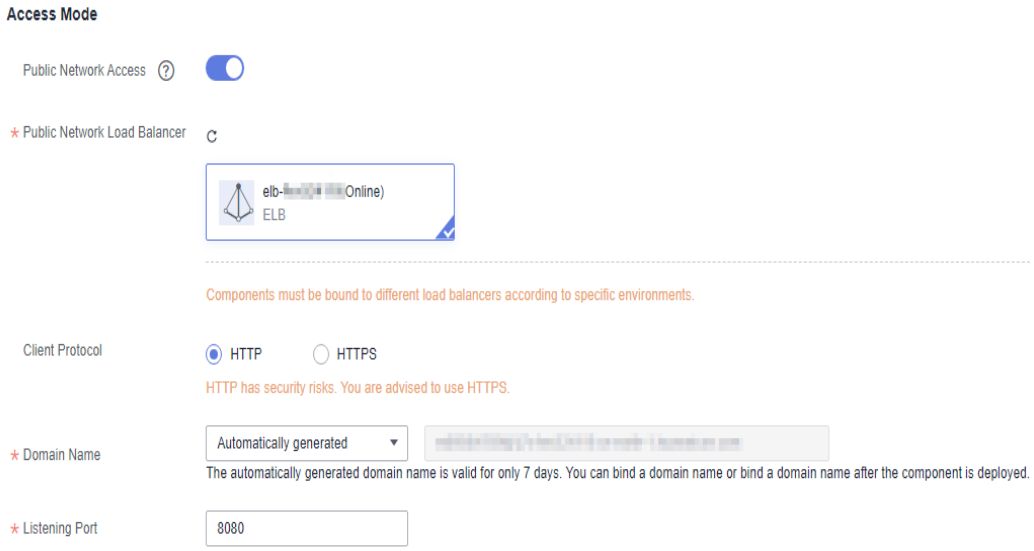
Scenario	Reference
Use the domain name to access an application.	▪ **Creating a Shared Load Balancer**. ▪ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.
4. Set **Listening Port**.

   Enter the listening port number of the application process.

Figure 5-5 Configuring public access



Step 10 In the **Headless Service Settings** area, set the service name and port number of the component by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

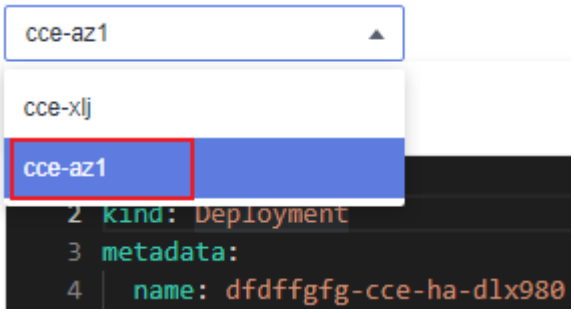
This operation is required when **Workload Type** is **Stateful** in **Step 5**. This operation enables pods in a StatefulSet to access each other and provides a fixed access domain name for each pod.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 11 If an HA environment is selected in **Step 5**, select the CCE cluster where the component is deployed and running.

For details about the HA environment, see **Creating a Kubernetes Environment in Resource Management Mode**.

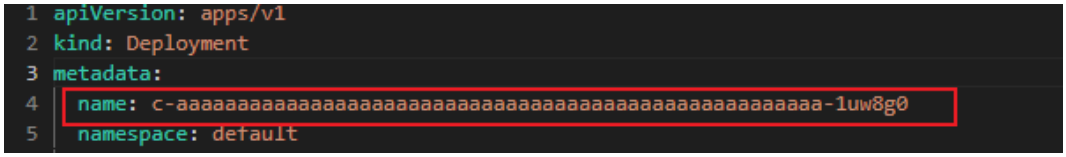


Step 12 Import or edit the YAML configuration file of the component.

- Click **Import YAML File** to import the edited YAML configuration file.
- Edit the configuration parameters as required.

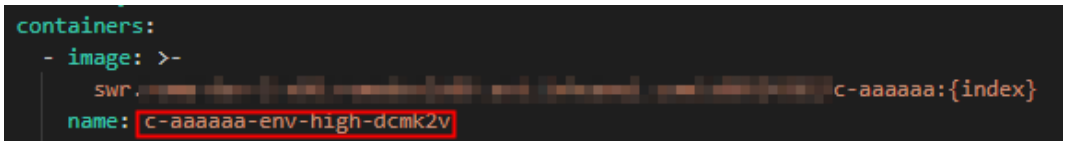
The parameters in the YAML configuration file are described in **Deployment**.

You can change **name** in the YAML configuration by referring to the description of **Workload Name** in **Step 5**. The workload name cannot be changed after the component is created and deployed.



You can change **name** in the YAML configuration by referring to the description of **Container Name** in **Step 6**. After a component is created and deployed, you can also change the container name by **Upgrading a Single Component**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.



Step 13 Set **Application Settings**.

Set the component configuration file by referring to **Managing Application Settings of a Container-Deployed Component**. This implements multi-environment component release and upgrade with one-time configuration through file mounting.

If the package type selected for **Upload Method** in **Step 6** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.

Step 14 Set **Log Settings**. Determine whether to collect component run logs to LTS by referring to the following table.

Log Settings can be set if a CCE cluster is bound to the environment selected in **Step 5**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Whether to Collect Component Run Logs to LTS	Description
Yes	Select **Auto connection to LTS**. The system determines the log collection path based on your settings in **Step 12**. • If container mounting is configured, logs are delivered to the container file path of LTS. • If host mounting is configured, logs are delivered to the node file path of LTS. • If both container and node mounting are configured, only the node mounting path can be retained due to LTS restrictions, and logs are delivered only to the node file path of LTS. • If you do not configure log collection, the log path is **/var/log/*** or **/var/logs/***, which corresponds to the container file path of LTS. After a component is configured and deployed, the system automatically generates log group and log stream configurations, and generates access configurations and host groups on the **Ingestion Management** page of the LTS console. • **Log Group**: The naming rule is **servicestage-***{Application ID}*. Use the ID of the application to which the component belongs. • **Log Stream**: The naming rule is **lts-topic-***{Component name}*. Use the name of the component connected to LTS. • **Ingestion**: The naming rule is **access-config-***{Component name}*. Use the name of the component connected to LTS. • **Host Group**: The naming rule is **k8s-log-***{CCE cluster ID}*. Use the ID of the CCE cluster where the component connected to LTS is located.
No	Do not select **Auto connection to LTS**. • After the component is configured and deployed, you can view running logs on the AOM console. For details, see **Viewing Log Files**. • After the component is configured and deployed, you can collect component logs to LTS by referring to **Managing Component LTS Logs**.

Step 15 Perform subsequent operations based on whether to deploy the component by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Deploy or Not	Procedure
Yes	Click **Create and Deploy**. • On the **View Deployment History** page, view the deployment logs and wait until the component deployment is complete. If "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details. • If the technology stack type of the component selected in **Step 6** is Java, Tomcat, Node.js, Python, or PHP, the system creates a build job of the **System created** type after the component is deployed. For details, see **Viewing Build Jobs**. • When deploying a component whose **Workload Type** is **Stateless**, **Stateful**, or **DaemonSet** in **Step 5**, you can click **Suspend** to stop the deployment if the component configuration is incorrect. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Single Component**.
No	Click **Save as Draft**. The component is saved as a draft in the component list of the application selected in **Step 5**. When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. You can use the saved component draft to continue creating a component.

----End

Follow-up Operations

- **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations** for **Draft** components.
- **Upgrading a Component in Single-batch Release Mode** and **Upgrading a Component in Rolling Release Mode** for **Running**, **Not ready**, **Abnormal**, and **Failed** components.
- **Upgrading a Component in Dark Launch Mode** for **Running**, **Not ready**, and **Abnormal** components.
- **Upgrading a Component in Dark Launch Mode** for **Running**, **Not ready**, and **Abnormal** components.
- **Cloning Components in Batches** for **Running**, **Not ready**, and **Abnormal** components.
- **Component Management** for **Running** and **Not ready** components.
- Maintain components by referring to **Component O&M** after the components are created and deployed.
- For **Running** components with public access enabled, you can **Changing the Component Access Domain Name** to access services provided by the component using domain names.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For **Running** components, you can **Configuring the Component Access Mode** to access the services provided by the component in a preset mode.

5.2.3 Creating a Component Based on a VM

ServiceStage supports a maximum of 15,000 components in the same project.

This section describes how to create a component, save a component draft, and deploy a component based on a VM.

When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. During component creation, you can click **Save as Draft** to prevent component parameter settings from being lost.

Prerequisites

- An application has been created because components can only be added to applications. For details, see **Creating an Application**.
- An environment supported by VM-based deployment has been created and resources have been managed because components need to be deployed in an environment.
  - For the environment types supported by VM-based deployment, see **Component Deployment Modes**.
  - For details about how to create an environment and manage resources, see **Environment Management**.
- Create a component based on a software package:
  - Upload the software package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Upload the software package to the OBS bucket. For details, see **Streaming Upload (PUT)**.
  - If the package type is software package and you need to use JFrog (example) as the software package repository, you can download the package from the HTTP/HTTPS custom file address. Upload the software package to the corresponding custom file address in advance.

  If the software package fails to be uploaded, see **What If a Software Package Fails to Be Uploaded?**

Creating and Deploying a Component Based on a VM

Step 1 Log in to ServiceStage.

Step 2 Select a component creation mode.

- To create a component from scratch, go to **Step 3**.
- To create a component based on a component draft, go to **Step 4**.

Step 3 To create a component from scratch, perform the following operations:

1. Use any of the following methods to go to the **Create Component** page.
   - Choose **Component Management** > **Create Component** > **Create Now**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- On the **Application Management** page, select the application for which you want to create a component, and click **More** > **Create Now** in the **Operation** column.
- On the **Application Management** page, click the application for which you want to create a component. On the displayed **Overview** page, click **Create Component** > **Create Now**.

2. Go to **Step 5**.

Step 4 To create a component based on a component draft, perform the following operations:

1. Use either of the following methods to go to the **Create Component** page.
   - On the **Application Management** page, click the application where the component draft is located. The **Component Management** page is displayed. In the component list, click the target component whose **Status** is **Draft**.
   - On the **Component Management** page, click the target component whose **Status** is **Draft**.
2. Go to **Step 5**.

Step 5 In the **Basic Information** area, configure the component by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Component Name	Name of a component, which cannot be changed after the component is created and deployed. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different environments.
*Component Version	Component version number, which can be automatically generated or customized. • Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0.
*Application	Application to which the component belongs.
*Environment	Select **VM** or **VM + Kubernetes**. For details, see **Environment Overview**.
*Deployment Mode	Select **VM**. This parameter is mandatory when **Environment** is set to **VM + Kubernetes**. For details, see **Component Deployment Modes**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Labels	ServiceStage allows you to add labels of the key-value pair type to manage and select components. Each component can have a maximum of 20 labels. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment creation may fail. Contact your administrator to learn more about tag policies. 1. Click **Add Label**. 2. In the **Add Label** dialog box, click **Add Label**. 3. Enter a key and value. – The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). The key cannot be the built-in **app**, **casid**, or **version**. – The value contains 0 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 4. Click **OK**.
Description	Component description. 1. Click and enter up to 128 characters. 2. Click to save the component description.

Step 6 In the **Component Package** area, configure the component package by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Stack	1. Select a component technology stack type based on the component deployment mode. For details, see **Table 5-1**. Only enabled technology stacks in the **Enabling** state can be selected. For details, see **Enabling a Technology Stack**. 2. Select a technology stack version from the **Name** drop-down list. Only enabled technology stack versions in the **Enabling** state can be selected. For details, see **Enabling a Technology Stack Version**. 3. Set **JVM** to configure the memory parameter size during Java code running. This parameter is available when a Java or Tomcat technology stack is selected. Click **Stack Settings** and enter the JVM parameter. For example: `-Xms256m -Xmx1024m` Multiple parameters are separated by spaces. 4. Set **Tomcat** to configure the parameters such as Tomcat request path and port number. This parameter is available when a Tomcat technology stack is selected. a. Click **Stack Settings** and select **Tomcat**. The **Tomcat** dialog box is displayed. b. Click **Use Sample Code** and edit the template file based on service requirements. In Tomcat configuration, the default **server.xml** configuration is used. The context path is **/**, and no application path is specified. If you need to customize an application path, customize the Tomcat context path by referring to **How Do I Customize a Tomcat Context Path?** c. Click **OK**.
*Source Code/ Software Package	Select a software package (JAR package, WAR package, ZIP package, or package). The package type is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	If the package type is software package, select an uploaded software package. For details about the upload method, see **Component Source**. If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. – If authentication is disabled, any user can download the software package in the custom file address by default. – Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

Step 7 Perform the following operations based on whether to continue setting component parameters:

- If yes, click **Next** and go to **Step 5**.
- If no, click **Save as Draft**.

  The component is saved as a draft in the component list of the application selected in **Step 5**. When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. You can use the saved component draft to continue creating a component.

Step 8 In the **Resources** area, set the resources required by the component.

- If **ECS** is selected for **Resource Type**, select the ECS that has been managed in the component deployment environment.
- If **AS** is selected for **Resource Type**, select the AS group to be used from **Resources** and then select the ECS in the AS group.
- If **Custom servers** is selected for **Resource Type**, select the custom server that has been managed in the component deployment environment.

The selected servers must have the VM agent installed and be running properly.

- If the status of the target server agent indicates that the agent is missing and to install it first, click **Install**. For details, see **Step 6** to **Step 8** in **Installing a VM Agent**.
- If the target ECS agent is offline, see **What Should I Do If the VM Agent Is Offline?**
- You can select up to 200 servers of the same resource type.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 9 In the **Access Mode** area, enable **Public Network Access**.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

Click to enable public access and set the following parameters:

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources:
     - Create a shared load balancer by referring to **Creating a Shared Load Balancer**.
     - Create a dedicated application load balancer by referring to **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.

   After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

Step 10 Set **Cloud Service Settings**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation Type	Operation	Description
Microservice Engine	Binding a microservice engine	**Cloud Service Engine** is a one-stop management platform for microservice solutions. After microservice components are connected to CSE, you can use service registry, service governance, and configuration management of CSE. In this way, you can focus on service development and improve product delivery efficiency and quality. Choose **Cloud Service Settings** > **Microservice Engine** and bind a microservice engine. For details, see **Binding a Microservice Engine**.
Distributed Cache Service	Binding a distributed cache	In a distributed system, the distributed cache service provides scalable and reliable user session management. Choose **Cloud Service Settings** > **Distributed Cache Service** and bind a distributed cache. For details, see **Binding a Distributed Cache**.
Relational Database Service	Binding a cloud database	RDS is required for persistent storage of application data. Choose **Cloud Service Settings** > **Relational Database Service** and bind a cloud database. For details, see **Binding a Cloud Database**.
GaussDB	Binding a GaussDB instance	GaussDB is required for persistent storage of application data. Choose **Cloud Service Settings** > **GaussDB** and bind a GaussDB instance. For details, see **Binding a GaussDB Instance**.
Distributed Message Service for RocketMQ	Binding a DMS for RocketMQ instance	DMS for RocketMQ is required for application components to store and consume messages. Choose **Cloud Service Settings** > **Distributed Message Service for RocketMQ** and bind a RocketMQ instance. For details, see **Binding a DMS for RocketMQ Instance**.
Cloud Search Service	Binding a CSS instance	Uses the multi-condition search, statistics, and report capabilities of structured and unstructured text provided by the Elasticsearch cluster of Cloud Search Service (CSS) to analyze component logs in all scenarios. Choose **Cloud Service Settings** > **Cloud Search Service** and bind a CSS instance. For details, see **Binding a CSS Instance**.

Step 11 Set **VM Settings**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Set VM settings by referring to **Configuring VM Settings**.

Step 12 Set **Application Settings**.

Bind the created configuration file to the component by referring to **Managing Application Settings of a VM-Deployed Component**. This implements multi-environment component release and upgrade with one-time configuration through file mounting.

Step 13 Set **Advanced Settings**.

For details, see **Configuring Application Performance Management of a VM-Deployed Component**.

Step 14 Set **Log Settings**.

Log Settings can be set for components deployed on the ECS managed in the environment. For details, see **Configuring Log Settings of a VM-Deployed Component**.

Step 15 Perform subsequent operations based on whether to deploy the component by referring to the following table.

Deploy or Not	Procedure
Yes	Click **Create and Deploy**. • On the **View Deployment History** page, view the deployment logs and wait until the component deployment is complete. If "Creation of the VM application instance failure", "Query of the VM task status failure", or "Query of the VM application instance status failure" is displayed, click **View Event** to view details in the event list. • During component deployment, if the component configuration is incorrect, you can click **Suspend** to stop the deployment. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Component in Rolling Release Mode**.
No	Click **Save as Draft**. The component is saved as a draft in the component list of the application selected in **Step 5**. When you create a component but do not complete the parameter settings, you can save the draft to prevent the parameters from being lost due to factors such as page closing. You can use the saved component draft to continue creating a component.

----End

Follow-up Operations

- **Creating a Component Based on a VM** for **Draft** components.
- **Upgrading a Component in Rolling Release Mode** for **Running**, **Not ready**, **Abnormal**, and **Failed** components.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Upgrading Components in Batches** for **Running**, **Not ready**, and **Abnormal** components.
- **Cloning Components in Batches** for **Running**, **Not ready**, and **Abnormal** components.
- **Configuring a Manual Scaling Policy** for **Running** and **Not ready** components.
- Maintain components by referring to **Component O&M** after the components are created and deployed.
- For **Running** components with public access enabled, you can **Changing the Component Access Domain Name** to access services provided by the component using domain names.

5.2.4 Creating a Component by Importing a CCE Workload

ServiceStage supports a maximum of 15,000 components in the same project.

This section describes how to import a workload from CCE to create and deploy a component.

This method is applicable only to the container-based deployment, and the component deployment environment must be bound to a CCE cluster.

Prerequisites

- An application has been created because components can be imported only under an application. For details, see **Creating an Application**.
- An environment supported by container-based deployment has been created and bound with resources such as the CCE cluster of the target workload for deploying and running components in an environment.
  - For the environment types supported by container-based deployment, see **Component Deployment Modes**.
  - For details about how to create an environment and manage resources, see **Environment Management**.
- A namespace has been created to isolate build data. For details, see **Managing Namespaces**.
- The target stateless or stateful workload has been created on CCE and has not been imported. For details, see **Creating a Workload**.

  To enable multi-container deployment, add and configure multiple containers when creating a workload. You can specify a container as the initialization container to install specific or scripts before the service container is started. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**.

  When a workload is imported to create a component, the component package parameters are fixed to the settings set during workload creation and cannot be modified.
- A Docker technology stack has been enabled and is in the **Enabling** state. If the Docker technology stack is not enabled, enable it by referring to **Enabling a Technology Stack**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Creating a Component by Importing a CCE Workload

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to open the **Select Component** dialog box:

- Choose **Component Management** > **Import Component**.
- On the **Application Management** page, click the application for which you want to create a component. On the displayed **Overview** page, click **Import Component**.

Step 3 In the **Select Component** dialog box, configure the target workload by referring to the following table and click **OK**.

Parameter	Description
Select an environment	Component deployment environment. Select a Kubernetes or VM + Kubernetes environment of the CCE cluster where the workload is located. If the workload cluster is not bound to any Kubernetes or VM + Kubernetes environment, bind the cluster to an environment by referring to **Binding a Cluster**.
CCE Cluster	CCE cluster to which the workload to be imported belongs.
Namespace	Namespace to which the workload to be imported belongs.
Workload Type	Type of the workload to be imported. • **Stateless**: Deployments do not store any data or status while they are running. Typical scenario: Nginx. • **Stateful**: StatefulSets store data and statuses while they are running. Typical scenario: MySQL.
Workload	Name of the workload to be imported. A workload that has been imported cannot be imported again.

Step 4 In the **Basic Information** area, configure the component by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Component Name	Name of a component, which cannot be changed after the component is created and deployed. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Component Version	Component version number, which can be automatically generated or customized. • Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0.
*Application	Application to which the component belongs.
*Environment	Component deployment environment. The value is fixed to the environment selected in **Step 3**.
*Deployment Mode	The value is fixed to **Container**. This parameter is available when a VM + Kubernetes environment is selected in **Step 3**. For details, see **Environment Overview**.
*Namespace	The value is the namespace to which the workload selected in **Step 3** belongs.
*Workload Type	The value is the workload type selected in **Step 3**.
*Workload	The value is the workload name selected in **Step 3**.
*Label	ServiceStage allows you to add labels of the key-value pair type to manage and select components, for example, for **Configuring a Scheduling Policy of a Component Instance**. Each component can have a maximum of 20 labels. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment creation may fail. Contact your administrator to learn more about tag policies. 1. Click **Add Label**. 2. In the **Add Label** dialog box, click **Add Label**. 3. Enter a key and value. – The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). The key cannot be the built-in **app**, **casid**, or **version**. – The value contains 0 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 4. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Description	Component description. 1. Click and enter up to 128 characters. 2. Click to save the component description.

Step 5 Click **Next**.

Step 6 In the **Access Mode** area, enable **Public Network Access**.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If a component is upgraded and maintained in ELB dark launch mode after being created and deployed, you need to enable public network access for the component. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

Click to enable public access and set the following parameters:

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	▪ **Creating a Shared Load Balancer**. ▪ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

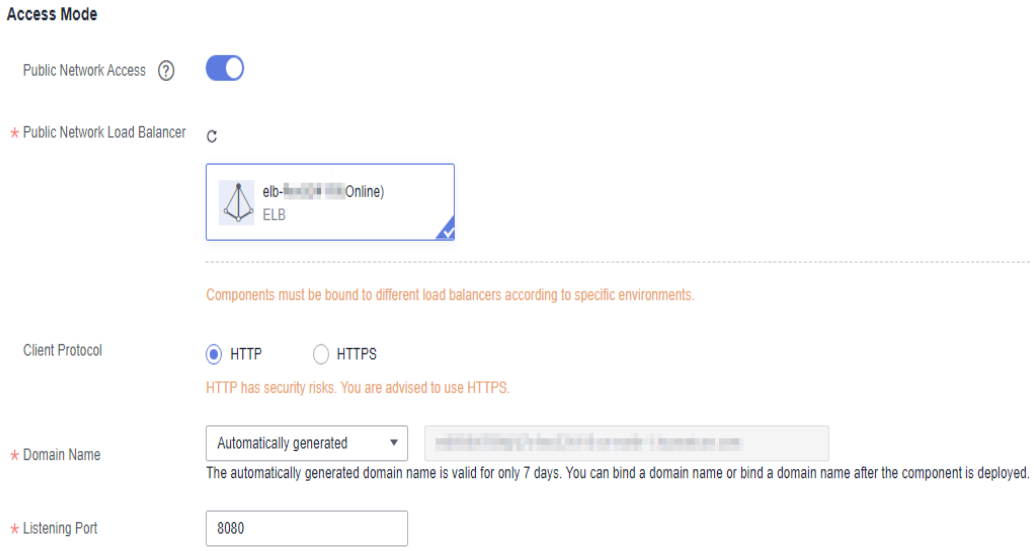
 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.
4. Set **Listening Port**.

   Enter the listening port number of the application process.

Figure 5-6 Configuring public access



Step 7 Set **Cloud Service Settings**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation Type	Operation	Description
Microservice Engine	Binding a microservice engine	**Cloud Service Engine** is a one-stop management platform for microservice solutions. After microservice components are connected to CSE, you can use service registry, service governance, and configuration management of CSE. In this way, you can focus on service development and improve product delivery efficiency and quality. Choose **Cloud Service Settings** > **Microservice Engine** and bind a microservice engine. For details, see **Binding a Microservice Engine**. • If multi-container deployment is enabled, the microservice engine cannot be bound. • To upgrade and maintain components in dark launch mode, bind the ServiceComb engine or registry/configuration center. For details, see **Upgrading a Component in Dark Launch Mode**.
Distributed Cache Service	Binding a distributed cache	In a distributed system, the distributed cache service provides scalable and reliable user session management. Choose **Cloud Service Settings** > **Distributed Cache Service** and bind a distributed cache. For details, see **Binding a Distributed Cache**.
Relational Database Service	Binding a cloud database	RDS is required for persistent storage of application data. Choose **Cloud Service Settings** > **Relational Database Service** and bind a cloud database. For details, see **Binding a Cloud Database**.
GaussDB	Binding a GaussDB instance	GaussDB is required for persistent storage of application data. Choose **Cloud Service Settings** > **GaussDB** and bind a GaussDB instance. For details, see **Binding a GaussDB Instance**.
Distributed Message Service for RocketMQ	Binding a DMS for RocketMQ instance	DMS for RocketMQ is required for application components to store and consume messages. Choose **Cloud Service Settings** > **Distributed Message Service for RocketMQ** and bind a RocketMQ instance. For details, see **Binding a DMS for RocketMQ Instance**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation Type	Operation	Description
Cloud Search Service	Binding a CSS instance	Uses the multi-condition search, statistics, and report capabilities of structured and unstructured text provided by the Elasticsearch cluster of Cloud Search Service (CSS) to analyze component logs in all scenarios. Choose **Cloud Service Settings** > **Cloud Search Service** and bind a CSS instance. For details, see **Binding a CSS Instance**.

Step 8 Click **Create Now**.

Perform subsequent operations as required by referring to the following table.

Operation	Procedure
Do not enable the component	• Click **Back to Component Management**. On the component list page, the imported component status is **Imported and to be restarted**. • To enable the component, see **Starting or Stopping a Component**.
Enable the component	1. Click **Enable Component Now**. 2. Click **Enable** in the **Operation** column. The component is successfully enabled and is in the **Running** state.

----End

Follow-up Operations

- **Starting or Stopping a Component** for **Imported and to be restarted** components.
- **Upgrading a Component in Single-batch Release Mode** and **Upgrading a Component in Rolling Release Mode** for **Running**, **Not ready**, **Abnormal**, and **Failed** components.
- **Upgrading a Component in Dark Launch Mode** for **Running**, **Not ready**, and **Abnormal** components.
- **Upgrading Components in Batches** for **Running**, **Not ready**, and **Abnormal** components.
- **Cloning Components in Batches** for **Running**, **Not ready**, and **Abnormal** components.
- **Configuring a Scaling Policy of a Component Instance** for **Running** and **Not ready** components.
- Maintain components by referring to **Component O&M** after the components are started.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For **Running** components, you can **Configuring the Component Access Mode** to access the services provided by the component in a preset mode.

5.2.5 Creating a Component Using a Component Template

ServiceStage supports a maximum of 15,000 components in the same project.

ServiceStage provides the capability of automatically creating components using component templates. You can configure components and create Kubernetes resources when creating components using a created component template.

Component Template Description

The component template supports only the YAML format. The created template file must be compressed into a ZIP package. Example structure of the package directory:

```
Component/
├── spec.yaml
├── variables.yaml
├── values.yaml
├── component/
│   ├── demo_component.yaml
│   ├── demo_config.yaml
│   ├── deployment.yaml
│   ├── configmap.yaml
│   ├── service.yaml
└── └── ...(Other Kubernetes resource files)
```

Kubernetes resource files take effect only for container-deployed components.

- **spec.yaml**: configuration information file, including the model version and Kubernetes resource creation sequence. Example file content:

```
# Model version number. Currently, only v1.0.0 is available.
model_version: v1.0.0

# Component list.
component:
  component_dir: comp_demo/

# Kubernetes resource object.
k8s_resources:
# Type: SYSTEM (default sequence) or CUSTOM.
  order_type: CUSTOM
# Sequence of the CUSTOM type.
  custom_orders:
    - ConfigMap
    - Deployment
    - Service
```

- **variables.yaml**: supported variable declaration file. All variables are declared in this file, including the variable name, type, description, and default value. Variables of the following types are supported: common types (string, integer, double, and bool), complex types (list(<T>), map(<T>), object(<T>)), ServiceStage model types (config, whose value is the group name or configuration name; runtimestack, whose value is the technology stack ID), and environment resource types (cce, ecs, customServer, elb, rds, dcs, cse, nacos, gaussdb, and dms_rocketmq (you can select multiple values and separate them with commas (,)). Example file content:

```
variable:
  namespace:
    type: string
    default: demo
```

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

```
  description: namespace
dep_name:
  type: string
  default: demo-dep
  description: workload name
svc_port:
  type: integer
  default: 8080
  description: service port number
value:
  type: string
  default: World
  description: Hello interface response data
cce_resource:
  type: cce
  description: CCE resource
image:
  type: string
  description: workload image
labels:
  type: list(object({key = string, value = string}))
  default: [{"key":"key1", "value":"value1"}]
  description: component label
resources:
  type: object({requests = object({cpu = string, memory = string}), limits = object({cpu = string,
memory = string})})
  default: {"limits": {"cpu":"100m", "memory":"256Mi"}}
  description: container resource application configuration
map_variable:
  type: map(list(object({name = string, age = integer})))
  default: {"persons": [{"name":"tom", "age": 20}], "animals": [{"name": "jack", "age": 18}]}
  description: combined data of the map, list, and object types
```

- **values.yaml**: variable value description file. Example file content:

```
dep_name: demo-dep
svc_port: 8080
comp_cpu_request: 0.5
comp_config: test-group/test.yaml
comp_elb: elb-id-xxx
comp_namespace: cce-id-xxx/default
```

- **component**: component directory. The directory name can be customized and contains the following files:
  - **demo_component.yaml**: component file. Only one component file can exist. For details, see **demo_component.yaml**.
  - **demo_config.yaml**: configuration file. A maximum of six configuration files are allowed. For details, see **config_demo.yaml**.
  - **cce_xxx.yaml**: workload type limit in Kubernetes resources. A maximum of 50 Kubernetes resources are allowed. There is only one Deployment, StatefulSet, DaemonSet, Pod, Job and CronJob resources, and the corresponding component is generated.
- **demo_component.yaml**: component file, which is used to describe component names and resources. Example file content:

```
kind: ServiceStageComponent
spec:
  name: ${var.comp_name}
  description: description
  # Either the environment ID or name must be specified. Query by ID is preferred.
  environment_id: ${var.env_id}
  environment_name: ${var.env_name}
  # Either the application ID or name must be specified. Query by ID is preferred.
  application_id: ${var.app_id}
  application_name: ${var.app_name}
  # Version is optional. If it is not specified, the system automatically generates one.
  version: 2024.0914.14111
```

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

```
 # Query by runtime_stack_id is preferred. If the value is empty, query by runtime_stack is used. The
default value of runtime_stack is Docker.
 runtime_stack_id: ${var.runtime_id}
 runtime_stack:
   name: "OpenJDK8"
   version: "1.3.8"
   type: "Java"
   deploy_mode: "virtualmachine"

 source:
   kind: package
   storage: obs
   url: "obs://obs-lxz/unit-provider-1.0-SNAPSHOT.jar"
   http_username: null
   http_password: null
   http_command: null
   http_headers:
     - key: key
       value: value
 jvm_opts: "-Xms256m"

 # TomcatOptsV3 model
 tomcat_opts:
   server_xml: |
     <?xml version="1.0" encoding="UTF-8"?>
     <Server port="-1" shutdown="SHUTDOWN">
     </Server>

 # TongWebOptsV3 model
 tongweb_opts:
   tongweb_xml: |
     <?xml version="1.0" encoding="UTF-8"?>
     <Server port="-1" shutdown="SHUTDOWN">
     </Server>
   external_vmoptions: "-Xms256m"

 # Environment variables
 envs:
   - name: key1
     value: value1
   - name: key2
     value: value2

 deploy_strategy:
   type: RollingRelease
   rolling_release:
     fail_strategy: STOP
     batches: 1
     termination_period_seconds: 30

 update_strategy:
   type: RollingUpdate
   termination_period_seconds: 30
   max_unavailable: 2
 refer_resources:
   - type: dcs
     parameters:
       password: test-xxx
     // refer_alias can be left empty and is automatically set to distributed_session,distributed_cache.
     refer_alias: distributed_session,distributed_cache
   - type: rds
     parameters:
       connection_type: JNDI
       jndi_name:
       db_name: demo
       db_user: root
       password: test-xxx
   - id: ${env.resources.elb.elb001.id}
     type: elb
```

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

```
    parameters:
      // Configure cert for HTTPS access.
      cert:
        id: xxxx
  - type: rds
    id: ${var.rds_id}
  - type: cce
    id: ${var.cce_id}
configs:
  # config_name: reference configuration. The naming rule is as follows: group name/configuration name. If there is only one container, container_name can be left blank. If there are multiple containers, set container_name based on the container name.
  - config_name: group/filename.yaml
    mount_path: /teast.yaml
    container_name: default
  - config_name: ${var.config}
    mount_path: /teast1.yaml
    container_name: default
external_accesses:
  - protocol: http
    forward_port: 8080
    address: demo.com
enable_sermant_injection: false
enable_sermant_spring_register: false
enable_sermant_dubbo_register: false
enable_sermant_config: false
```

Component resources can be referenced in the following modes:

- Specify only the resource type. The system automatically binds resources of this type in the environment. An error is reported when more than one resource meets the conditions.
- Specify the logical ID of a resource. For example, **${env.resources.elb.elb001.id}** matches the load balancer whose logical ID is elb001 in the environment.
- Specify the physical resource ID, for example, **${var.cce_id}**. **cce_id** is a variable of the cce type and its value is the CCE cluster ID.

- **config_demo.yaml**: configuration file, which describes the component configuration. Example file content:

```
kind: ServiceStageConfig
spec:
  name: test-group/test.yaml
  description: desc
  sensitive: true
  content: |
    test: world
    hello: 12313
```

Prerequisites

- An application has been created because components can be created only under an application. For details, see **Creating an Application**.
- An environment has been created based on the component deployment mode and resources have been managed because components need to be deployed in an environment.
  - For the environment types supported by each mode, see **Component Deployment Modes**.
  - For details about how to create an environment and manage resources, see **Environment Management**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- You have compressed the prepared component template files into a ZIP package and uploaded the package to the repository. For details, see **Component Template Description**.
  - Upload the template package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Select the corresponding template package from OBS. Upload the template package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**.
  - If you need to use JFrog (example) as the repository for storing the component template package, you can use an HTTP/HTTPS custom file download address as the component template package download address. Upload the component template package to the custom file address in advance.

  If the software package fails to be uploaded, see **What If a Software Package Fails to Be Uploaded?**

Creating a Component Using a Component Template

Step 1 Log in to ServiceStage.

Step 2 Use any of the following methods to go to the **Create Component** page.

- Choose **Component Management** > **Create Component** > **Use Template**.
- On the **Application Management** page, select the application for which you want to create a component, and click **More** > **Use Template** in the **Operation** column.
- On the **Application Management** page, click the application for which you want to create a component. On the displayed **Overview** page, click **Create Component** > **Use Template**.

Step 3 In the **Basic Information** area, configure the component by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Application	Application to which the component belongs.
*Environment	Component deployment environment.

Step 4 In the **Set Template Package** area, select the uploaded component template package based on the template package storage mode.

- **CodeArts release repository**: Click **Select Software Package**, select the uploaded template package from the CodeArts release repository, and click **OK**.
- **OBS**: Click **Select Software Package**, select the uploaded template package from the OBS bucket, and click **OK**.
- **SWR repository**: Click **Select Software Package**, select the uploaded template package from the SWR repository, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Custom file address**: If you need to use JFrog (example) as the repository for storing the template package, perform the following operations to obtain the template package that has been uploaded to the custom file address:

  a. Enter the HTTP/HTTPS custom file download address where the software package is located. The custom file address must be the complete path of the component template package file, for example, **https://172.16.0.1:8082/artifactory/template-component.zip**.

  b. Determine whether to enable authentication.

     If authentication is disabled, any user can download the software package in the custom file address by default.

     Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

Step 5 Click **Next**.

- In the **Preview Template Package** area, you can view the details of the component template file.
- In the **Set Parameters** area, you can view and modify the component parameters.

Step 6 Click **Create and Deploy**.

- During component deployment, if the component configuration is incorrect, you can click **Suspend** to stop the deployment. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Component Using a Component Template**.
- On the **View Deployment History** page, view the deployment logs and wait until the component deployment is complete.

----End

Follow-up Operations

- After a component created using a component template is deployed, the component can be upgraded only using the template. For details, see **Upgrading a Component Using a Component Template**.
- After a component created using a component template is deployed, maintain the component by referring to **Component O&M**.

5.3 Upgrading a Single Component

5.3.1 Upgrading a Component in Single-batch Release Mode

After a component is created and deployed, you can upgrade all instances of a component at a time in single-batch release mode.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

For details about how to upgrade multiple component versions of the same application in batches, see **Upgrading Components in Batches**.

Restrictions

During the upgrade, component services will be interrupted. This is applicable only to the test upgrade scenario or the upgrade scenario where services are to be stopped.

Prerequisites

- Only the components that are created and deployed based on a container using the following methods and whose **Workload Type** is **Stateless** or **DaemonSet** can be upgraded in single-batch release mode:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component by Importing a CCE Workload**
  - **Creating a Component Using a Component Template**
- The component status is **Running**, **Not ready**, **Abnormal** or **Failed**. For details about how to check the component status, see **Viewing Component Details**.

Upgrading a Component in Single-batch Release Mode

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 Go to the **Upgrade Component** page based on whether the component is created by **Creating a Component Using a Component Template**.

- If yes, choose **Upgrade** > **Upgrade** in the upper right corner of the page.
- If no, click **Upgrade** in the upper right corner of the page.

Step 4 Select **Single-batch Release** for **Type**.

Step 5 Click **Next** and set the component version configuration information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
Stack	Select a component technology stack and version. For details, see **Technology Stack**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*YAML Mode	Whether to use YAML configurations to upgrade components. • Disabled: The UI configurations are used to upgrade components. • Enabled: The YAML configurations are used to upgrade components. The latest load information of the component is automatically synchronized from CCE where the target component is deployed for component upgrade after modification. Alternatively, click **Import YAML File** to import the edited YAML configuration file of the target component. If YAML configurations are used to upgrade components, the parameters in the YAML configuration file are described in **Deployment**.
*Software Package/ Image	The value is fixed to the package type selected during component creation and deployment. • **YAML Mode** disabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package or image package, the option is fixed to the software package type (JAR, WAR, or ZIP) or image package type selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**. • **YAML Mode** enabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package, the option is fixed to the software package type (JAR, WAR, or ZIP) selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	• **YAML Mode** disabled: If the component source is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. If you select an image package: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – If **Multi-container** is selected in **Creating and Deploying a Component**, click **Select Image Package** to add a container and select an image package source. – If **Multi-container** is selected in **Creating and Deploying a Component**, choose > **Modify Image Package** in the upper right corner of the selected container card to modify the image package source of the container. – If **Multi-container** is selected in **Creating and Deploying a Component**, choose > **Delete** in the upper right corner of the selected container card to delete the container. – If **Multi-container** is selected in **Creating and Deploying a Component**, deselect **Multi-container** to deploy a component based a single container. – If **Multi-container** is not selected in **Creating and Deploying a Component**, select **Multi-container** to deploy a component based on multiple containers. Click **Select Image Package** to add a container, select an image package source, and set **Container Name**. – To customize a container name, enter 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and start with a lowercase letter and end with a lowercase letter or digit. • **YAML Mode** enabled: If the component source is software package, select an uploaded software package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.
Image Access Credential	An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.
*Command	This parameter is mandatory when **YAML Mode** is disabled, the package type is **Source code repository**, the component is deployed based on a container, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. For example: **cd ./weather/** **mvn clean package**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Dockerfile Address	This parameter is mandatory when **YAML Mode** is disabled, the package type is **Source code repository**, the component is deployed based on a container, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Component Version	Component version number, which can be automatically generated or customized. ● Automatically-generated: Click **Generate**. By default, the version number is the timestamp when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. ● Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.
*Container Name	This parameter is mandatory when the technology stack is not Docker and **YAML Mode** is disabled. The name contains 1 to 63 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only lowercase letters, digits, and hyphens (-) are allowed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Resources	This parameter is available when **YAML Mode** is disabled. You can customize **CPU** and **Memory** to set their quota, and set the maximum/minimum number of CPU cores and memory size (GiB) that can be used by components. In this way, you can set the number of resources required by each component instance. When resource **Request** is specified for a component instance, CCE uses this information to determine the node to which the component instance is scheduled. When resource **Limit** is specified for a component instance, CCE ensures that the resources occupied by the running component instance do not exceed the limit. CCE will also reserve the requested number of system resources for the component instance. For details about how to configure them, see **Resource Management**. To modify them, select the item to be changed and enter a new value. Unselected parameters indicate no restriction. If the package type selected for **Upload Method** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.
JVM Parameters	This parameter is available when **YAML Mode** is disabled and the technology stack type is Java or Tomcat. It configures the memory size during Java code running. Enter the JVM parameter. For example: `-Xms256m -Xmx1024m` Multiple parameters are separated by spaces.
Tomcat	This parameter is available when **YAML Mode** is disabled and the technology stack type is Tomcat. It configures parameters such as the request path and port number of Tomcat. 1. Select **Tomcat**. The **Tomcat** dialog box is displayed. 2. Click **Use Sample Code** and edit the template file based on service requirements. In Tomcat configuration, the default **server.xml** configuration is used. The context path is **/**, and no application path is specified. If you need to customize an application path, customize the Tomcat context path by referring to **How Do I Customize a Tomcat Context Path?** 3. Click **OK**.
Cloud Service Settings	If **YAML Mode** is disabled, set this parameter as required by referring to **Managing Cloud Service Settings of a Component**.
Container Settings	If **YAML Mode** is disabled, set this parameter as required by referring to **Managing Container Settings of a Container-Deployed Component**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Application Settings	If **YAML Mode** is disabled, set this parameter as required by referring to **Managing Application Settings of a Container-Deployed Component**. If the package type selected for **Upload Method** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.
Log Settings	If **YAML Mode** is enabled, set this parameter as required by referring to **Step 14**.
Advanced Settings	If **YAML Mode** is disabled, set this parameter as required by referring to **Managing Advanced Settings of a Container-Deployed Component**.

Step 6 Click **Upgrade**.

- During component upgrade, if the component configuration is incorrect, you can click **Suspend** to stop the upgrade. Then, click **Upgrade** and upgrade the component by referring to **Upgrading a Single Component**.
- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been upgraded.
- On the **View Deployment History** page, view the deployment logs.

  If "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

----End

Follow-up Operations

Operation	Description
Redeploy a component	You can select a historical version configuration from the **View Deployment History** list and use the version configuration as a template to redeploy components. For details, see **Redeploying a Component**.
Roll back a component	You can roll back a component from the latest version to the version before the upgrade or redeployment. For details, see **Rolling Back a Component**.

5.3.2 Upgrading a Component in Rolling Release Mode

After a component is created and deployed, you can upgrade a component in rolling release mode.

In rolling release mode, only one or more instances are upgraded at a time and then added to the production environment. This process repeats until all old instances are upgraded. Services will not be interrupted during the upgrade.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

For details about how to upgrade multiple component versions of the same application in batches, see **Upgrading Components in Batches**.

Prerequisites

- Only the components created and deployed using the following methods can be upgraded in rolling release mode:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component Based on a VM**
  - **Creating a Component by Importing a CCE Workload**
  - **Creating a Component Using a Component Template**
- For container-deployed components, the workload type must be **Stateless**, **Stateful**, or **DaemonSet**.
- If a component is deployed based on VM and bound to a ServiceComb engine, the following conditions must be met to ensure graceful application offline during rolling release:
  a. If the microservice component is developed using ServiceComb 2.7.8 or Spring Cloud earlier than Huawei 1.10.4-2021.0.x, add a label whose **Key** is **microservicename** and **Value** is the corresponding microservice instance name for the component. For details, see **Adding a Component Label**.
  b. If security authentication is enabled for the ServiceComb engine bound to the component:
     - For ServiceComb engine 2.5.0 or later, perform **Disabling Security Authentication** or **Importing an IAM Account**.
     - For ServiceComb engine earlier than 2.5.0, perform **Disabling Security Authentication** or **Upgrading a Microservice Engine Version** and then **Importing an IAM Account**.
- The component status is **Running**, **Not ready**, **Abnormal** or **Failed**. For details about how to check the component status, see **Viewing Component Details**.

Upgrading a Component in Rolling Release Mode

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 Go to the **Upgrade Component** page based on whether the component is created by **Creating a Component Using a Component Template**.

- If yes, choose **Upgrade** > **Upgrade** in the upper right corner of the page.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If no, click **Upgrade** in the upper right corner of the page.

Step 4 Select **Rolling Release** for **Type**.

Step 5 Click **Next** and set the component version configuration information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
Stack	Select a component technology stack and version. For details, see **Technology Stack**.
*YAML Mode	For container-deployed components, you can choose whether to use YAML configurations to upgrade components. • Disabled: The UI configurations are used to upgrade components. • Enabled: The YAML configurations are used to upgrade components. The latest load information of the component is automatically synchronized from CCE where the target component is deployed for component upgrade after modification. Alternatively, click **Import YAML File** to import the edited YAML configuration file of the target component. If YAML configurations are used to upgrade components, the parameters in the YAML configuration file are described in **Deployment**.
*Software Package/ Image	The value is fixed to the package type selected during component creation and deployment. • For container-deployed components with **YAML Mode** disabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package or image package, the option is fixed to the software package type (JAR, WAR, or ZIP) or image package type selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**. • For container-deployed components with **YAML Mode** enabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package, the option is fixed to the software package type (JAR, WAR, or ZIP) selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**. • For VM-deployed components: If the component source is software package, the value is fixed to the software package type (JAR, WAR, ZIP, or compressed package) selected during component creation. It is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	• For container-deployed components with **YAML Mode** disabled: If the component source is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. If you select an image package: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – If **Multi-container** is selected in **Creating and Deploying a Component**, click **Select Image Package** to add a container and select an image package source. – If **Multi-container** is selected in **Creating and Deploying a Component**, choose > **Modify Image Package** in the upper right corner of the selected container card to modify the image package source of the container. – If **Multi-container** is selected in **Creating and Deploying a Component**, choose > **Delete** in the upper right corner of the selected container card to delete the container. – If **Multi-container** is selected in **Creating and Deploying a Component**, deselect **Multi-container** to deploy a component based a single container. – If **Multi-container** is not selected in **Creating and Deploying a Component**, select **Multi-container** to deploy a component based on multiple containers. Click **Select Image Package** to add a container, select an image package source, and set **Container Name**. – To customize a container name, enter 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and start with a lowercase letter and end with a lowercase letter or digit. • For container-deployed components with **YAML Mode** enabled: If the component source is software package, select an uploaded software package. For details about the upload method, see **Component Source**. • For VM-deployed components: If the component source is software package, you can select an uploaded software package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.
Image Access Credential	This parameter is available for container-deployed components. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.
*Command	This parameter is mandatory when the component is deployed based on a container, **YAML Mode** is disabled, the package type is **Source code repository**, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. For example: **cd ./weather/** **mvn clean package**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Dockerfile Address	This parameter is mandatory when the component is deployed based on a container, **YAML Mode** is disabled, the package type is **Source code repository**, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Component Version	Component version number, which can be automatically generated or customized. ● Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. ● Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.
*Container Name	This parameter is mandatory when the component is deployed based on a container, the technology stack is not Docker, and **YAML Mode** is disabled. The name contains 1 to 63 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only lowercase letters, digits, and hyphens (-) are allowed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Resources	This parameter is available when the component is deployed based on a container and **YAML Mode** is disabled. You can customize **CPU** and **Memory** to set their quota, and set the maximum/minimum number of CPU cores and memory size (GiB) that can be used by components. In this way, you can set the number of resources required by each component instance. When resource **Request** is specified for a component instance, CCE uses this information to determine the node to which the component instance is scheduled. When resource **Limit** is specified for a component instance, CCE ensures that the resources occupied by the running component instance do not exceed the limit. CCE will also reserve the requested number of system resources for the component instance. For details about how to configure them, see **Resource Management**. To modify them, select the item to be changed and enter a new value. Unselected parameters indicate no restriction. If the package type selected for **Upload Method** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.
JVM Parameters	This parameter is available when the component is deployed based on a container with **YAML Mode** disabled or based on VM, and the technology stack type is Java or Tomcat. It configures the memory size during Java code running. Enter the JVM parameter. For example: `-Xms256m -Xmx1024m` Multiple parameters are separated by spaces.
Tomcat	This parameter is available when the component is deployed based on a container with **YAML Mode** disabled or based on VM, and the technology stack type is Tomcat. It configures parameters such as the request path and port number of Tomcat. 1. Select **Tomcat**. The **Tomcat** dialog box is displayed. 2. Click **Use Sample Code** and edit the template file based on service requirements. In Tomcat configuration, the default **server.xml** configuration is used. The context path is **/**, and no application path is specified. If you need to customize an application path, customize the Tomcat context path by referring to **How Do I Customize a Tomcat Context Path?** 3. Click **OK**.
Cloud Service Settings	• If the component is deployed based on a container and **YAML Mode** is disabled, set this parameter as required by referring to **Managing Cloud Service Settings of a Component**. • If the component is deployed based on VM, set this parameter as required by referring to **Managing Cloud Service Settings of a Component**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Container Settings	If the component is deployed based on a container and **YAML Mode** is disabled, set this parameter as required by referring to **Managing Container Settings of a Container-Deployed Component**.
VM Settings	If the component is deployed based on VM, set this parameter as required by referring to **Configuring VM Settings**.
Application Settings	• If the component is deployed based on a container, set this parameter as required by referring to **Managing Application Settings of a Container-Deployed Component**. If the package type selected for **Upload Method** is image package and multi-container deployment is enabled, set this parameter for each container instance as required. • If the component is deployed based on VM, set this parameter as required by referring to **Managing Application Settings of a VM-Deployed Component**.
Advanced Settings	• If the component is deployed based on a container and **YAML Mode** is disabled, set this parameter as required by referring to **Managing Advanced Settings of a Container-Deployed Component**. • If the component is deployed based on VM, set this parameter as required by referring to **Configuring Application Performance Management of a VM-Deployed Component**.
Log Settings	• If the component is deployed based on a VM and **YAML Mode** is enabled, set this parameter as required by referring to **Step 14**. • If the component is deployed on an ECS based on a VM, set this parameter as required by referring to **Configuring Log Settings of a VM-Deployed Component**.
*Deployment Batch	This parameter is available when the component is deployed based on VM. Number of batches in which component instances are upgraded. The value range is [1, Total number of instances]. Total number of instances refers to the number of running instances of the component. For components deployed based on VM, up to 40 component instances can be deployed in each batch. For example, if there are 4 component instances and **Deployment Batch** is set to **2**, these component instances are upgraded in two batches, and each batch involves two component instances.
*Pause Before Stop	This parameter is available when the component is deployed based on VM. The Rx traffic will be processed during this period of time. The value ranges from 0 to 600, in seconds. The default value is **30**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Best Effort Mode	This parameter is available when the component is deployed based on VM. • Disabled: When a component instance fails to be upgraded, the upgrade of the remaining instances will be stopped. • Enabled: If the upgrade fails, the remaining instances will be upgraded. Exercise caution when enabling this mode. After this mode is enabled, all instances may fail to be upgraded in extreme cases, affecting service functions.

Step 6 Click **Upgrade**.

- During component upgrade, if the component configuration is incorrect, you can click **Suspend** to stop the upgrade. Then, click **Upgrade** and upgrade the component by referring to **Upgrading a Single Component**.
- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been upgraded.
- On the **View Deployment History** page, view the deployment logs.

  For container-deployed components, if "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

  For VM-deployed components, if "Creation of the VM application instance failure", "Query of the VM task status failure", or "Query of the VM application instance status failure" is displayed, click **View Event** to view details in the event list.

----End

Follow-up Operations

Operation	Description
Redeploy a component	You can select a historical version configuration from the **View Deployment History** list and use the version configuration as a template to redeploy components. For details, see **Redeploying a Component**.
Rolling back a component	You can roll back a component from the latest version to the version before the upgrade or redeployment. For details, see **Rolling Back a Component**.

5.3.3 Upgrading a Component in Dark Launch Mode

After a component is created and deployed, you can upgrade a component in ServiceStage dark launch mode.

For details about how to upgrade multiple component versions of the same application in batches, see **Upgrading Components in Batches**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Introduction to Dark Launch

In dark launch mode, a certain proportion of instances are upgraded, and traffic is directed to the new version to verify whether functions of the new version are normal. Then, the remaining instances will be upgraded in rolling mode. Dark launch ensures stability of the entire system. During initial dark launch, problems can be detected and fixed.

Table 5-4 Dark launch types and description

Type	Description
Microservice Dark Launch	Applies to container-deployed ServiceComb and Spring Cloud components. Dark launch tasks function on microservices. Multiple microservices can work together to roll out new features. 1. The Java, Tomcat, or Docker technology stack must be selected for the component. 2. The component must be bound to a ServiceComb engine with security authentication disabled and multi-language access to service mesh disabled. • For ServiceComb engines, Java chassis 2.7.8 or later and Spring Cloud Huawei 1.10.4-2021.0.x or later are required. • For registry/configuration centers, Spring Cloud Huawei version must be 1.11.9-2021.0.x, 1.11.9-2022.0.x, 1.11.9-2023.0.x, or later.
ELB Dark Launch	Applies to ELB traffic-based and container-deployed components. Dark launch tasks function on ELB. • The public network access of the component must be enabled when **Creating and Deploying a Component**. • Use a dedicated application load balancer if you select a content-based dark launch policy. For details, see **Creating a Dedicated Load Balancer**.

Restrictions

During the dark launch upgrade in this operation, do not perform dark launch of the component bound to a ServiceComb engine or through registry/configuration enter. Otherwise, this operation will fail.

- For details about how to perform dark launch of a component through ServiceComb engines, see **Dark Launch**.
- For details about how to perform dark launch of a component through registry/configuration centers, see **Configuring Dark Launch**.

Prerequisites

- A CCE cluster must be bound to the component deployment environment.
- Only the components that are created and deployed based on a container using the following methods and whose **Workload Type** is **Stateless** (if the

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

component technology stack is Docker, multi-container deployment must be disabled) can be upgraded in dark launch mode:

- **Creating a Component Based on a Container Using Manual Configurations**
- **Creating a Component Based on a Container Using YAML Configurations**
- **Creating a Component by Importing a CCE Workload**
- **Creating a Component Using a Component Template**

- The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**.

Upgrading a Component in Dark Launch Mode

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 Go to the **Upgrade Component** page based on whether the component is created by **Creating a Component Using a Component Template**.

- If yes, choose **Upgrade** > **Upgrade** in the upper right corner of the page.
- If no, click **Upgrade** in the upper right corner of the page.

Step 4 Select **Dark Launch** for **Type**.

Step 5 Click **Next** and set the component version configuration information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
Stack	Select a component technology stack and version. For details, see **Technology Stack**.
*Software Package/ Image	The value is fixed to the package type selected during component creation and deployment. • If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. • If you select a software package or image package, the option is fixed to the software package type (JAR, WAR, or ZIP) or image package type selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	If the component source is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • If the package type is image package: – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – In the upper right corner of the container card, choose > **Modify Image Package** to modify the image package source of the container.
Image Access Credential	An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	This parameter is mandatory when the package type is **Source code repository** and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. For example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	This parameter is mandatory when the package type is **Source code repository** and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Component Version	Component version number, which can be automatically generated or customized. • Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Container Name	This parameter is mandatory when the technology stack is not Docker. The name contains 1 to 63 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only lowercase letters, digits, and hyphens (-) are allowed.
Resources	You can customize **CPU** and **Memory** to set their quota, and set the maximum/minimum number of CPU cores and memory size (GiB) that can be used by components. In this way, you can set the number of resources required by each component instance. When resource **Request** is specified for a component instance, CCE uses this information to determine the node to which the component instance is scheduled. When resource **Limit** is specified for a component instance, CCE ensures that the resources occupied by the running component instance do not exceed the limit. CCE will also reserve the requested number of system resources for the component instance. For details about how to configure them, see **Resource Management**. To modify them, select the item to be changed and enter a new value. Unselected parameters indicate no restriction.
JVM Parameters	This parameter is available when the technology stack type is Java or Tomcat. It configures the memory size during Java code running. Enter the JVM parameter. For example: `-Xms256m -Xmx1024m` Multiple parameters are separated by spaces.
Tomcat	This parameter is available when the technology stack type is Tomcat. It configures parameters such as the request path and port number of Tomcat. 1. Select **Tomcat**. The **Tomcat** dialog box is displayed. 2. Click **Use Sample Code** and edit the template file based on service requirements. In Tomcat configuration, the default **server.xml** configuration is used. The context path is **/**, and no application path is specified. If you need to customize an application path, customize the Tomcat context path by referring to **How Do I Customize a Tomcat Context Path?** 3. Click **OK**.
Container Settings	Set this parameter as required by referring to **Managing Container Settings of a Container-Deployed Component**.
Application Settings	Set this parameter as required by referring to **Managing Application Settings of a Container-Deployed Component**.
Advanced Settings	Set this parameter as required by referring to **Managing Advanced Settings of a Container-Deployed Component**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Dark Launch Traffic Control	• Enabled: Traffic is controlled during dark launch. • Disabled: Traffic is not controlled during dark launch. Only workloads are controlled.
*Deployment Architecture	Available when **Dark Launch Traffic Control** is enabled, indicating the traffic access mode. 1. Click **Select** and select an instance deployment architecture for dark lunch. 2. Click **OK**. – You must choose the correct architecture. Otherwise, dark launch will fail and services will be affected. – Click **Change** to reselect a deployment architecture.
*Dark Launch Policy	Available when **Dark Launch Traffic Control** is enabled. ELB dark launch supports only the **Traffic ratio-based** policies. • **Traffic ratio-based**: Flexibly and dynamically adjust the traffic ratio of different versions. • **Content-based**: Control the version to which the request is sent according to the request.
*Traffic Ratio	When **Dark Launch Policy** is set to **Traffic ratio-based**, set the traffic ratio. • **Traffic Ratio**: percentage of traffic directed to the new version. • **Current Traffic Ratio**: percentage of traffic directed to the current version.
*Takes effect	When **Dark Launch Policy** is set to **Content-based**, set the effective mode of the dark launch policy. • **With single criterion**: The dark launch policy takes effect when any matching rule is met. • **With full criteria**: The dark launch policy takes effect when all matching rules are met.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Matching Rule	When **Dark Launch Policy** is set to **Content-based**, set the matching rule for the dark launch policy to take effect. 1. Click **Add Matching Condition**. 2. Set **Matching Type**. Currently, only **Request header** is supported. 3. Set **Parameter Name**, that is, the key corresponding to **Matching Type**. 4. Set **Condition Type**, that is, the matching rule that **Condition Value** meets. – **Equal**: The value corresponding to **Matching Type** is equal to the configured **Condition Value**. – **Matching**: **Condition Value** supports PERL-compatible regular expressions. It is supported by only microservice dark launch. – **Enumerated**: Multiple **Condition Value** items are separated by commas (,). 5. Set **Condition Value**, that is, the value corresponding to **Matching Type**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Instances Deployed for Dark Launch	Select a mode for adding an instance during dark launch. To change the number of component instances to meet the requirements, see **Configuring a Manual Scaling Policy**. • Blue-green Add new version instances with the same number as the old version instances, and switch traffic to the new version based on dark launch rules. After verification based on the dark launch rules, switch all traffic to the new version and delete the old version. Scenarios: lossless service capacity, fast rollback, and when resources are needed in the cluster. This mode requires one or more component instances. • Canary (increase, then decrease) Add new version instances with the specified number, and switch traffic to the new version based on dark launch rules. After verification based on the dark launch rules, delete some old instances, add corresponding new instances, and switch traffic. Repeat this process until all traffic is switched to the new instances and old instances are deleted. Scenarios: lossless service capacity, slow rollback, and when resources are needed in the cluster. This mode requires two or more component instances. • Canary (decrease, then increase) Delete some old instances and add corresponding new instances, and switch traffic to the new version based on dark launch rules. After verification based on the dark launch rules, delete some old instances, add corresponding new instances, and switch traffic. Repeat this process until all traffic is switched to the new instances and old instances are deleted. Scenarios: lossy service capacity and slow rollback. This mode requires two or more component instances.
*First Batch of Dark Launch Instances	Set this parameter when **Instances Deployed for Dark Launch** is set to **Canary (increase, then decrease)** or **Canary (decrease, then increase)**. The value range is [1, Total number of instances – 1]. Total number of instances refers to the number of running instances of the component. For example, if there are 6 component instances and **First Batch of Dark Launch Instances** is set to **1**, 1 instance will be upgraded in the first batch.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Deployment Batch with Remaining Instances	Set this parameter when **Instances Deployed for Dark Launch** is set to **Canary (increase, then decrease)** or **Canary (decrease, then increase)**. That is, the number of batches whose remaining instances will be upgraded. For example, if there are 6 component instances, **First Batch of Dark Launch Instances** is set to **1**, and **Deployment Batch with Remaining Instances** is set to **3**, there are 5 instances remaining to be deployed in 3 batches, and these 5 instances will be upgraded in the sequence 2:2:1.
New Workload Name	• **Default**: After dark launch is complete, the new workload name starts with **new-** by default. • **Custom**: After dark launch is complete, the new workload name is the name you set. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).

Step 6 Click **Upgrade**.

- During component upgrade, if the component configuration is incorrect, you can click **Suspend** to stop the upgrade. Then, click **Upgrade** and upgrade the component by referring to **Upgrading a Single Component**.
- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been upgraded.
- On the **Deployment Records** page, view the deployment logs.

  For container-deployed components, if "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

----End

Follow-up Operations

Operation	Description
View the deployment logs	Select a deployment record in the **Deployment Records** list and view component deployment logs on the **Deployment Log** tab.
View system monitoring	If the component is upgraded through dark launch, choose **System Monitoring** to check the CPU and memory usage of instances of the dark launch version and the current version after the first batch of dark launch is complete.
Modify a dark launch policy	If the component is upgraded through dark launch, you can modify the dark launch policy by referring to **Modifying a Component Dark Launch Policy** after the first batch of dark launch is complete.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Complete processes	If the component is upgraded through dark launch after the first batch of dark launch is successful and the functions of the new version are normal, you can perform the following operations to upgrade the remaining component instances: 1. Select the deployment record of the **Dark Launch** type. 2. Click **Complete Processes**. 3. In the displayed dialog box, click **OK**. Upgrade the remaining instances to the new version.
Roll back a component	The component can be rolled back in the following scenarios: • The first batch of dark launch is complete if the component is upgraded through dark launch. • All component instances have been upgraded to the new version. To roll back the component configuration to the source version, refer to **Rolling Back a Component**.
Redeploy a component	After all component instances are upgraded to the new version, select a deployment record and use it as a template to deploy components again. For details, see **Redeploying a Component**.

Modifying a Component Dark Launch Policy

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **View Deployment History** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**. In the left navigation pane, choose **Deployment Records**.
- On the **Component Management** page, click the target component. In the left navigation pane, choose **Deployment Records**.

Step 3 In the **Deployment Records** list, select the target deployment record of the **Dark Launch** type.

Step 4 Click **Modify Dark Launch Policy** and modify the dark launch configuration by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Dark Launch Policy	Select a dark lunch policy. • **Traffic ratio-based**: Flexibly and dynamically adjust the traffic ratio of different versions. • **Content-based**: Control the version to which the request is sent according to the request.
*Traffic Ratio	When **Dark Launch Policy** is set to **Traffic ratio-based**, set the traffic ratio. • **Traffic Ratio**: percentage of traffic directed to the new version. • **Current Traffic Ratio**: percentage of traffic directed to the current version.
*Takes effect	When **Dark Launch Policy** is set to **Content-based**, set the effective mode of the dark launch policy. • **With single criterion**: The dark launch policy takes effect when any matching rule is met. • **With full criteria**: The dark launch policy takes effect when all matching rules are met.
*Matching Rule	When **Dark Launch Policy** is set to **Content-based**, set the matching rule for the dark launch policy to take effect. 1. Click **Add Matching Condition**. 2. Set **Matching Type**. Currently, only **Request header** is supported. 3. Set **Parameter Name**, that is, the key corresponding to **Matching Type**. 4. Set **Condition Type**, that is, the matching rule that **Condition Value** meets. – **Equal**: The value corresponding to **Matching Type** is equal to the configured **Condition Value**. – **Matching**: **Condition Value** supports PERL-compatible regular expressions. It is supported by only microservice dark launch. – **Enumerated**: Multiple **Condition Value** items are separated by commas (,). 5. Set **Condition Value**, that is, the value corresponding to **Matching Type**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*First Batch of Dark Launch Instances	Set this parameter when **Instances Deployed for Dark Launch** is set to **Canary (increase, then decrease)** or **Canary (decrease, then increase)**. The value range is [1, Total number of instances – 1]. Total number of instances refers to the number of running instances of the component. For example, if there are 6 component instances and **First Batch of Dark Launch Instances** is set to **1**, 1 instance will be upgraded in the first batch.
*Deployment Batch with Remaining Instances	Set this parameter when **Instances Deployed for Dark Launch** is set to **Canary (increase, then decrease)** or **Canary (decrease, then increase)**. That is, the number of batches whose remaining instances will be upgraded. For example, if there are 6 component instances, **First Batch of Dark Launch Instances** is set to **1**, and **Deployment Batch with Remaining Instances** is set to **3**, there are 5 instances remaining to be deployed in 3 batches, and these 5 instances will be upgraded in the sequence 2:2:1.

Step 5 Click **Modify Dark Launch Policy** and wait until the dark launch is complete.

----End

5.3.4 Upgrading a Component Using a Component Template

Upgrade the version configurations of the component created by **Creating a Component Using a Component Template**.

Prerequisites

- Only components created using a component template support version upgrade using a component template. For details, see **Creating a Component Using a Component Template**.
- The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**.
- You have compressed the prepared component template files into a ZIP package and uploaded the package to the repository. For details, see **Component Template Description**.
  - Upload the template package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Select the corresponding template package from OBS. Upload the template package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**.
  - If you need to use JFrog (example) as the repository for storing the component template package, you can use an HTTP/HTTPS custom file download address as the component template package download

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

address. Upload the component template package to the custom file address in advance.

Upgrading a Component Created Using a Component Template

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page:

- On the **Application Management** page, click the application to which the target component belongs, and click the component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 Choose **Upgrade** > **Use Template** in the upper right corner of the page.

Step 4 In the **Set Template Package** area, select the uploaded component template package based on the template package storage mode.

- CodeArts release repository
  - If CodeArts release repository is not the component template repository, click **Select Software Package**, select the uploaded template package from the CodeArts release repository, and click **OK**.
  - Otherwise, click , select the uploaded template package from the CodeArts release repository, and click **OK**. Alternatively, click > **Select Software Package**, select the uploaded template package from the CodeArts release repository, and click **OK**.
- OBS
  - If OBS is not the component template repository, click **Select Software Package**, select the uploaded template package from the OBS bucket, and click **OK**.
  - Otherwise, click , select the uploaded template package from the OBS bucket, and click **OK**. Alternatively, click > **Select Software Package**, select the uploaded template package from the OBS bucket, and click **OK**.
- SWR repository
  - If SWR repository is not the component template repository, click **Select Software Package**, select the uploaded template package from the SWR repository, and click **OK**.
  - Otherwise, click , select the uploaded template package from the SWR repository, and click **OK**. Alternatively, click > **Select Software Package**, select the uploaded template package from the SWR repository, and click **OK**.
- **Custom file address**: If you need to use JFrog (example) as the repository for storing the template package, perform the following operations to obtain the template package that has been uploaded to the custom file address:
  a. Enter the HTTP/HTTPS custom file download address where the software package is located. The custom file address must be the complete path of the component template package file, for example, **https://172.16.0.1:8082/artifactory/template-component.zip**.
  b. Determine whether to enable authentication.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If authentication is disabled, any user can download the software package in the custom file address by default.

Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

Step 5 Click **Next**.

- In the **Preview Template Package** area, you can view the details of the component template file.
- In the **Set Parameters** area, you can view and modify the component parameters.

Step 6 Click **Upgrade**.

- During component deployment, if the component configuration is incorrect, you can click **Suspend** to stop the deployment. Then, click **Upgrade** and upgrade the component.
- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been upgraded.
- On the **Deployment Records** page, view the deployment logs.

  If "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

----End

5.4 Upgrading Components in Batches

After components are created and deployed, you can reconfigure and deploy multiple components of the same application in rolling release mode. In rolling release mode, only one or more instances are upgraded at a time and then added to the production environment. This process repeats until all old instances are upgraded. Services will not be interrupted during the upgrade.

For details about how to upgrade a single component, see **Upgrading a Single Component**.

Prerequisites

- Only the components created and deployed using the following methods support batch component version upgrade:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component Based on a VM**
  - **Creating a Component by Importing a CCE Workload**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- The component status is **Running**, **Not ready**, **Abnormal** or **Failed**. For details about how to check the component status, see **Viewing Component Details**.
- If a component is deployed based on a container and the selected technology stack is Docker, multi-container deployment must be disabled.
- For container-deployed components, you are advised to upgrade no more than 30 component instances in batches. Otherwise, CCE will limit the traffic and the upgrade will take a long time.

Upgrading Components in Batches

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Click the application where the target components are located. The **Overview** page of the application is displayed.

Step 4 Select the components to be upgraded in batches in **Component List** and click **More** > **Upgrade**.

Step 5 Set the version configuration information of the components to be upgraded by referring to the following table.

Parameter	Description
Component Version	Target version of the upgraded component. • By default, the version number is the time when you start to upgrade the component. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click and select another package type. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected image package to change the image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Deployment Batch	Number of batches in which component instances are upgraded. The value range is [1, Total number of instances]. Total number of instances refers to the number of running instances of the component. For example, if there are 4 component instances and **Deployment Batch** is set to **2**, these component instances are upgraded in two batches, and each batch involves two component instances. After setting this parameter for container-deployed components, you can reset it in **Step 6** in **Configuring a Component Upgrade Policy**.

Step 6 Click **Advanced Settings** in the **Operation** column to configure advanced settings.

- Select a component technology stack and version. For details, see **Technology Stack**.
- Set **Cloud Service Settings** by referring to **Managing Cloud Service Settings of a Component**.
- For container-deployed components, set other parameters by referring to **Managing Container Settings of a Container-Deployed Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components:
  - Set other parameters by referring to **Managing VM-Deployed Component Configurations**.
  - Select **Upgrade Policy** and set **Pause Before Stop**.

    The Rx traffic will be processed during this period of time. The unit is second and the value ranges from 0 to 600.

Step 7 (Optional) Click **Pre-check Advanced Settings** to check whether the advanced settings of each component are correct.

For example, if the component has been bound to a specified microservice engine, distributed cache, or cloud database instance but the specified instance is not bound to the environment to which the component belongs, an error is reported during advanced settings pre-check. Perform the following operations to reconfigure the parameters and then continue:

1. In the displayed dialog box, confirm the information and click **OK**.
2. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 6**.

Step 8 Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct.

- If no, reset the advanced settings of the component based on the error information by referring to **Step 6**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If yes, wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component version configuration is successfully upgraded. On the **Deployment Records** page, view the deployment logs.
  - For container-deployed components, if "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.
  - For VM-deployed components, if "Creation of the VM application instance failure", "Query of the VM task status failure", or "Query of the VM application instance status failure" is displayed, click **View Event** to view details in the event list.
  - During the component upgrade, if a component is incorrectly configured, you can click the component to go to the **Overview** page and click **Suspend** to stop the component upgrade. Then, click **Upgrade** and upgrade the component by referring to **Upgrading a Single Component**.

----End

Follow-up Operations

Operation	Description
Roll back a component	After the version configuration of all component instances is upgraded to the new version, if you need to roll back the component to the source version, see **Rolling Back a Component**.
Redeploy a component	You can select a historical version configuration from the deployment record list and use the version configuration as a template to redeploy components. For details, see **Redeploying a Component**.

5.5 Redeploying a Component

5.5.1 Redeploying a Component in Single-batch Release Mode

You can select a historical version configuration from the deployment record list and use the version configuration as a template to redeploy components in single-batch release mode.

During component redeployment in single-batch release mode, component services will be interrupted. This is applicable only to the test upgrade scenario or the upgrade scenario where services are to be stopped.

Prerequisites

- Only container-deployed components that have been upgraded can be redeployed in single-batch release mode.

  For details about how to upgrade components, see **Upgrading a Single Component** or **Upgrading Components in Batches**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- The component version configuration that has been rolled back by referring to **Rolling Back a Component** cannot be used as a template to redeploy the component.

Redeploying a Component in Single-batch Release Mode

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **View Deployment History** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Deployment Records** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Deployment Records** in the left navigation pane.

Step 3 In the **Deployment Records** list, select the deployment record of the historical version to be used as the configuration template.

Step 4 Click **Redeploy** in the upper right corner of the page. The **Redeploy** dialog box is displayed.

Step 5 Select **Single-batch Release** for **Deployment Type** and click **OK**.

Step 6 Configure the component version by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
Stack	Select a component technology stack and version. For details, see **Technology Stack**.
*YAML Mode	Whether to use YAML configurations to redeploy components. • Disabled: The UI configurations are used to redeploy components. • Enabled: The YAML configurations are used to redeploy components. The latest load information of the component is automatically synchronized from CCE where the target component is deployed for component redeployment after modification. Alternatively, click **Import YAML File** to import the edited YAML configuration file of the target component. The parameters in the YAML configuration file are described in **Deployment**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Software Package/ Image	The value is fixed to the package type selected during component creation and deployment. • **YAML Mode** disabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package or image package, the option is fixed to the software package type (JAR, WAR, or ZIP) or image package type selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**. • **YAML Mode** enabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package, the option is fixed to the software package type (JAR, WAR, or ZIP) selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	• **YAML Mode** disabled: If the component source is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. If you select an image package: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • **YAML Mode** enabled: If the component source is software package, select an uploaded software package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Image Access Credential	An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.
*Command	This parameter is mandatory when **YAML Mode** is disabled, the package type is **Source code repository**, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. For example: **cd ./weather/** **mvn clean package**
*Dockerfile Address	This parameter is mandatory when **YAML Mode** is disabled, the package type is **Source code repository**, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Component Version	Component version number, which can be automatically generated or customized. • Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.
*Container Name	This parameter is mandatory when the technology stack is not Docker. The name contains 1 to 63 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only lowercase letters, digits, and hyphens (-) are allowed.
Resources	When **YAML Mode** is disabled, the configuration of the selected historical version is used and cannot be modified.
JVM Parameters	When **YAML Mode** is disabled and the technology stack type is Java or Tomcat, the memory parameter size configured for running the Java code of the selected historical version is used and cannot be changed.
Tomcat	When **YAML Mode** is disabled and the technology stack type is Tomcat, the configuration, such as Tomcat request path and port number, of the selected historical version is used and cannot be changed.
Cloud Service Settings	When **YAML Mode** is disabled, the configuration of the selected historical version is used and cannot be modified.
Container Settings	When **YAML Mode** is disabled, the configuration of the selected historical version is used and cannot be modified.
Application Settings	When **YAML Mode** is disabled, the configuration of the selected historical version is used and cannot be modified.
Log Settings	When **YAML Mode** is enabled, the configuration of the selected historical version is used and cannot be modified.
Advanced Settings	When **YAML Mode** is disabled, the configuration of the selected historical version is used and cannot be modified.

Step 7 Click **Upgrade**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- During component redeployment, if the component configuration is incorrect, you can click **Suspend** to stop the redeployment. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Single Component**.
- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been redeployed.
- On the **View Deployment History** page, view the deployment logs.

  If "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

----End

5.5.2 Redeploying a Component in Rolling Release Mode

You can select a historical version configuration from the deployment record list and use the version configuration as a template to redeploy components in rolling release mode.

In rolling release mode, only one or more instances are deployed at a time and then added to the production environment. This process repeats until all old instances are upgraded. Services will not be interrupted during the deployment.

Prerequisites

- You have upgraded a component. For details, see **Upgrading a Single Component** or **Upgrading Components in Batches**.
- The component version configuration that has been rolled back by referring to **Rolling Back a Component** cannot be used as a template to redeploy the component.

Redeploying a Component in Rolling Release Mode

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **View Deployment History** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Deployment Records** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Deployment Records** in the left navigation pane.

Step 3 In the **Deployment Records** list, select the deployment record of the historical version to be used as the configuration template.

Step 4 Click **Redeploy** in the upper right corner of the page. The **Redeploy** dialog box is displayed.

Step 5 Select **Rolling Release** for **Deployment Type** and click **OK**.

Step 6 Configure the component version by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Stack	Select a component technology stack and version. For details, see **Technology Stack**.
*YAML Mode	For container-deployed components, you can choose whether to use YAML configurations to redeploy components. • Disabled: The UI configurations are used to redeploy components. • Enabled: The YAML configurations are used to redeploy components. The latest load information of the component is automatically synchronized from CCE where the target component is deployed for component redeployment after modification. Alternatively, click **Import YAML File** to import the edited YAML configuration file of the target component. The parameters in the YAML configuration file are described in **Deployment**.
*Software Package/ Image	The value is fixed to the package type selected during component creation and deployment. • For container-deployed components with **YAML Mode** disabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package or image package, the option is fixed to the software package type (JAR, WAR, or ZIP) or image package type selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**. • For container-deployed components with **YAML Mode** enabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package, the option is fixed to the software package type (JAR, WAR, or ZIP) selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**. • For VM-deployed components: If the component source is software package, the value is fixed to the software package type (JAR, WAR, ZIP, or compressed package) selected during component creation. It is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	• For container-deployed components with **YAML Mode** disabled: If the component source is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. If you select an image package: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • For container-deployed components with **YAML Mode** enabled: If the component source is software package, select an uploaded software package. For details about the upload method, see **Component Source**. • For VM-deployed components: If the component source is software package, you can select an uploaded software package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Image Access Credential	This parameter is available for container-deployed components. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.
*Command	This parameter is mandatory when the component is deployed based on a container, **YAML Mode** is disabled, the package type is **Source code repository**, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. For example: **cd ./weather/** **mvn clean package**
*Dockerfile Address	This parameter is mandatory when the component is deployed based on a container, **YAML Mode** is disabled, the package type is **Source code repository**, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Component Version	Component version number, which can be automatically generated or customized. ● Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. ● Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.
*Container Name	This parameter is mandatory when the component is deployed based on a container, the technology stack is not Docker, and **YAML Mode** is disabled. The name contains 1 to 63 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only lowercase letters, digits, and hyphens (-) are allowed.
Resources	When **YAML Mode** is disabled, the configuration of the selected historical version is used and cannot be modified.
JVM Parameters	When **YAML Mode** is disabled and the technology stack type is Java or Tomcat, the memory parameter size configured for running the Java code of the selected historical version is used and cannot be changed.
Tomcat	When **YAML Mode** is disabled and the technology stack type is Tomcat, the configuration, such as Tomcat request path and port number, of the selected historical version is used and cannot be changed.
Cloud Service Settings	● If the component is deployed based on a container and **YAML Mode** is disabled, the configuration of the selected historical version is used and cannot be changed. ● If the component is deployed based on a VM, the configuration of the selected historical version is used and cannot be modified.
Container Settings	If the component is deployed based on a container and **YAML Mode** is disabled, the configuration of the selected historical version is used and cannot be changed.
VM Settings	If the component is deployed based on a VM, the configuration of the selected historical version is used and cannot be modified.
Application Settings	The value is fixed to the configuration of the selected historical version and cannot be changed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Advanced Settings	If the component is deployed based on a container and **YAML Mode** is disabled, click **Upgrade Policy** to set **Deployment Batch** as required. Number of batches in which component instances are upgraded. The value range is [1, Total number of instances]. Total number of instances refers to the number of running instances of the component. For example, if there are 4 component instances and **Deployment Batch** is set to **2**, these component instances are upgraded in two batches, and each batch involves two component instances.
Log Settings	• If the component is deployed based on a container and **YAML Mode** is enabled, the configuration of the selected historical version is used and cannot be changed. • For VM-deployed components on an ECS, the configuration of the selected historical version is used and cannot be modified.
*Deployment Batch	This parameter is available when the component is deployed based on VM. Number of batches in which component instances are upgraded. The value range is [1, Total number of instances]. Total number of instances refers to the number of running instances of the component. For example, if there are 4 component instances and **Deployment Batch** is set to **2**, these component instances are upgraded in two batches, and each batch involves two component instances.
*Pause Before Stop	This parameter is available when the component is deployed based on VM. The Rx traffic will be processed during this period of time. The value ranges from 0 to 600, in seconds. The default value is **30**.
Best Effort Mode	This parameter is available when the component is deployed based on VM. • Disabled: When a component instance fails to be upgraded, the upgrade of the remaining instances will be stopped. • Enabled: If the upgrade fails, the remaining instances will be upgraded. Exercise caution when enabling this mode. After this mode is enabled, all instances may fail to be upgraded in extreme cases, affecting service functions.

Step 7 Click **Upgrade**.

- During component redeployment, if the component configuration is incorrect, you can click **Suspend** to stop the redeployment. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Single Component**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been redeployed.
- On the **View Deployment History** page, view the deployment logs.

  For container-deployed components, if "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

  For VM-deployed components, if "Creation of the VM application instance failure", "Query of the VM task status failure", or "Query of the VM application instance status failure" is displayed, click **View Event** to view details in the event list.

----End

5.5.3 Redeploying a Component in Dark Launch Mode

You can select a historical version configuration from the deployment record list and use the version configuration as a template to redeploy components in dark launch mode.

Introduction to Dark Launch

In dark launch mode, a certain proportion of instances are upgraded, and traffic is directed to the new version to verify whether functions of the new version are normal. Then, the remaining instances will be upgraded in rolling mode. Dark launch ensures stability of the entire system. During initial dark launch, problems can be detected and fixed.

Table 5-5 Dark launch types and description

Type	Description
Microservice Dark Launch	Applies to container-deployed ServiceComb and Spring Cloud components. Dark launch tasks function on microservices. Multiple microservices can work together to roll out new features. 1. The Java, Tomcat, or Docker technology stack must be selected for the component. 2. The component must be bound to a ServiceComb engine with security authentication disabled and multi-language access to service mesh disabled. • For ServiceComb engines, Java chassis 2.7.8 or later and Spring Cloud Huawei 1.10.4-2021.0.x or later are required. • For registry/configuration centers, Spring Cloud Huawei version must be 1.11.9-2021.0.x, 1.11.9-2022.0.x, 1.11.9-2023.0.x, or later.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Type	Description
ELB Dark Launch	Applies to ELB traffic-based and container-deployed components. Dark launch tasks function on ELB. • The public network access of the component must be enabled when **Creating and Deploying a Component**. • Use a dedicated application load balancer if you select a content-based dark launch policy. For details, see **Creating a Dedicated Load Balancer**.

Restrictions

During the dark launch upgrade in this operation, do not perform dark launch of the component bound to a ServiceComb engine or through registry/configuration enter. Otherwise, this operation will fail.

- For details about how to perform dark launch of a component through ServiceComb engines, see **Dark Launch**.
- For details about how to perform dark launch of a component through registry/configuration centers, see **Configuring Dark Launch**.

Prerequisites

- You have upgraded a component. For details, see **Upgrading a Single Component** or **Upgrading Components in Batches**.
- A CCE cluster must be bound to the component deployment environment.
- The component is a stateless component deployed based on a container. If the component technology stack is Docker, multi-container deployment must be disabled.
- The component version configuration that has been rolled back by referring to **Rolling Back a Component** cannot be used as a template to redeploy the component.

Redeploying a Component in Dark Launch Mode

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **View Deployment History** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Deployment Records** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Deployment Records** in the left navigation pane.

Step 3 In the **Deployment Records** list, select the deployment record of the historical version to be used as the configuration template.

Step 4 Click **Redeploy** in the upper right corner of the page. The **Redeploy** dialog box is displayed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 5 Select **Dark Launch** for **Deployment Type** and click **OK**.

Step 6 Configure the component version by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
Stack	Select a component technology stack and version. For details, see **Technology Stack**.
*Software Package/ Image	The value is fixed to the package type selected during component creation and deployment. • If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. • If you select a software package or image package, the option is fixed to the software package type (JAR, WAR, or ZIP) or image package type selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**.
*Upload Method	If the component source is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • If you select an image package: – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – In the upper right corner of the container card, choose > **Modify Image Package** to modify the image package source of the container.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Image Access Credential	An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.
*Command	This parameter is mandatory when the package type is **Source code repository** and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. For example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	This parameter is mandatory when the package type is **Source code repository** and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Component Version	Component version number, which can be automatically generated or customized. • Automatically-generated: Click **Generate**. By default, the version number is the time when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.
*Container Name	This parameter is mandatory when the technology stack is not Docker. The name contains 1 to 63 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only lowercase letters, digits, and hyphens (-) are allowed.
Resources	The value is fixed to the configuration of the selected historical version and cannot be changed.
JVM Parameters	When the technology stack type is Java or Tomcat, the memory parameter size configured for running the Java code of the selected historical version is used and cannot be changed.
Tomcat	When the technology stack type is Tomcat, the configuration, such as Tomcat request path and port number, of the selected historical version is used and cannot be changed.
Container Settings	The value is fixed to the configuration of the selected historical version and cannot be changed.
Application Settings	The value is fixed to the configuration of the selected historical version and cannot be changed.
Advanced Settings	The value is fixed to the configuration of the selected historical version and cannot be changed.
*Deployment Architecture	You must choose the correct architecture. Otherwise, dark launch will fail and services will be affected. 1. Click **Select** and select an instance deployment architecture for dark lunch. 2. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Dark Launch Policy	Select a dark lunch policy. • **Traffic ratio-based**: Flexibly and dynamically adjust the traffic ratio of different versions. • **Content-based**: Control the version to which the request is sent according to the request.
*Traffic Ratio	When **Dark Launch Policy** is set to **Traffic ratio-based**, set the traffic ratio. • **Traffic Ratio**: percentage of traffic directed to the new version. • **Current Traffic Ratio**: percentage of traffic directed to the current version.
*Takes effect	When **Dark Launch Policy** is set to **Content-based**, set the effective mode of the dark launch policy. • **With single criterion**: The dark launch policy takes effect when any matching rule is met. • **With full criteria**: The dark launch policy takes effect when all matching rules are met.
*Matching Rule	When **Dark Launch Policy** is set to **Content-based**, set the matching rule for the dark launch policy to take effect. 1. Click **Add Matching Condition**. 2. Set **Matching Type**. Currently, only **Request header** is supported. 3. Set **Parameter Name**, that is, the key corresponding to **Matching Type**. 4. Set **Condition Type**, that is, the matching rule that **Condition Value** meets. – **Equal**: The value corresponding to **Matching Type** is equal to the configured **Condition Value**. – **Matching**: **Condition Value** supports PERL-compatible regular expressions. It is supported by only microservice dark launch. – **Enumerated**: Multiple **Condition Value** items are separated by commas (,). 5. Set **Condition Value**, that is, the value corresponding to **Matching Type**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Instances Deployed for Dark Launch	Select a mode for adding an instance during dark launch. To change the number of component instances to meet the requirements, see **Configuring a Manual Scaling Policy**. • Blue-green Add new version instances with the same number as the old version instances, and switch traffic to the new version based on dark launch rules. After verification based on the dark launch rules, switch all traffic to the new version and delete the old version. Scenarios: lossless service capacity, fast rollback, and when resources are needed in the cluster. This mode requires one or more component instances. • Canary (increase, then decrease) Add new version instances with the specified number, and switch traffic to the new version based on dark launch rules. After verification based on the dark launch rules, delete some old instances, add corresponding new instances, and switch traffic. Repeat this process until all traffic is switched to the new instances and old instances are deleted. Scenarios: lossless service capacity, slow rollback, and when resources are needed in the cluster. This mode requires two or more component instances. • Canary (decrease, then increase) Delete some old instances and add corresponding new instances, and switch traffic to the new version based on dark launch rules. After verification based on the dark launch rules, delete some old instances, add corresponding new instances, and switch traffic. Repeat this process until all traffic is switched to the new instances and old instances are deleted. Scenarios: lossy service capacity and slow rollback. This mode requires two or more component instances.
*First Batch of Dark Launch Instances	Set this parameter when **Instances Deployed for Dark Launch** is set to **Canary (increase, then decrease)** or **Canary (decrease, then increase)**. The value range is [1, Total number of instances – 1]. Total number of instances refers to the number of running instances of the component. For example, if there are 6 component instances and **First Batch of Dark Launch Instances** is set to **1**, 1 instance will be upgraded in the first batch. The remaining 5 instances will be upgraded according to **Deployment Batch with Remaining Instances**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Deployment Batch with Remaining Instances	Set this parameter when **Instances Deployed for Dark Launch** is set to **Canary (increase, then decrease)** or **Canary (decrease, then increase)**. That is, the number of batches whose remaining instances will be upgraded. For example, if there are 6 component instances, **First Batch of Dark Launch Instances** is set to **1**, and **Deployment Batch with Remaining Instances** is set to **3**, there are 5 instances remaining to be deployed in 3 batches, and these 5 instances will be upgraded in the sequence 2:2:1.

Step 7 Click **Upgrade**.

- During component redeployment, if the component configuration is incorrect, you can click **Suspend** to stop the redeployment. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Single Component**.
- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been redeployed.
- On the **View Deployment History** page, view the deployment logs.

  If "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

----End

Follow-up Operations

Operation	Description
View the deployment logs	Select a deployment record in the **Deployment Records** list and view component deployment logs on the **Deployment Log** tab.
View system monitoring	If the component is upgraded through dark launch, choose **System Monitoring** to check the CPU and memory usage of instances of the dark launch version and the current version after the first batch of dark launch is complete.
Modify a dark launch policy	If the component is upgraded through dark launch, you can modify the dark launch policy by referring to **Modifying a Component Dark Launch Policy** after the first batch of dark launch is complete.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Complete processes	If the component is upgraded through dark launch after the first batch of dark launch is successful and the functions of the new version are normal, you can perform the following operations to upgrade the remaining component instances: 1. Select the deployment record of the **Dark Launch** type. 2. Click **Complete Processes**. 3. In the displayed dialog box, click **OK**. Upgrade the remaining instances to the new version.
Roll back a component	The component can be rolled back in the following scenarios: • The first batch of dark launch is complete if the component is upgraded through dark launch. • All component instances have been upgraded to the new version. To roll back the component configuration to the source version, refer to **Rolling Back a Component**.

5.5.4 Redeploying a Component Using a Component Template

Redeploy a component created by **Creating a Component Using a Component Template**.

Prerequisites

- You have upgraded a component. For details, see **Upgrading a Component Using a Component Template**.
- The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**.
- You have compressed the prepared component template files into a ZIP package and uploaded the package to the repository. For details, see **Component Template Description**.
  - Upload the template package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Select the corresponding template package from OBS. Upload the template package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**.
  - If you need to use JFrog (example) as the repository for storing the component template package, you can use an HTTP/HTTPS custom file address as the component template package download address. Upload the component template package to the custom file address in advance.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Redeploying a Component Using a Component Template

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **View Deployment History** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Deployment Records** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Deployment Records** in the left navigation pane.

Step 3 In the **Deployment Records** list, select the deployment record of the historical version to be used as the configuration template.

Step 4 Click **Redeploy** in the upper right corner of the page. The **Redeploy** dialog box is displayed.

Step 5 Select a deployment type.

- Single-batch release: During component redeployment in this mode, component services will be interrupted. This is supported only by container-deployed components and is applicable to the test upgrade scenario or the upgrade scenario where services are to be stopped.
- Rolling release: Only one or more instances are deployed at a time and then added to the production environment. This process repeats until all old instances are upgraded. Services will not be interrupted during the deployment.

Step 6 Click **OK**.

Step 7 In the **Set Template Package** area, select the uploaded component template package based on the template package storage mode.

- CodeArts release repository
  - If CodeArts release repository is not the component template repository, click **Select Software Package**, select the uploaded template package from the CodeArts release repository, and click **OK**.
  - Otherwise, click , select the uploaded template package from the CodeArts release repository, and click **OK**. Alternatively, click > **Select Software Package**, select the uploaded template package from the CodeArts release repository, and click **OK**.
- OBS
  - If OBS is not the component template repository, click **Select Software Package**, select the uploaded template package from the OBS bucket, and click **OK**.
  - Otherwise, click , select the uploaded template package from the OBS bucket, and click **OK**. Alternatively, click > **Select Software Package**, select the uploaded template package from the OBS bucket, and click **OK**.
- SWR repository
  - If SWR repository is not the component template repository, click **Select Software Package**, select the uploaded template package from the SWR repository, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

  - Otherwise, click , select the uploaded template package from the SWR repository, and click **OK**. Alternatively, click > **Select Software Package**, select the uploaded template package from the SWR repository, and click **OK**.

- **Custom file address**: If you need to use JFrog (example) as the repository for storing the template package, perform the following operations to obtain the template package that has been uploaded to the custom file address:

  a. Enter the HTTP/HTTPS custom file address where the software package is located. The custom file address must be the complete path of the component template package file, for example, **https://172.16.0.1:8082/artifactory/template-component.zip**.

  b. Determine whether to enable authentication.

     If authentication is disabled, any user can download the software package in the custom file address by default.

     Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

Step 8 Click **Next**.

- In the **Preview Template Package** area, you can view the details of the component template file.
- In the **Set Parameters** area, you can view and modify the component parameters.

Step 9 Click **Upgrade**.

- During component redeployment, if the component configuration is incorrect, you can click **Suspend** to stop the redeployment. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Component Using a Component Template**.
- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been upgraded.
- On the **Deployment Records** page, view the deployment logs.

----End

5.5.5 Redeploying a Job Component

Container-deployed components whose **Workload Type** is **Job** can be redeployed as required.

Prerequisites

- Only the components whose **Workload Type** is **Job** and that are created and deployed using the following methods support this operation:
  - **Creating a Component Based on a Container Using Manual Configurations**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component Using a Component Template**
- The component status is **Completed** or **Failed**. For details about how to check the component status, see **Viewing Component Details**.

Redeploying a Job Component

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page.

- On the **Application Management** page, click the application to which the target component belongs, and click the component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 Click **Redeployment** in the upper right corner of the page.

Step 4 Configure the component version by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
Stack	Select a component technology stack and version. For details, see **Technology Stack**.
*YAML Mode	Whether to use YAML configurations to redeploy components. • Disabled: The UI configurations are used to redeploy components. • Enabled: The YAML configurations are used to redeploy components. The latest load information of the component is automatically synchronized from CCE where the target component is deployed for component redeployment after modification. Alternatively, click **Import YAML File** to import the edited YAML configuration file of the target component. If YAML configurations are used to redeploy components, the parameters in the YAML configuration file are described in **Deployment**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Software Package/ Image	The value is fixed to the package type selected during component creation and deployment. • **YAML Mode** disabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package or image package, the option is fixed to the software package type (JAR, WAR, or ZIP) or image package type selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**. • **YAML Mode** enabled: If you select **Source code repository**, create authorization by referring to **Creating Repository Authorization** and set the code source. If you select a software package, the option is fixed to the software package type (JAR, WAR, or ZIP) selected during component creation and deployment. It is determined by the selected technology stack type. For details, see **Table 5-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Upload Method	• **YAML Mode** disabled: If the component source is software package or image package, select an uploaded software package or image package. For details about the upload method, see **Component Source**. If you select an image package: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – If **Multi-container** is selected in **Creating and Deploying a Component**, click **Select Image Package** to add a container and select an image package source. – If **Multi-container** is selected in **Creating and Deploying a Component**, choose > **Modify Image Package** in the upper right corner of the selected container card to modify the image package source of the container. – If **Multi-container** is selected in **Creating and Deploying a Component**, choose > **Delete** in the upper right corner of the selected container card to delete the container. – If **Multi-container** is selected in **Creating and Deploying a Component**, deselect **Multi-container** to deploy a component based a single container. – If **Multi-container** is not selected in **Creating and Deploying a Component**, select **Multi-container** to deploy a component based on multiple containers. Click **Select Image Package** to add a container, select an image package source, and set **Container Name**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • **YAML Mode** enabled: If the component source is software package, select an uploaded software package. For details about the upload method, see **Component Source**. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.
Image Access Credential	An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled. • You can select up to 16 image access credentials. • Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.
*Command	This parameter is mandatory when **YAML Mode** is disabled, the package type is **Source code repository**, the component is deployed in the Kubernetes environment, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. • **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. • **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. – If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. – To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather/** **mvn clean package**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Dockerfile Address	This parameter is mandatory when **YAML Mode** is disabled, the package type is **Source code repository**, the component is deployed based on a container, and the selected technology stack type is Java, Tomcat, Node.js, Python, or PHP. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Component Version	Component version number, which can be automatically generated or customized. ● Automatically-generated: Click **Generate**. By default, the version number is the timestamp when you click **Generate**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. ● Customized: Enter a value in the format of A.B.C or A.B.C.D. A, B, C, and D are natural numbers. For example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.
*Container Name	This parameter is mandatory when the technology stack is not Docker and **YAML Mode** is disabled. The name contains 1 to 63 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only lowercase letters, digits, and hyphens (-) are allowed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Resources	This parameter is available when **YAML Mode** is disabled. You can customize **CPU** and **Memory** to set their quota, and set the maximum/minimum number of CPU cores and memory size (GiB) that can be used by components. In this way, you can set the number of resources required by each component instance. When resource **Request** is specified for a component instance, CCE uses this information to determine the node to which the component instance is scheduled. When resource **Limit** is specified for a component instance, CCE ensures that the resources occupied by the running component instance do not exceed the limit. CCE will also reserve the requested number of system resources for the component instance. For details about how to configure them, see **Resource Management**. To modify them, select the item to be changed and enter a new value. Unselected parameters indicate no restriction. If the package type selected for **Upload Method** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.
JVM Parameters	This parameter is available when **YAML Mode** is disabled and the technology stack type is Java or Tomcat. It configures the memory size during Java code running. Enter the JVM parameter. For example: `-Xms256m -Xmx1024m` Multiple parameters are separated by spaces.
Tomcat	This parameter is available when **YAML Mode** is disabled and the technology stack type is Tomcat. It configures parameters such as the request path and port number of Tomcat. 1. Select **Tomcat**. The **Tomcat** dialog box is displayed. 2. Click **Use Sample Code** and edit the template file based on service requirements. In Tomcat configuration, the default **server.xml** configuration is used. The context path is **/**, and no application path is specified. If you need to customize an application path, customize the Tomcat context path by referring to **How Do I Customize a Tomcat Context Path?** 3. Click **OK**.
Cloud Service Settings	If **YAML Mode** is disabled, set this parameter as required by referring to **Managing Cloud Service Settings of a Component**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Container Settings	If **YAML Mode** is disabled, set the following parameters as required by referring to **Managing Container Settings of a Container-Deployed Component**. • Set **Environment Variable** by referring to **Adding an Environment Variable of a Component**. • Set **Startup Command** by referring to **Configuring the Startup Command and Lifecycle of a Component**. • Set **Data Storage** by referring to **Configuring Data Storage**. • Set **Lifecycle** by referring to **Configuring the Startup Command and Lifecycle of a Component**. • Set **Log Collection** by referring to **Configuring a Log Policy of an Application**.
Application Settings	Set this parameter as required by referring to **Managing Application Settings of a Container-Deployed Component**. If the package type selected for **Upload Method** is image package and multi-container deployment is enabled, set this parameter for each container instance as required.
Log Settings	If **YAML Mode** is enabled, set this parameter as required by referring to **Step 14**.
Advanced Settings	If **YAML Mode** is disabled, set this parameter as required by referring to **Configuring Component Job Settings**.

Step 5 Click **Redeployment** and wait until the component is deployed.

----End

5.6 Rolling Back a Component

You can roll back a component from the latest version to the version before the upgrade or redeployment.

Restrictions

During the rollback, component services will be interrupted. Do not perform this operation during peak hours.

Prerequisites

- Any of the following operations has been performed on the target component:
  - **Upgrading a Single Component**
  - **Upgrading Components in Batches**
  - **Redeploying a Component**
- The component has not been rolled back.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Rolling Back a Component

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Deployment Records** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Deployment Records** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Deployment Records** in the left navigation pane.

Step 3 In the **Deployment Records** list, select the deployment record of the latest version.

Step 4 Click **Roll Back**.

Step 5 In the displayed dialog box, click **OK**.

- Wait until the component status changes from **Upgrading/Rolling back the component** to **Running**, indicating that the component has been rolled back to the source version.
- On the **View Deployment History** page, view the deployment logs.

  For container-deployed components, if "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

  For VM-deployed components, if "Creation of the VM application instance failure", "Query of the VM task status failure", or "Query of the VM application instance status failure" is displayed, click **View Event** to view details in the event list.

----End

5.7 Cloning Components in Batches

This avoids complex and repeated configurations because multiple microservice components may have same configurations, for example, scheduling policy, AS policy, data storage, log storage, and lifecycle configurations, and only minor differences need to be modified. ServiceStage provides the function of cloning and deploying components in batches to improve component deployment efficiency and user experience.

Prerequisites

- Only the components created and deployed using the following methods can be cloned in batches:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component Based on a VM**
  - **Creating a Component by Importing a CCE Workload**
- The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Cloning Components in Batches

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Click the application where the target components are located. The **Overview** page of the application is displayed.

Step 4 Select the components to be cloned in batches in **Component List** and click **More** > **Clone**.

Step 5 Set the version configuration information of the components to be cloned by referring to the following table.

Parameter	Description
Component	Name of a component, which cannot be changed after the component is cloned. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). For container-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments. For VM-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different environments.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Component Version	Version number of a component. • By default, the version number is the time when you start to clone components in batches. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. You can perform the following operations to synchronize component versions in batches: 1. Move the cursor to the **Component Version** text box of the specified component and click . 2. Select other components whose versions need to be synchronized. 3. Click **OK**.
Application	Select the application to which the component belongs. You can perform the following operations to synchronize the applications to which components belong in batches: 1. Move the cursor to the **Application** drop-down list of the specified component and click . 2. Select other components whose applications need to be synchronized. 3. Click **OK**.
Environment	Select the component deployment environment. For container-deployed components, only the Kubernetes environment can be selected. For VM-deployed components, only the VM environment can be selected. You can perform the following operations to set the environment to which components belong in batches for components deployed in the same mode: 1. Move the cursor to the **Environment** drop-down list of the specified component and click . 2. Select other components whose environments need to be synchronized. 3. Click **OK**. After the environment is reset, the cluster to which the container-deployed components belong is changed to the cluster bound to the new environment, and the namespace to which the component belongs is changed to the default namespace of the cluster bound to the new environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Cluster	Select a CCE cluster for deploying and running the container-deployed component. You can perform the following operations to synchronize the namespaces to which components belong in batches: 1. Move the cursor to the **Cluster** drop-down list of the specified component and click . 2. Select other components whose clusters need to be synchronized. 3. Click **OK**. After the cluster is set, the cluster is changed only when the selected cluster is bound to the environment to which the component belongs. In addition, the namespace to which the component belongs is changed to the default namespace of the new cluster.
Namespace	Select the namespace of the CCE cluster in the build environment for the container-deployed component to isolate build data. For details, see **Managing Namespaces**. You can perform the following operations to synchronize the namespaces to which components belong in batches: 1. Move the cursor to the **Namespace** drop-down list of the specified component and click . 2. Select other components whose namespaces need to be synchronized. 3. Click **OK**. After the namespace is set, the namespace is changed only when the selected namespace exists in the cluster to which the component belongs.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click and select another package type. For details, see **Component Source**. The technology stack and version of the reselected package type must be the same as those of the component to be cloned. Otherwise, the cloning will fail. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Instances	Set the number of running instances of the component. • For container-deployed components, the value ranges from 1 to 200. You can perform the following operations to synchronize the number of component instances in batches: 1. Move the cursor to the **Instances** text box of the specified component and click . 2. Select other components whose number of component instances need to be synchronized. 3. Click **OK**. • For VM-deployed components, you can perform the following operations to reset the number of component instances: 1. Click **Select**. 2. Reselect the number of servers for component deployment. The servers to be selected must have the VM agent installed and be running properly. If the status of the target server agent indicates that the agent is missing and to install it first, click **Install**. For details, see **Step 6** to **Step 8** in **Installing a VM Agent**. 3. Click **OK**.

Step 6 Reset the component build parameters.

This parameter is available when the component is deployed based on a container and the technology stack type is Java, Tomcat, Node.js, Python, or PHP.

1. Locate the target component and click **Build** in the **Operation** column.
2. Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. – **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. – **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. ▪ If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. ▪ To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The build environment must be a Kubernetes environment, and can access the Internet. You are advised to select **Use current environment**. If the CCE cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. – **Use independent environment**: Use an independent build environment to build an image. The CCE clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. – **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the CCE cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	– If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. – If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Select Cluster	Select a cluster for build if **Environment** is set to **HA Environment**. For details about the HA environment, see **Environment Overview**.
*Namespace	Select the namespace of the CCE cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Node Label	In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: – If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. – If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.

3. Click **OK**.

Step 7 Locate the target component and click **Advanced Settings** in the **Operation** column.

Step 8 Select a component technology stack and version. For details, see **Technology Stack**.

Step 9 Set **Headless Service Settings** for container-deployed stateful components.

It provides a fixed access domain name for every pod for mutual pod access in a StatefulSet.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 10 Set **Public Network Access**.

- For container-deployed components, go to **Step 11**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For VM-deployed components, go to **Step 12**.

Step 11 Click to enable public network access for container-deployed components.

Public network access can be configured if **Workload Type** of the component is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the component deployment environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If you need to upgrade and maintain components in ELB dark launch mode, enable public network access for the components. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

  a. Set **Public Network Load Balancer**.

  An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

  - Select a load balancer that has been bound to an EIP in the selected environment.
  - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
  - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	○ **Creating a Shared Load Balancer**. ○ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

  b. Set **Client Protocol**.

  - **HTTP** has security risks. You are advised to select **HTTPS**.
  - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

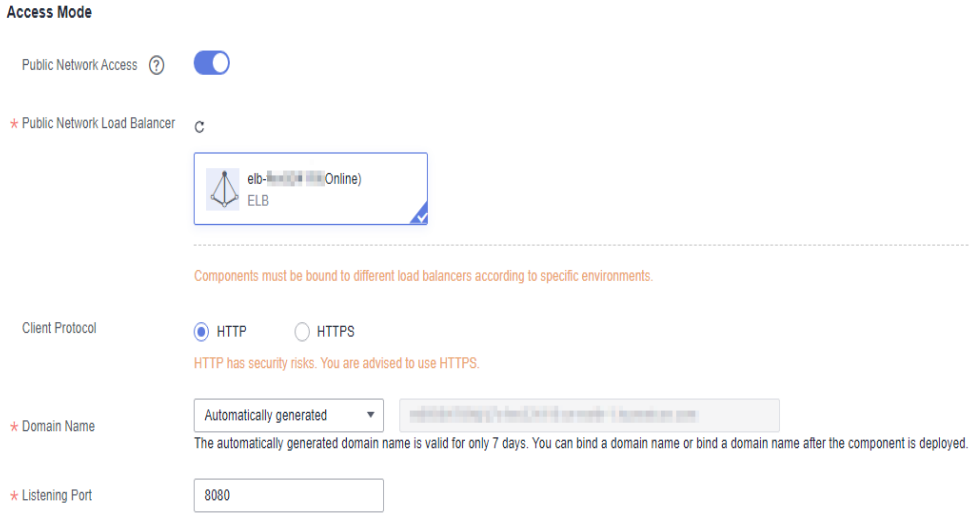
c. Set **Domain Name**.

- If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
- If **Bound** is selected, enter a domain name.

d. Set **Listening Port**.

Enter the listening port number of the application process.

Figure 5-7 Configuring public access



Step 12 Click to enable public network access for VM-deployed components.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources:
     - Create a shared load balancer by referring to **Creating a Shared Load Balancer**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

    - Create a dedicated application load balancer by referring to **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.

   After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

Step 13 Set other component configurations.

- For container-deployed components, see **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components, see **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.

Step 14 Click **OK**.

Step 15 (Optional) Click **Pre-check Advanced Settings** to check whether the advanced settings of each component are correct.

If the component has been bound to a specified microservice engine, distributed cache, or cloud database instance but the specified instance is not bound to the environment to which the component belongs, an error is reported during advanced settings pre-check. Perform the following operations to reconfigure the parameters and then continue:

1. In the **Pre-check Advanced Settings** dialog box that is displayed, confirm the information and click **OK**.
2. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 13**.

Step 16 Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct.

- If no, reset the parameters based on the error information by referring to **Step 13**.
- If yes, wait until the component statuses change from **Initializing** to **Running**, indicating that the components have been cloned. On the **Deployment Records** page, view the deployment logs.

  For container-deployed components, if "Querying the Status of a Workload Instance" is displayed, click **View Event** to view details.

  During the component cloning, if a component is incorrectly configured, you can click the component to go to the **Overview** page and click **Suspend** to

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

stop the component cloning. Then, click **Upgrade** and reconfigure the component by referring to **Upgrading a Single Component**.

----End

5.8 Synchronizing Component Configurations in Batches

You can synchronize latest configurations after container-based component workload changes (such as workloads upgraded using CCE).

Prerequisites

- Only the components created and deployed using the following methods support batch component version synchronization:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component by Importing a CCE Workload**
- The component status is **Running**, **Not ready**, **Processing**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**.

Synchronizing Component Configurations in Batches

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Click the application where the target components are located. The **Overview** page of the application is displayed.

Step 4 Select the components whose configurations are to be synchronized in batches in **Component List** and click **More** > **Sync**.

----End

5.9 Managing Component Details

5.9.1 Managing Component Labels

Labels are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use labels to classify them based on usage, owner, or environment. Each label contains a key and a value. Labels are attached to resources in the format of "key:value".

The following describes how to manage labels of a container-deployed stateless workload. Workload labels are often used for affinity and anti-affinity scheduling. For details, see **Configuring a Scheduling Policy of a Component Instance**. You can add labels to multiple workloads or a specified workload.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

You can manage the labels of stateless workloads, stateful workloads, and Daemon sets based on service requirements. This section uses stateless workloads as an example to describe how to manage labels.

As shown in **Figure 5-8**, labels release, env, and role are defined for workloads APP 1, APP 2, and APP 3. The values of these labels vary with workload.

- APP 1: [release:alpha;env:development;role:frontend]
- APP 2: [release:beta;env:testing;role:frontend]
- APP 3: [release:alpha;env:production;role:backend]

If you set **key** to **role** and **value** to **frontend** when using workload scheduling or another function, APP 1 and APP 2 will be selected.

Figure 5-8 Label example

ServiceStage allows you to add labels of the key-value pair type to manage and select components.

Each component can have a maximum of 20 labels.

If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, environment creation may fail. Contact your administrator to learn more about tag policies.

Labels can be managed for components in the **Running** and **Not ready** states. For details about how to check the component status, see **Viewing Component Details**.

This section describes how to add or delete a component label.

Adding a Component Label

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 Click **Manage Label**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 On the displayed page, click **Add Label** and enter a key and value.

It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources.

- The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). The key cannot be the same as an existing key, and cannot be the built-in **app**, **casid**, or **version**.
- The value contains 0 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.).

Step 5 Click **OK**.

----End

Deleting a Component Label

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 Click **Manage Label**.

Step 4 Select the label to be deleted and click ⊖.

Step 5 Click **OK**.

----End

5.9.2 Changing the Component Description

After a component is created and deployed, you can change the description.

Prerequisites

Components in the **Running** and **Not ready** states support description change. For details about how to check the component status, see **Viewing Component Details**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Changing the Component Description

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 Change the component description.

1. Click next to the description.

2. Enter up to 128 characters as the component description in the text box.
3. Click to save the component description.

----End

5.9.3 Managing Component Instances

After a component is created and deployed, you can manage component instances on the component **Instance List** page.

Context

For components deployed on VMs, a component instance is a running process on the VM. For components deployed in CCE clusters, a component instance is a pod, which is the minimum unit for deploying applications or services.

- For components deployed on VMs:
  - When **Creating a Component Based on a VM** or **Creating a Component Using a Component Template**, if the component name contains fewer than 54 characters, the component instance name can contain up to 61 characters. The instance name format is as follows:
    `{All of the component name}-{All or part of the environment name}_{Five random characters generated by the system}`
  - When **Creating a Component Based on a VM** or **Creating a Component Using a Component Template**, if the component name contains 54 characters or more, the component instance name must contain 60 characters. The instance name format is as follows:
    `{First 54 characters of the component name}_{Five random characters generated by the system}`
- For components deployed in CCE clusters:
  - When **Creating a Component Based on a Container Using Manual Configurations**, **Creating a Component Based on a Container Using YAML Configurations**, or **Creating a Component Using a Component Template**, if you select **Stateful** for **Workload Type** and customize a

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

workload name, the component instance name can contain up to 52 characters.

- When **Creating a Component Based on a Container Using Manual Configurations**, **Creating a Component Based on a Container Using YAML Configurations**, or **Creating a Component Using a Component Template**, if you select **Stateless** or **DaemonSet** for **Workload Type** and customize a workload name, the component instance name can contain up to 63 characters.
- When **Creating a Component Based on a Container Using Manual Configurations**, **Creating a Component Based on a Container Using YAML Configurations**, or **Creating a Component Using a Component Template**, if the workload name is automatically generated by the system, the component instance name can contain up to 52 characters.
- When **Creating a Component by Importing a CCE Workload**, the workload name is fixed to the name of the imported CCE workload.

The number of component instances deployed on a VM is the number of VMs selected during component deployment. The number of component instances deployed in a CCE cluster is the number of instances set during component deployment, that is, the number of pods.

Managing Component Instances

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Instance List** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 On the **Instance List** page, you can perform the following operations.

Operation	Description
Restart a single instance	If an instance of a container-deployed component is abnormal, you can delete the instance to restart it. 1. In **Instance List**, locate the target instance and choose **More** > **Delete Instance** in the **Operation** column. 2. In the displayed dialog box, click **OK**. Manually deleting an instance may cause residual resources on the node where the instance is located. You need to manually clear these resources.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
View instance running monitoring information	By viewing the instance running monitoring information, you can learn about the CPU and memory usage of a single running instance. 1. In **Instance List**, locate the target instance and click **Monitoring** in the **Operation** column. 2. On the **Monitoring** tab, view the instance running monitoring information.
View instance running events	ServiceStage allows you to view details about events that occur during the running of a specified instance. 1. In **Instance List**, locate the target instance and click **Events** in the **Operation** column. 2. On the **Events** tab, view the events that occur during instance running.
View running instance containers	For container-deployed components, ServiceStage allows you to view information about the container where a specified instance runs, including the container name, running status, and mounted image. 1. In **Instance List**, locate the target instance and choose **More** > **Containers** in the **Operation** column. 2. On the **Containers** tab, view the information about the container where the instance runs.
View instance container logs	For container-deployed components, you can view component container logs to locate and rectify faults that occur during component running. 1. In **Instance List**, locate the target instance and choose **More** > **Container Logs** in the **Operation** column. 2. On the **Container Logs** tab, view the information about the container where the instance runs. – Select the container where the component instance is located from the **Container** drop-down list. – Specify **Lines** to configure how many rows you can view. You can view up to 100, 200, 500, 5000, or 10,000 lines. – Click **Download** to download the logs. You can also view component container logs by referring to **Viewing Container Logs**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.9.4 Configuring the Component Access Mode

This section describes how to configure the access mode for a component. After the configuration, you can access the services provided by the component in the configured mode.

Prerequisites

- You can set the component access mode only for the components created and deployed using the following methods:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component by Importing a CCE Workload**
- The component status is **Running**. For details about how to check the component status, see **Viewing Component Details**.
- The workload type of the component must be **Stateless**, **Stateful**, or **DaemonSet**.

Configuring the Component Access Mode

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Access Mode** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Access Mode** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Access Mode** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 Click **Add Service** and set the following parameters. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Service Name	The service name can be the same as the component name. The name contains 1 to 63 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only lowercase letters, digits, and hyphens (-) are allowed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Access Mode	If a CCE cluster is bound to the component deployment environment, the following service access modes are supported: • **Intra-cluster access**: allows access from other services in the same cluster over TCP/UDP. • **Intra-VPC access**: allows access from other services in the same VPC over TCP/UDP. • **Public network access**: allows access from the Internet over TCP/UDP, including EIP. If a custom Kubernetes cluster is bound to the component deployment environment, the following access service modes are supported: • **Intra-cluster access**: allows access from other services in the same cluster over TCP/UDP. • **NodePort**: exposes services through each Node's IP address and static port (NodePort).
Intra-VPC Load Balancing	This parameter is available when **Access Mode** is set to **Intra-VPC access**.
*Access Type	• This parameter is available when **Access Mode** is set to **Intra-VPC access** and **Intra-VPC load balancing** is enabled. • This parameter is available when **Access Mode** is set to **Public network access**.
Service Affinity	This parameter is available when **Access Mode** is set to **Intra-VPC access** or **Public network access**.
*Port Mapping	Sets **Protocol**, **Container Port**, and **Access Port** for accessing the service. • **Protocol**: The value can be **TCP**, **UDP**, or **TCP/UDP**. • **Container Port**: The value range is 1 to 65535. • **Access Port**: The value range is 1 to 65535.

Step 5 Click **OK**.

----End

5.9.5 Changing the Component Access Domain Name

For components with enabled public network access and set access domain names, you can change the domain names after the components are deployed.

Prerequisites

- An automatically generated domain name is valid only for seven days. After the validity period expires, the domain name must be changed to a custom domain name.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- You can set the component access mode only for the components deployed using the following methods with public network access enabled:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component by Importing a CCE Workload**
  - **Creating a Component Based on a VM**
- The component status is **Running**. For details about how to check the component status, see **Viewing Component Details**.
- You have obtained the domain name from the domain name provider.
- You have obtained the elastic IP address of the ELB bound to the component.

Changing the Component Access Domain Name

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Access Mode** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Access Mode** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Access Mode** in the left navigation pane.

Step 3 Click **Set domain**.

1. Enter the obtained **Domain Name**.
2. Enter **Listening Port**.
3. (Optional) Enable **HTTPS**.
   - Click **Use existing** to select an existing certificate.
   - Click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

----End

5.9.6 Starting or Stopping a Component

After components are deployed, you can start, stop, or restart one or more components as required.

Starting or Stopping a Single Component

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to start or stop a component:

- Method 1

  Use either of the following methods to go to the **Overview** page and start or stop a component by referring to **Table 5-6**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

  - On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
  - On the **Component Management** page, click the target component.
- Method 2

  Use either of the following methods to go to the component list and start or stop a component by referring to **Table 5-6**.
  - On the **Application Management** page, click the application to which the component belongs. In **Component List**, click **More** in the **Operation** column of the target component.
  - On the **Component Management** page, click **More** in the **Operation** column of the target component.

Table 5-6 Starting or stopping a single component

Operation	Description
Stop a component	Click **Stop** to stop a component in the **Running** or **Processing** state.
Start a component	Click **Start** to start a component in the **Stopped** state.
Restart a component	Click **Restart** to restart a component in the **Running**, **Not ready**, or **Imported and to be restarted** state. • For components in the **Running** or **Not ready** state, click **OK** and wait until the component status changes from **Restarting** to **Running**. • For components in the **Imported and to be restarted** state, click **Enable** on the **Enable Component** page and wait until the component is restarted.

----End

Starting or Stopping Multiple Components

Step 1 Log in to ServiceStage.

Step 2 Choose **Application Management**.

Step 3 Click the application where the target components are located. The **Overview** page of the application is displayed.

Step 4 Select the target components in **Component List** and start or stop them by referring to **Table 5-7**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-7 Starting or stopping multiple components

Operation	Description
Stop components	Click **Stop** above the component list to stop components in the **Running** or **Processing** state.
Start components	Click **Start** above the component list to start the components in the **Stopped** state.
Restart components	Choose **More** > **Restart** above the component list to restart the components in the **Running**, **Not ready**, or **Imported and to be restarted** state.

----End

5.9.7 Manually Synchronizing the Component Status

ServiceStage periodically syncs the number and status of component instances from compute resources that run component instances. To manually synchronize the component status, perform the following operations:

Prerequisites

- Only components in the **Running**, **Not ready**, **Abnormal**, or **Failed** state support manual component status synchronization. For details about how to check the component status, see **Viewing Component Details**.
- For container-deployed components, **Workload Type** must be **Stateless**, **Stateful**, or **DaemonSet**.

Manually Synchronizing the Component Status

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to select the target component:

- On the **Application Management** page, click the application to which the target component belongs, and click the component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 In the **Operation** column, choose **More** > **Sync** to synchronize the component status.

----End

5.10 Configuring a Scaling Policy of a Component Instance

5.10.1 Component Instance Scaling Policy Overview

After scaling policies are configured, instances can be automatically added or deleted based on resource changes or a specified schedule. This reduces manual

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

resource adjustment to cope with service changes and service peak, helping you save resources and labor costs.

ServiceStage supports the following scaling policies:

- **Manual scaling**: Manually configure an instance scaling policy for a component. After the policy is configured, the number of component instances increases or decreases immediately.
- **HPA**: In Kubernetes, a HorizontalPodAutoscaler is a built-in component that provides horizontal pod autoscaling (HPA). It supports the application-level cooldown time window and scaling threshold functions based on the Kubernetes HPA.

5.10.2 Configuring a Manual Scaling Policy

This section describes how to manually configure an instance scaling policy for a component. After the policy is configured, the number of component instances increases or decreases immediately.

Prerequisites

- You can set the manual scaling policy only for the components created and deployed using the following methods:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component Based on a VM**
  - **Creating a Component by Importing a CCE Workload**
- The component is deployed based on a container and the **Workload Type** is **Stateless** or **Stateful**.
- The component status is **Running** or **Not ready**. For details about how to check the component status, see **Viewing Component Details**.

Configuring a Manual Scaling Policy

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Scaling** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Scaling** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Scaling** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 In the **Manual Scaling** area on the **Scaling** page, configure a manual scaling policy based on the component deployment mode by referring to the following table:

Deployment Mode	Procedure
Container-based deployment	1. Click and change the number of instances. 2. Click . The configuration takes effect when the component status changes from **Scaling** to **Running**.
VM-based deployment	1. Click . 2. Select **Scale out** or **Scale in** for **Type** and add or delete component running instances based on the site requirements. If **Instances** is **1**, only **Scale out** can be selected. – Scale out Select the available server instances that have been managed in the current component running environment. Click **Add ECS** and create an ECS to run new component instances. For details, see **Purchasing a Custom ECS**. Only 64-bit Linux hosts in the **Running** state are supported. Ensure that the following inbound rule has been added to the security group of the host node. For details, see **Adding a Security Group Rule**. ▪ **Action**: **Allow** ▪ **Type**: **IPv4** ▪ **Protocol & Port**: **TCP**, **30000-32767** ▪ **Source**: **0.0.0.0/0** – Scale in Select the target component instance. The number of running component instances can be reduced to 1. You can determine the target component instance based on the server where the component instance is located. 3. Click **OK**.

----End

5.10.3 Managing HPA Policies

In Kubernetes, a HorizontalPodAutoscaler is a built-in component that provides horizontal pod autoscaling (HPA). It supports the application-level cooldown time window and scaling threshold functions based on the Kubernetes HPA.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Restrictions

- You can set the HPA policy only for the components created and deployed using the following methods:
  - **Creating a Component Based on a Container Using Manual Configurations**
  - **Creating a Component Based on a Container Using YAML Configurations**
  - **Creating a Component by Importing a CCE Workload**
- **Workload Type** of the component is **Stateless**.
- The component status is **Running** or **Not ready**. For details about how to check the component status, see **Viewing Component Details**.
- The version of the CCE cluster bound to the component must be 1.15 or later.

Creating an HPA Policy

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Scaling** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Scaling** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Scaling** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 Click next to **Auto Scaling** to enable auto scaling policy configuration. The **Create AS Policy** page is displayed.

- If metrics-server has not been installed in the CCE cluster, go to **Step 5**.
- If metrics-server has been installed in the CCE cluster, go to **Step 8**.

Step 5 Click **Install one** to install metrics-server by referring to **Installing Metrics Server**.

Wait until the plug-in is installed. If **Status** is **Running**, the plug-in is installed successfully.

Step 6 Return to the **Create AS Policy** page.

Step 7 Click **Refresh** to refresh the page.

Step 8 Configure the HPA policy.

1. Policy Name

   Enter a policy name. After the policy is created, its name cannot be changed.

   Enter 4 to 63 characters. Start with a lowercase letter and end with a lowercase letter or digit. Only use lowercase letters, digits, and hyphens (-).
2. Cooldown Period

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

This parameter is supported only in clusters of v1.15 to v1.23.

Enter a scale-out/scale-in cooldown period, in minute. The minimum period is 1 minute.

This parameter indicates the interval between consecutive scaling operations. The cooldown period ensures that a scaling operation is initiated only when the previous one is completed and the system is running stably.

3. Instances

   Enter the minimum and maximum numbers of pods. The minimum number is an integer ranging from 1 to 299. The maximum number is an integer ranging from 1 to 1500 and must be greater than the minimum number.

   After the policy is triggered, the workload pods are scaled within this range.

4. Scaling Behavior

   This parameter is supported only in clusters of v1.25 or later.

   - **Default**: scales workloads using the **Kubernetes default behavior**.
   - **Custom**: scales workloads using custom policies such as stabilization window, steps, and priorities. For parameters that are not set, the **default values recommended by the Kubernetes community** are used.

     - **Disable scale-out/scale-in**: Select whether to disable scale-out or scale-in.
     - **Stabilization Window**: a period during which CCE continuously checks whether the metrics used for scaling keep fluctuating. CCE triggers scaling if the desired state is not maintained for the entire window. This window restricts the unwanted flapping of pod count due to metric changes.
     - **Step**: scaling step. You can set the number or percentage of pods to be scaled in or out within a specified period. If there are multiple policies, you can select the policy that maximizes or minimizes the number of pods.

5. Triggering Condition

   You can configure trigger condition on the UI or by editing the YAML file.

   Before setting trigger conditions of **CPU usage** or **Memory usage**, you need to set the CPU quota (**CPU**) or memory quota (**Memory**) for the component.

   To modify the quotas, click **Increase quota**, set component resource parameters, and upgrade the component. You can customize **CPU** and **Memory** to set their quota, and set the maximum/minimum number of CPU cores and memory size (GiB) that can be used by components. In this way, you can set the number of resources required by each component instance. For details about how to upgrade a component, see **Upgrading a Component in Single-batch Release Mode**.

   - UI

     - **Metric**: Select **CPU usage** or **Memory usage**.

       NOTE

       Usage = Average resource usage of all pods in a workload/Requested resources

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Desired Value**: desired average resource usage. Number of pods to be scaled (rounded up) = (Current metric value/Desired value) x Number of current pods

  NOTE

  When calculating the number of pods to be added or reduced, the HPA policy uses the maximum number of pods in the last 5 minutes.

- **Threshold**: Scaling is not triggered when the desired value is within the range. The desired value must be within the threshold range. If the metric value is greater than the scale-in threshold and less than the scale-out threshold, no scaling is triggered.

  This parameter is supported only in clusters of v1.15 or later.

– YAML

To configure custom metric parameters by using **YAML**, ensure that the prometheus add-on has been installed for the CCE cluster.

Install the prometheus add-on for the CCE cluster. For details, see **prometheus**.

```
metrics:
  - type: Resource
    resource:
      name: cpu
      target:
        type: Utilization
        averageUtilization: 50
  - type: Resource
    resource:
      name: memory
      target:
        type: Utilization
        averageUtilization: 50
  - type: Pods
    pods:
      metric:
        name: packets-per-second
      target:
        type: AverageValue
        averageValue: 1k
  - type: Object
    object:
      metric:
        name: requests-per-second
      describedObject:
        apiVersion: networking.k8s.io/v1beta1
        kind: Ingress
        name: main-route
      target:
        type: Value
        value: 10k
```

As shown in the preceding example, in addition to using the CPU and memory usage as metrics, you can use the YAML format to customize metric parameters and support more metrics such as pods, object, and external.

Step 9 Click **Create Now**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Modifying an HPA Policy

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Scaling** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Scaling** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Scaling** in the left navigation pane.

Step 3 On the **Scaling** page, choose **Policy**, click **Edit**, and reconfigure parameters.

1. Cooldown Period

   This parameter is supported only in clusters of v1.15 to v1.23.

   Enter a scale-out/scale-in cooldown period, in minute. The minimum period is 1 minute.

   This parameter indicates the interval between consecutive scaling operations. The cooldown period ensures that a scaling operation is initiated only when the previous one is completed and the system is running stably.

2. Instances

   Enter the minimum and maximum numbers of pods. The minimum number is an integer ranging from 1 to 299. The maximum number is an integer ranging from 1 to 1500 and must be greater than the minimum number.

   After the policy is triggered, the workload pods are scaled within this range.

3. Scaling Behavior

   This parameter is supported only in clusters of v1.25 or later.

   - **Default**: scales workloads using the **Kubernetes default behavior**.
   - **Custom**: scales workloads using custom policies such as stabilization window, steps, and priorities. For parameters that are not set, the **default values recommended by the Kubernetes community** are used.

     - **Disable scale-out/scale-in**: Select whether to disable scale-out or scale-in.
     - **Stabilization Window**: a period during which CCE continuously checks whether the metrics used for scaling keep fluctuating. CCE triggers scaling if the desired state is not maintained for the entire window. This window restricts the unwanted flapping of pod count due to metric changes.
     - **Step**: scaling step. You can set the number or percentage of pods to be scaled in or out within a specified period. If there are multiple policies, you can select the policy that maximizes or minimizes the number of pods.

4. Triggering Condition

   You can configure trigger condition on the UI or by editing the YAML file.

   Before setting trigger conditions of **CPU usage** or **Memory usage**, you need to set the CPU quota (**CPU**) or memory quota (**Memory**) for the component.

   To modify the quotas, click **Increase quota**, set component resource parameters, and upgrade the component. You can customize **CPU** and

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Memory to set their quota, and set the maximum/minimum number of CPU cores and memory size (GiB) that can be used by components. In this way, you can set the number of resources required by each component instance. For details about how to upgrade a component, see **Upgrading a Component in Single-batch Release Mode**.

- UI
  - **Metric**: Select **CPU usage** or **Memory usage**.

    NOTE

    Usage = Average resource usage of all pods in a workload/Requested resources

  - **Desired Value**: desired average resource usage. Number of pods to be scaled (rounded up) = (Current metric value/Desired value) x Number of current pods

    NOTE

    When calculating the number of pods to be added or reduced, the HPA policy uses the maximum number of pods in the last 5 minutes.

  - **Threshold**: Scaling is not triggered when the desired value is within the range. The desired value must be within the threshold range. If the metric value is greater than the scale-in threshold and less than the scale-out threshold, no scaling is triggered.

    This parameter is supported only in clusters of v1.15 or later.
- YAML

  To configure custom metric parameters by using **YAML**, ensure that the prometheus add-on has been installed for the CCE cluster.

  Install the prometheus add-on for the CCE cluster. For details, see **prometheus**.

```
metrics:
  - type: Resource
    resource:
      name: cpu
      target:
        type: Utilization
        averageUtilization: 50
  - type: Resource
    resource:
      name: memory
      target:
        type: Utilization
        averageUtilization: 50
  - type: Pods
    pods:
      metric:
        name: packets-per-second
      target:
        type: AverageValue
        averageValue: 1k
  - type: Object
    object:
      metric:
        name: requests-per-second
      describedObject:
        apiVersion: networking.k8s.io/v1beta1
        kind: Ingress
        name: main-route
```

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

```
target:
  type: Value
  value: 10k
```

As shown in the preceding example, in addition to using the CPU and memory usage as metrics, you can use the YAML format to customize metric parameters and support more metrics such as pods, object, and external.

Step 4 Click **OK**.

----End

Viewing the Running Status of an HPA Policy

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Scaling** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Scaling** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Scaling** in the left navigation pane.

Step 3 On the **Scaling** page:

- Click the **Status** tab to view the policy running status.
- Click the **Event** tab to view events that occur during policy running.

----End

Disabling an HPA Policy

After the HPA policy is disabled, all data will be cleared. Exercise caution when performing this operation.

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Scaling** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Scaling** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Scaling** in the left navigation pane.

Step 3 On the **Scaling** page, click on the right of **Auto Scaling**.

Step 4 Click **OK**.

----End

5.11 Component O&M

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.11.1 Viewing Component Running Metrics

After a component is created and deployed, you can go to its **Metric Monitoring Graphs** page to view the statistics of component running metrics.

Prerequisites

- You have created and deployed a component. For details, see **Creating and Deploying a Component**.
- For container-deployed components, a CCE cluster must be bound to the component deployment environment.

Viewing Component Running Metrics

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Metric Monitoring Graphs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Monitoring Overview** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Monitoring Overview** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

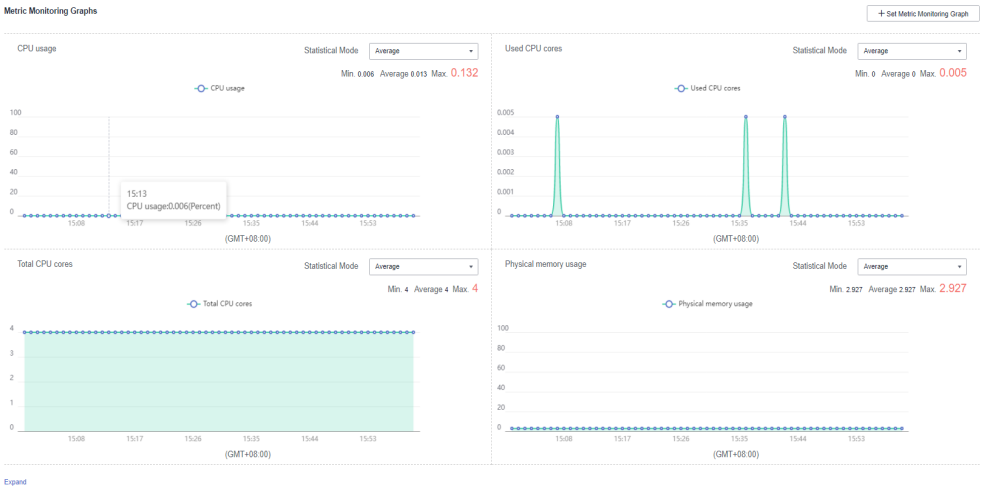
1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 On the **Metric Monitoring Graphs** page, view the statistics of component running metrics in the last hour at an interval of 1 minute.

- Click on the component running metric page for which you want to suspend statistics collection.
- Click on the component running metric page to continue collecting statistics on the running metric.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 5-9 Metric graph



----End

5.11.2 Customizing Component Running Metrics

After creating and deploying a component based on a container, if a CCE cluster is bound to the component deployment environment, you can go to its **Metric Monitoring Graphs** page to customize the component running metrics to be viewed.

Prerequisites

A container-deployed component is available. For details, see:

- **Creating a Component Based on a Container Using Manual Configurations**
- **Creating a Component Based on a Container Using YAML Configurations**
- **Creating a Component by Importing a CCE Workload**
- **Creating a Component Using a Component Template**

Customizing Component Running Metrics

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Metric Monitoring Graphs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Monitoring Overview** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Monitoring Overview** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 On the **Metric Monitoring Graphs** page, click **Set Metric Monitoring Graph**.

- Select the system metrics to be viewed and select **Statistical Mode**.
- Deselect the system metrics that you do not need to view.

  You can click **Clear** next to **Selected Metric** to clear all selected system metrics.

Step 5 Click **OK**.

----End

5.11.3 Managing Component Logs

5.11.3.1 Managing Component AOM Logs

ServiceStage component logs are connected to AOM by default. You can view, search for, and export logs to locate and rectify faults that occur during component running.

For container-deployed components, component running logs can be connected to AOM only if a CCE cluster is bound to the component deployment environment.

Managing Component AOM Logs

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Logs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Logs** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Logs** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 On the **Running Logs(AOM)** page, manage component running logs by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
View Logs	1. Select the instance whose logs you want to view. 2. Select the log file you want to view. 3. Select the time range of the logs you want to view. If you select **Custom**, the time range cannot exceed 30 days. 4. Enter keywords in the search box and click to view details about the specified logs. If you cannot view logs on the ServiceStage **Logs** page, see **Why Can't I View ServiceStage Logs?**
Exporting logs	1. Click **Last Records**. 2. Select the number of log records to be exported. 3. Open the **log.txt** file exported locally and view the exported log records.
View AOM Logs	Click **View AOM Logs**. You can view the component running logs on the AOM console. For details, see **Viewing Log Files**.

----End

5.11.3.2 Managing Component LTS Logs

5.11.3.2.1 LTS Log Overview

Log Tank Service (LTS) collects log data from hosts and cloud services. By processing massive amounts of logs efficiently, securely, and in real time, LTS provides useful insights for you to optimize the availability and performance of cloud services and applications. It also helps you efficiently perform real-time decision-making, device O&M, and service trend analysis.

ServiceStage allows you to interconnect with LTS to view, search for, and export LTS logs. You can also view container logs to locate and rectify faults that occur during component running.

For container-deployed components, component running logs can be connected to LTS only if a CCE cluster is bound to the component deployment environment.

5.11.3.2.2 Configuring an LTS Log Group

After configuring LTS logs for components that are not interconnected with LTS, you can view the component running logs collected by LTS on the ServiceStage console and query the logs.

Restrictions

If you want to collect the run logs of the container-deployed component to LTS for processing, there are the following restrictions:

- CCE cluster nodes whose container engine is Docker are supported.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- CCE cluster nodes whose container engine is Containerd are supported. You must be using ICAgent 5.12.130 or later.
- To collect container log directories mounted to host directories to LTS, you must configure the node file path.
- Currently, container file log collection supports only the Overlay2 storage driver, not Device Mapper. Run the following command to check the storage driver type:
  ```
  docker info | grep "Storage Driver"
  ```

Configuring an LTS Log Group

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Logs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Logs** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Logs** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 On the **Running Logs(LTS)** page, click **Configure Now**.

Step 5 Select a configuration method by referring to the following table.

Configuration Method	Description
Associate an existing LTS log group	Select the log group to be associated from the **Log Group** drop-down list. • You have created a log group. For details, see **Creating a Log Group**. • You have created a log stream. For details, see **Creating a Log Stream**. • The path of the logs to be collected has been configured in the log stream. – If the access mode of container-deployed components is **CCE (Cloud Container Engine)**, see **Ingesting CCE Application Logs to LTS**. – If the access mode of container-deployed components is **ServiceStage - Containerized Application Logs**, see **Ingesting ServiceStage Containerized Application Logs to LTS**. – If the access mode is **ServiceStage - Cloud Host Logs**, see **Ingesting ServiceStage Cloud Host Logs to LTS**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuration Method	Description
Automatically created and associated	• After a container-deployed component is configured and deployed, the system automatically generates log group and log stream configurations, and generates access configurations and host groups on the **Ingestion Management** page of the LTS console. – **Log Group**: The naming rule is **servicestage-***{Application ID}*. Use the ID of the application to which the component belongs. – **Log Stream**: The naming rule is **lts-topic-***{Component name}*. Use the name of the component connected to LTS. – **Ingestion**: The naming rule is **access-config-***{Component name}*. Use the name of the component connected to LTS. – **Host Group**: The naming rule is **k8s-log-***{CCE cluster ID}*. Use the ID of the CCE cluster where the component connected to LTS is located. • After a VM-deployed component is configured and deployed, the system automatically generates log group and log stream configurations, and generates access configurations and host groups on the **Ingestion Management** page of the LTS console. – **Log Group**: The naming rule is **servicestage-***{Application ID}*. Use the ID of the application to which the component belongs. – **Log Stream**: The naming rule is **lts-topic-***{Component name}*. Use the name of the component connected to LTS. – **Ingestion**: The naming rule is **access-config-***{Component name}*. Use the name of the component connected to LTS. – **Host Group**: The naming rule is **vm-***{Instance ID}*. Use the ID of the component instance connected to LTS.

Step 6 Click **OK**.

After the log group is configured, you can view the component running logs collected by LTS on the **Running Logs(LTS)** tab.

Click **Go To** or to go to the log stream page on the LTS console. You can manage component run logs. For details, see **Log Management**.

----End

5.11.3.2.3 Searching for Running Logs

After associating a component with an LTS log group, you can set a keyword and time range to search for logs.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

The component has been associated with a log group. For details, see **Configuring an LTS Log Group**.

Searching for Running Logs

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Logs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Logs** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Logs** in the left navigation pane.

Step 3 On the **Running Logs(LTS)** page, select a log stream from the drop-down list.

Step 4 Select a time range in the upper right corner.

- **From now**: queries log data generated in a time range that ends with the current time, such as the previous 1, 5, or 15 minutes. For example, if the current time is 19:20:31 and **1 hour** is selected as the relative time from now, the charts on the dashboard display the log data that is generated from 18:20:31 to 19:20:31.
- **From last**: queries log data generated in a time range that ends with the current time, such as the previous 1 or 15 minutes. For example, if the current time is 19:20:31 and **1 hour** is selected as the relative time from last, the charts on the dashboard display the log data that is generated from 18:00:00 to 19:00:00.
- **Specified**: queries log data that is generated in a specified time range.

Step 5 On the log details page, you can search for logs using the following methods:

- Enter a keyword in the search box and click **Search**, or select a historical record, index field, or keyword from the displayed drop-down list.
- On the **Raw Logs** tab, hover the cursor over a field in the log content, click it, and choose **Copy**, **Add To Search**, or **Exclude from Search** from the displayed menu.
- In the displayed drop-down list, press the up and down arrows on the keyboard to select a keyword or search syntax, press **Tab** or **Enter**, and click **Search**.

----End

5.11.3.2.4 Quickly Querying Logs

To search for logs using a keyword repeatedly, perform the following operations to configure quick query.

Prerequisites

The component has been associated with a log group. For details, see **Configuring an LTS Log Group**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Quickly Querying Logs

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Running Logs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Logs** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Logs** in the left navigation pane.

Step 3 On the **Running Logs(LTS)** page, select a log stream from the drop-down list.

Step 4 Click .

Step 5 Enter a name and keyword, and click **OK**.

----End

5.11.3.2.5 Using Visualization to Analyze Logs

Visualization support SQL query and analysis for structured log fields. After log structuring, wait about 1–2 minutes for SQL query and analysis.

Prerequisites

- The log stream has been structured. For details, see **Log Structuring**
- The component has been associated with a log group. For details, see **Configuring an LTS Log Group**.

Using Visualization to Analyze Logs

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Logs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Logs** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Logs** in the left navigation pane.

Step 3 On the **Running Logs(LTS)** page, select a log stream from the drop-down list.

Step 4 Select a time range.

- **From now**: queries log data generated in a time range that ends with the current time, such as the previous 1, 5, or 15 minutes. For example, if the current time is 19:20:31 and **1 hour** is selected as the relative time from now, the charts on the dashboard display the log data that is generated from 18:20:31 to 19:20:31.
- **From last**: queries log data generated in a time range that ends with the current time, such as the previous 1 or 15 minutes. For example, if the current time is 19:20:31 and **1 hour** is selected as the relative time from last, the charts on the dashboard display the log data that is generated from 18:00:00 to 19:00:00.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Specified**: queries log data that is generated in a specified time range.

Step 5 Click the **Visualization** tab.

Step 6 Click the SQL query box and enter the SQL query statement.

According to the data returned by the SQL query, select a chart type to display the query result. For details, see **LTS Visualization**.

----End

5.11.3.2.6 Viewing Real-Time Logs

After associating a component with an LTS log group, you can view the component logs reported in real time.

Prerequisites

The component has been associated with a log group. For details, see **Configuring an LTS Log Group**.

Viewing Real-Time Logs

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Logs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Logs** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Logs** in the left navigation pane.

Step 3 On the **Running Logs(LTS)** page, select a log stream from the drop-down list.

Step 4 Click the **Real-Time Logs** tab and view the real-time logs in the **Log Content** area.

One log record is reported every one minute. You can control log display by clicking **Clear** or **Pause** in the upper right corner.

- **Clear**: clears all displayed logs.
- **Pause**: pauses the real-time log display.

  After you click **Pause**, the button changes to **Continue**. You can click **Continue** to resume the log display.
- If you are viewing real-time logs, do not switch to another page. Otherwise, the real-time viewing function will be disabled. After this function is enabled again, the real-time logs that are viewed last time will not be displayed.

----End

5.11.3.2.7 Unbinding an LTS Log Group

You can unbind an LTS log group if it is no longer used.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

The component has been associated with a log group. For details, see **Configuring an LTS Log Group**.

Unbinding an LTS Log Group

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Logs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Logs** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Logs** in the left navigation pane.

Step 3 On the **Running Logs(LTS)** page, select the target LTS log group from the **Log Group** drop-down list and click **Unbind**.

Step 4 Click **OK**.

After the unbinding, you cannot view LTS logs on the **Running Logs(LTS)** page.

----End

5.11.4 Viewing Container Logs

ServiceStage allows you to view container logs of container-deployed components to locate and rectify faults that occur during component running.

Viewing Container Logs

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Logs** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Logs** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Logs** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 On the **Container Logs** page, view the component container logs.

- Select the target instance from the **Instance** drop-down list.
- Specify **Lines** to configure how many rows you can view.

  You can view up to 100, 200, 500, 5000, or 10,000 lines.
- Click **Download** to download the logs.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.11.5 Configuring Alarm Thresholds for Resource Monitoring

When a component is deployed based on a container and a CCE cluster is bound to the component deployment environment, if you need to monitor some resources and respond to exceptions in a timely manner, you can create threshold rules for metrics of these key resources, so that you can find and handle exceptions in time.

- If the metric meets the threshold conditions within a specified period, the system sends a threshold alarm.
- If no metric is reported within a specified period, the system sends a data insufficiency event.
- If you cannot query the change information about the threshold rule status on the ServiceStage console, you can enable the notification function to send the change information to related personnel through SMS messages or emails.

Configuring Alarm Thresholds for Resource Monitoring

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Threshold Alarms** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Threshold Alarms** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Threshold Alarms** in the left navigation pane.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 Click **Set Threshold Rule** and set threshold rule parameters by referring to **Table 5-8**. Parameters marked with an asterisk (*) are mandatory.

Table 5-8 Threshold rule parameters

Parameter	Description
*Threshold Name	Name of a threshold rule. Enter up 255 characters. The following special characters are not allowed: $#%&;<=> The name must be unique and cannot be modified once specified.
Description	Description of an alarm threshold rule. Enter up to 255 characters.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Statistic Method	Method used to measure the metrics. • Average • Minimum • Maximum • Sum • Samples
Statistical Periods	Interval at which metric data is collected. • 1 minute • 5 minute • 15 minute • 1 hour
Metric	Metrics to be monitored. • Total CPU cores • Used CPU cores • CPU usage • Handles • Total physical memory • Physical memory usage • Used physical memory • Status • Threads • Total virtual memory
*Threshold Condition	Trigger of a threshold alarm. A threshold condition consists of two parts: operators (≥, ≤, >, and <) and threshold value. For example, if this parameter is set to **≥ 80**, the system generates a threshold alarm when the metric is greater than or equal to 80.
Consecutive Periods	When the metric meets the threshold condition for a specified number of consecutive periods, a threshold alarm will be generated.
Alarm Severity	Severity of a threshold alarm. Options: **Critical**, **Major**, **Minor**, and **Warning**.
Send Notifications	Whether to send notifications. • If you select **Yes** (recommended), SMN will send notifications to you when a threshold alarm is triggered and you will be billed for the SMN service. • If you select **No**, no notifications will be sent to you.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Topic Name	If **Send Notification** is set to **Yes**, select a created topic. A topic is a specified event to publish messages and subscribe to notifications. It serves as a message transmission channel between publishers and subscribers. For details about how to create a topic, see **Creating a Topic**.
*Trigger Condition	Trigger condition for sending a notification when **Send Notification** is set to **Yes**. • **An alarm occurred**: When a threshold alarm is generated, the system sends a notification to a specified user by email or SMS message. • **The alarm is cleared**: When the alarm is cleared, the system sends a notification to a specified user by email or SMS message.

Step 5 Click **OK**.

----End

Follow-up Operations

After a threshold rule is created, you can manage threshold alarms by referring to **Table 5-9**.

Table 5-9 Operations related to threshold alarm management

Operation	Description
Modify a Threshold Alarm	When you find that the current threshold rule is not properly set, you can perform the following operations to modify the threshold rule to better meet your service requirements. 1. Click **Modify** in the **Operation** column of the threshold alarm list. 2. On the **Modify Threshold Rule** page, modify the parameters of the threshold rule as prompted. 3. Click **Modify**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Delete a Threshold Alarm	When you find that the current threshold rule is no longer needed, you can perform the following operations to delete the threshold rule to release more threshold rule resources. 1. Delete one or multiple threshold rules. • To delete a single threshold, click **Delete** in the **Operation** column of the threshold rule list. • To delete threshold rules in bathes, select one or more threshold rules and click **Delete** on the upper part of the page. 2. In the displayed dialog box, click **OK**.
Search for Threshold Alarms	1. Select a time segment from the drop-down list. 2. Enter the keyword of the alarm name or description in the search box on the upper right corner of the page. 3. Click or press **Enter**. Alternatively, click **Advanced Search**, set the search criteria, and click **Search**.
View Threshold-Crossing Alarms	If the metric meets the threshold conditions within a specified period, the system sends a threshold alarm. View the alarm in the threshold alarm list.
View Alarm History	Click **History** in the **Operation** column of the threshold rule list to view historical alarms.
Check the data insufficiency event.	If no metric is reported within a specified period, the system sends a data insufficiency event. You can view the event on the **Event** page. For details, see **Viewing Component Running Events**.

5.11.6 Viewing Component Running Events

For container-deployed components, you can view events that occur during component running to locate and rectify faults that occur during component running.

Viewing Component Running Events

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Events** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **O&M Configurations** > **Events** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **O&M Configurations** > **Events** in the left navigation pane.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 If the component is created by **Creating a Component by Importing a CCE Workload** and is not enabled:

1. In the displayed dialog box, click **OK**.
2. On the **Enable Component** page, click **Enable** and wait until the component is restarted.

Step 4 On the **Events** page, view the component running events.

- You can select the query time to view the component running events within a specified time range.
- You can enter an event keyword to search for and view specific component running events.

----End

5.12 Managing Cloud Service Settings of a Component

5.12.1 Binding a Microservice Engine

Cloud Service Engine is a one-stop management platform for microservice solutions. After microservice components are connected to CSE, you can use service registry, service governance, and configuration management of CSE. In this way, you can focus on service development and improve product delivery efficiency and quality. You can bind a microservice engine when configuring **Cloud Service Settings**.

NOTE

After a component developed based on ServiceComb 2.7.8 or later or Spring Cloud Huawei 1.10.4-2021.0.x or later is connected to a microservice engine, the following attributes are injected into the **properties** parameter of the **MicroServiceInstance** parameter when a microservice instance is created in the microservice engine:

1. **CAS_APPLICATION_ID**: ID of the application to which the component belongs.
2. **CAS_COMPONENT_NAME**: component name.
3. **CAS_ENVIRONMENT_ID**: ID of the component deployment environment.
4. **CAS_INSTANCE_ID**: component instance ID.
5. **CAS_INSTANCE_VERSION**: component instance version.

For details about the **MicroServiceInstance** parameter, see **MicroServiceInstance**.

Prerequisites

- Microservice engines (ServiceComb engines or registry/configuration centers) have been managed in the environment where the component is deployed. For details, see **Managing Basic Resources**.
- For container-deployed components, a CCE cluster must be bound to the component deployment environment.

Binding a Microservice Engine

Step 1 Choose **Cloud Service Settings** > **Microservice Engine**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 2 Click **Bind Microservice Engine**.

Step 3 Select a microservice engine instance that has been bound in the environment.

If no microservice engine instance is bound to the environment, click **Add one** and add a microservice engine instance to the environment. For details, see **Managing Basic Resources**.

- Microservice engines (ServiceComb engines and registry/configuration centers) can be bound to VM-deployed components with a Java or Tomcat technology stack.
- ServiceComb engines can be bound to container-deployed components with a Java, Tomcat, Python, Php, Node.js, or Docker technology stack. Registry/configuration centers can be bound to container-deployed components with a Java, Tomcat, or Docker technology stack.

Step 4 Click **OK**.

Move the cursor to a bound microservice engine and perform the following operations:

- Rebind a microservice engine: Click , select the target microservice engine again, and click **OK**.
- Delete a bound microservice engine: Click .

Step 5 (Optional) For container-deployed components, specify **Addons** based on the engine bound to the component.

Engine Type	Plug-in Type	Description
ServiceComb engine	Mesher	Enter the listening port number of the application process to enable multi-language access to service mesh. You can use Mesher to connect components that are not developed in the microservice framework to the ServiceComb engine. • For non-microservice components developed using Java, Tomcat, or Docker, you can enable Mesher and use Mesher to connect the components to the ServiceComb engine for microservice registry and discovery. • For components developed using Python, PHP, or Node.js, forcibly enable Mesher and connect the components to the ServiceComb engine for microservice registry and discovery.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Engine Type	Plug-in Type	Description
	Sermant Injector	Sermant Agent can be automatically mounted using the Sermant Injector plug-in installed in CCE in the component deployment environment. In this way, components can be connected to the ServiceComb engine though Sermant Agent. For details about how to install the Sermant Injector plug-in, see **Installing Sermant Injector**. • **Enable Dynamic Configuration** is selected by default. If you use the dynamic configuration provided by Sermant, the dynamic configuration of the bound ServiceComb engine injected by the microservice component does not take effect. If it is not selected, the dynamic configuration of the bound ServiceComb engine injected by the microservice component is used. • By default, **Spring Cloud** and **Dubbo** are selected for **Enable Registry Center** so that Spring Cloud and Dubbo microservice components can use the registry capability provided by Sermant, and **Engine** is the ServiceComb engine that you select. The injected registry configuration of the bound ServiceComb engine does not take effect. • If only **Spring Cloud** is selected for **Enable Registry Center**, Spring Cloud microservice components can use the registry capability provided by Sermant, and **Engine** is the ServiceComb engine that you select. Dubbo microservice components use the injected registry configuration of the bound ServiceComb engine. • If only **Dubbo** is selected for **Enable Registry Center**, Dubbo microservice components can use the registry capability provided by Sermant, and **Engine** is the ServiceComb engine that you select. Spring Cloud microservice components use the injected registry configuration of the bound ServiceComb engine. • If **Spring Cloud** and **Dubbo** are not selected for **Enable Registry Center**, microservice components use the injected registry configuration of the bound ServiceComb engine. • If **Spring Cloud** or **Dubbo** is selected for **Enable Registry Center**, the ServiceComb engine bound to the component is used as the registry center engine by default.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Engine Type	Plug-in Type	Description
		To change the registry center engine, move the cursor to the bound registry center engine, click , select another ServiceComb engine that has been managed in the component deployment environment, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Engine Type	Plug-in Type	Description
Registry/ Configuration center	Sermant Injector	Sermant Agent can be automatically mounted using the Sermant Injector plug-in installed in CCE in the component deployment environment. In this way, components can be connected to the registry/ configuration center though Sermant Agent. For details about how to install the Sermant Injector plug-in, see **Installing Sermant Injector**. • **Enable Dynamic Configuration** is selected by default. If you use the dynamic configuration provided by Sermant, the dynamic configuration of the bound registry/configuration center injected by the microservice component does not take effect. If it is not selected, the dynamic configuration of the bound registry/configuration center injected by the microservice component is used. • By default, **Spring Cloud** and **Dubbo** are selected for **Enable Registry Center** so that Spring Cloud and Dubbo microservice components can use the registry capability provided by Sermant, and **Engine** is the ServiceComb engine or registry/configuration center you select. The injected registry configuration of the bound registry/configuration center does not take effect. • If only **Spring Cloud** is selected for **Enable Registry Center**, Spring Cloud microservice components can use the registry capability provided by Sermant, and **Engine** is the registry/ configuration center or ServiceComb engine you select. Dubbo microservice components use the injected registry configuration of the bound registry/configuration center. • If only **Dubbo** is selected for **Enable Registry Center**, Dubbo microservice components can use the registry capability provided by Sermant, and **Engine** is the registry/configuration center or ServiceComb engine you select. Spring Cloud microservice components use the injected registry configuration of the bound registry/ configuration center. • If **Spring Cloud** and **Dubbo** are not selected for **Enable Registry Center**, microservice components use the injected registry configuration of the bound registry/configuration center. • If **Spring Cloud** or **Dubbo** is selected for **Enable Registry Center**, the registry/configuration

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Engine Type	Plug-in Type	Description
		center bound to the component is used as the registry center engine by default. To change the registry center engine, move the cursor to the bound registry center engine, click , select another registry/configuration center or ServiceComb engine that has been managed in the component deployment environment, and click **OK**. If a component is bound to a registry/configuration center and the Sermant Injector plug-in is selected, dark launch is not supported. For details, see **Upgrading a Component in Dark Launch Mode**.

----End

5.12.2 Binding a Distributed Cache

Traditional single-instance applications use local session management. Session contexts generated by user requests are stored in the process memory. After the load balancing module is added, multi-instance sessions need to be shared using distributed storage.

ServiceStage provides the out-of-the-box distributed session function. It uses the **Distributed Cache Service** as the session persistence layer. Without code modification, ServiceStage supports distributed session management for Tomcat applications, Node.js applications that use express-session, and PHP applications that use session handle.

You can bind a distributed cache instance when configuring **Cloud Service Settings**. After binding, you can read environment variables upon application running to obtain information about the distributed cache. For details, see **Common Environment Variables**.

Prerequisites

- A distributed cache has been created. For details, see **Buying a DCS Redis Instance**.
- For container-deployed components, a CCE cluster must be bound to the component deployment environment.

Binding a Distributed Cache

Step 1 Choose **Cloud Service Settings** > **Distributed Cache Service**.

Step 2 Click **Bind Distributed Cache**.

Step 3 Select a distributed cache instance that has been bound in the environment.

If no distributed cache instance is bound to the environment, click **Add one** and add a distributed cache instance to the environment. For details, see **Managing Basic Resources**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 If the DCS instance must be accessed using a password, enter the access password.

Step 5 Click **OK**.

Move the cursor to a bound distributed cache and perform the following operations:

- Rebind a distributed cache: Click and select the target distributed cache again. If the selected DCS instance access mode is password access, you need to enter the access password. Click **OK**.
- Delete a bound distributed cache: Click .

----End

5.12.3 Binding a Cloud Database

To store application data permanently, you need to use Relational Database Service (RDS). Based on the cloud computing platform, ServiceStage provides RDS for MySQL which is reliable, scalable, easy to manage, and ready for use. **RDS for MySQL** enables you to easily set and scale relational databases on the cloud. Using the RDS service, you can perform nearly all necessary tasks without programming. This service simplifies operation procedures and reduces routine O&M workloads, so that you can focus on application and service development.

You can bind a cloud database when configuring **Cloud Service Settings**. After binding, you can read environment variables upon application running to obtain MySQL information. For details, see **Common Environment Variables**.

Prerequisites

- An RDS MySQL DB instance has been created. For details, see **Buying an RDS for MySQL DB Instance**.
- For container-deployed components, a CCE cluster must be bound to the component deployment environment.

Binding a Cloud Database

Step 1 Choose **Cloud Service Settings** > **Relational Database Service**.

Step 2 Click **Bind Cloud Database**.

Step 3 Select a cloud database instance that has been bound in the environment and click **OK**.

If no cloud database instance is bound to the environment, click **Add one** and add a cloud database instance to the environment. For details, see **Managing Basic Resources**.

Step 4 Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Connection Type	Select a connection type. • **JNDI**: standard Java connection mode. You need to enter the JNDI name. • **Spring Cloud Connector**: Spring connection mode.
*Database Name	Enter a database name.
*Database Account	Enter a database account.
*Database Password	Enter the database password.

Step 5 Click **OK**.

Move the cursor to an RDS instance and perform the following operations:

- Rebind an RDS instance: Click and select the target RDS instance again.
- Delete a bound RDS instance: Click .

----End

5.12.4 Binding a GaussDB Instance

GaussDB is required for persistent storage of application data. ServiceStage allows you to bind GaussDB to components. GaussDB features high availability, reliability, and security, auto scaling, one-click deployment, quick backup and restoration, and monitoring and alarm reporting. You can use **GaussDB** to meet your daily work requirements, such as development, O&M, optimization, monitoring, and migration.

You can bind a GaussDB instance when configuring **Cloud Service Settings**.

Prerequisites

- You have created a GaussDB instance. For details, see **Buying a GaussDB Instance**.
- For container-deployed components, a CCE cluster must be bound to the component deployment environment.

Binding a GaussDB Instance

Step 1 Choose **Cloud Service Settings** > **GaussDB**.

Step 2 Click **Bind GaussDB**.

Step 3 Select a GaussDB instance that has been bound in the environment.

If no GaussDB instance is bound to the environment, click **Add One** and add a GaussDB instance to the environment. For details, see **Managing Basic Resources**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 Set parameters by referring to the following table.

Parameter	Description
Database Name	Enter a database name.
Database User Name	Enter the account for accessing the database.
Database Login Password	Enter the password for accessing the database.

Step 5 Click **OK**.

Move the cursor to a GaussDB instance and perform the following operations:

- Rebind a GaussDB instance: Click and select the target GaussDB instance again.
- Delete a bound GaussDB instance: Click .

----End

5.12.5 Binding a DMS for RocketMQ Instance

DMS for RocketMQ is required for application components to store and consume messages. ServiceStage allows you to bind components to DMS for RocketMQ, a message middleware based on the cloud computing platform. DMS for RocketMQ features low latency, high reliability, high throughput, dynamic expansion, and convenience. DMS for RocketMQ allows you to easily build message queues that are fully managed, scalable, reliable, scalable, and easy to manage.

You can bind a DMS for RocketMQ instance when configuring **Cloud Service Settings**.

Prerequisites

- You have created a DMS for RocketMQ instance. For details, see **Buying a RocketMQ Instance**.
- For container-deployed components, a CCE cluster must be bound to the component deployment environment.

Binding a DMS for RocketMQ Instance

Step 1 Choose **Cloud Service Settings** > **Distributed Message Service for RocketMQ**.

Step 2 Click **Bind RocketMQ**.

Step 3 Select a DMS for RocketMQ instance that has been bound in the environment.

If no DMS for RocketMQ instance is bound to the environment, click **Add One** and add an instance to the environment. For details, see **Managing Basic Resources**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 Set parameters by referring to the following table.

Parameter	Description
Username	Enter the username.
Password	Enter the password of the login user.

Step 5 Click **OK**.

Move the cursor to a bound DMS for RocketMQ instance and perform the following operations:

- Rebind a DMS for RocketMQ: Click and select the target instance again.
- Delete a bound DMS for RocketMQ: Click .

----End

5.12.6 Binding a CSS Instance

Cloud Search Service (CSS) is an online distributed search service. You can use the multi-condition search, statistics, and report capabilities of its structured and unstructured text to analyze component logs in all scenarios.

You can bind a CSS instance when configuring **Cloud Service Settings**.

Prerequisites

- You have created an Elasticsearch cluster. For details, see **Creating an Elasticsearch Cluster**.
- For container-deployed components, a CCE cluster must be bound to the component deployment environment.

Binding a CSS Instance

Step 1 Choose **Cloud Service Settings** > **Cloud Search Service**.

Step 2 Click **Bind CSS**.

Step 3 Select the CSS Elasticsearch instance bound to the environment.

If no CSS Elasticsearch instance is bound to the environment, click **Add one** and add a CSS Elasticsearch instance to the environment. For details, see **Managing Basic Resources**.

Step 4 Set parameters by referring to the following table.

Parameter	Description
Username	Enter the username.
Password	Enter the password of the login user.

Step 5 Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Move the cursor to a bound CSS Elasticsearch instance and perform the following operations:

- Rebind a CSS Elasticsearch instance: Click and select the target instance again.
- Delete a bound CSS Elasticsearch instance: Click .

----End

5.13 Managing Container Settings of a Container-Deployed Component

5.13.1 Adding an Environment Variable of a Component

Environment variables of a component are set in the component running environment and provide great flexibility for applications. Environment variables set for a component are local environment variables and take effect only for this component.

If you add an application environment variable to the application where the component is located and the name of the application environment variable is the same as that of the component environment variable in the application, the application environment variable is shielded by the component environment variable and does not take effect for the component. For details about environment variables of an application, see **Managing Environment Variables of an Application**.

Priority of environment variables of the component, application, and environment levels: component > application > environment.

For details about environment variables of an environment, see **Managing Environment Variables of an Environment**.

Adding an Environment Variable of a Component

Step 1 Choose **Container Settings** > **Environment Variable**.

Step 2 Set component environment variables by referring to the following table.

Type	Procedure
Add manually	Enter the environment variable name and parameter value. 1. Click **Add Environment Variable** and select **Add manually**. 2. Set **Name** and **Variable/Variable Reference** to add an environment variable. **Name** must be unique in the same component. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.).

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Type	Procedure
Import from secret	Import the value of a key in a secret as the environment variable value. 1. Create a secret. For details, see **Creating a Secret**. 2. Click **Add Environment Variable** and select **Add from secret**. 3. Set **Name**. **Name** must be unique in the same component. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.). 4. Select a secret key from the **Variable/Variable Reference** drop-down list. After the import, there will be a component environment variable named after the variable name entered in **Step 2.3**. The value is the value of the selected key.
Import from configuration items	Import the value of a key in a configuration item as the environment variable value. 1. Create a configuration item. For details, see **Creating a Configuration Item**. 2. Click **Add Environment Variable** and select **Add from ConfigMap**. 3. Set **Name**. **Name** must be unique in the same component. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.). 4. Select a configuration item key from the **Variable/Variable Reference** drop-down list. After the import, there will be a component environment variable named after the variable name entered in **Step 2.3**. The value is the value of the selected key.
Import from a file	Click **Import** and select a local configuration file. The file must be a key-value pair mapping file in JSON or YAML format. A maximum of 200 environment variables can be imported at a time. Example: `{"key1": "value1", "key2": "value2"}` • **key1** and **key2** are the component environment variable names. The variable names must be unique in a component. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.). • **value1** and **value2** are the component environment variable values.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Type	Procedure
Resource reference	1. Click **Add Environment Variable** and select **Resource reference**. 2. Set **Name**. **Name** must be unique in the same component. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.). 3. Select the resource application value or limit value defined by the container from the **Variable/Variable Reference** drop-down list as the value of the environment variable.
Variable/ Variable Reference	Use the field defined by the instance as the value of the environment variable. 1. Click **Add Environment Variable** and select **Variable/ Variable Reference**. 2. Set **Name**. **Name** must be unique in the same component. Enter 1 to 64 characters. Start with a letter, underscore (_), or hyphen (-). Only use letters, digits, underscores (_), hyphens (-), and periods (.). 3. In the **Variable/Variable Reference** drop-down list, select the field defined by the instance as the value of the environment variable.

----End

5.13.2 Configuring the Startup Command and Lifecycle of a Component

For container-deployed components, ServiceStage provides callback functions for the lifecycle management during and after application component startup. For example, if you want an application component to perform a certain operation before stopping, you can register a hook function.

Lifecycle Callback Functions

ServiceStage provides the following lifecycle callback functions:

- Startup command: corresponds to the **ENTRYPOINT** startup command of Docker, and is used to start an application component.
- Post-start processing: triggered after an application component is started.
- Pre-stop processing: triggered before an application component is stopped (Terminated), not when an application component exits (Completed) due to execution completion.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuring the Startup Command and Lifecycle of a Component

Step 1 Choose **Container Settings**.

Step 2 Click **Startup Command** to set **Command** and **Parameter** for the container.

A Docker image has metadata that stores image information. If no **Lifecycle** command or parameter is set, the container runs the default command and parameter provided during image creation. The Docker defines the default command and parameter as **CMD** and **Entrypoint**. For details about the two fields, see ***Entrypoint Description*** and ***CMD Description***.

If the running command and parameter of the application are set during application component deployment, the default **Entrypoint** and **CMD** will be overwritten during image building. **Table 5-10** describes the rules.

Table 5-10 Startup command parameters

Image Entrypoint	Image CMD	Application Running Command	Application Running Parameter	Final Execution
[touch]	[/root/test]	Not set	Not set	[touch /root/test]
[touch]	[/root/test]	[mkdir]	Not set	[mkdir]
[touch]	[/root/test]	Not set	[/opt/test]	[touch /opt/test]
[touch]	[/root/test]	[mkdir]	[/opt/test]	[mkdir /opt/test]

1. **Command**: Enter an executable command, for example, **/run/server**.

   If there are multiple commands, you are advised to run **/bin/sh** or other shell commands. Other commands are used as parameters.

2. **Parameter**: Enter a parameter that controls the container running command, for example, **--port=8080**.

   If there are multiple parameters, separate them in different lines.

Step 3 Click **Lifecycle** and set **Post-start Processing** and **Pre-stop Processing**.

- Set **Post-start Processing**.

  Table 5-11 describes the **Post-start Processing** parameters. Select either of the processing modes.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-11 Post-start Processing parameters

Processing Mode	Command

CLI mode	Command to be executed in the component instance.
	The command format is **Command** *Args[1] Args[2]*.....
	Command is a system command (bash mode) or a user-defined executable program (binary mode). If no path is specified, the system searches for an executable program in the default path.
	To run multiple commands, write them in different lines or into a script.
	Commands that are executed in the background or asynchronously are not supported.
	For example, the following commands need to be executed:
	`exec:`
	`  command:`
	`  - /install.sh`
	`  - install_agent`
	Write **/install.sh install_agent** in the script. This command indicates that **install.sh** will be executed after the component is deployed.

HTTP request mode	HTTP call request. – **Path**: (optional) URL of a request. Example: **/nginx**. – **Port**: (mandatory) request port. Example: **80**. – **Host Address**: (optional) IP address of the request. The default value is the IP address of the node where the application is located. Example: **127.0.0.1**.

- Set **Pre-stop Processing**.

  Table 5-12 describes the **Pre-stop Processing** parameters. Select either of the processing modes.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-12 Pre-stop Processing parameters

Processing Mode	Command

CLI mode	Command to be executed in the component instance.
	The command format is **Command** *Args[1] Args[2]....*. **Command** is a system command (bash mode) or a user-defined executable program (binary mode). If no path is specified, the system searches for an executable program in the default path.
	To run multiple commands, write them in different lines or into a script.
	For example, the following commands need to be executed:
	`exec:`
	`  command:`
	`  - /uninstall.sh`
	`  - uninstall_agent`
	Enter **/uninstall.sh uninstall_agent** in the script. This command indicates that **uninstall.sh** will be executed before the component is deleted or stopped.

HTTP request mode	HTTP call request. – **Path**: (optional) URL of a request. Example: **/nginx**. – **Port**: (mandatory) request port. Example: **80**. – **Host Address**: (optional) IP address of the request. The default value is the IP address of the node where the application is located. Example: **127.0.0.1**.

----End

5.13.3 Configuring Data Storage

On-disk files in a container are ephemeral, which presents the following problems to component instances running in the container:

1. When a container is rebuilt, files in the container will be lost, which does not meet data persistence requirements.
2. When multiple containers run in a pod at the same time, files need to be shared among the containers, which poses a challenge of ensuring data consistency between container instances.

Setting data storage for container-deployed components can provide persistent storage to ensure data consistency during container restart, migration, or expansion.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Scenario

Table 5-13 Storage scenarios

Storage Type	Scenario
EVS Disks	EVS disks can be set only when a CCE cluster is bound to the component deployment environment. Supported specifications: • Common I/O: The backend storage is provided by the Serial Advanced Technology Attachment (SATA) storage media. Common I/O is applicable to scenarios where large capacity is needed but high read/write rate is not required, and the volume of transactions is low. Examples include development testing and enterprise office applications. • High I/O: The backend storage is provided by the Serial Attached SCSI (SAS) storage media. High I/O is applicable to scenarios where relatively high performance, high read/write rate, and real-time data storage are required. Examples include creating file systems and sharing distributed files. • Ultra-high I/O: The backend storage is provided by the Solid-State Drive (SSD) storage media. Ultra-high I/O is applicable to scenarios where high performance, high read/write rate, and data-intensive applications are required. Examples include NoSQL, relational database, and data warehouse (such as Oracle RAC and SAP HANA).
SFS File Systems	SFS can be set only when a CCE cluster is bound to the component deployment environment. SFS applies to a wide range of scenarios, including media processing, content management, big data, and workload analysis.
OBS Buckets	OBS can be set only when a CCE cluster is bound to the component deployment environment. • Standard OBS buckets: This type of OBS buckets applies to scenarios where a large number of hotspot files or small-sized files need to be accessed frequently (multiple times per month on average) and data can be quickly obtained. For example, cloud applications, data analysis, content analysis, and hotspot objects. • Infrequent access OBS buckets: This type of OBS buckets applies to scenarios where data is not frequently accessed (less than 12 times per year on average) but fast access response is required. For example, static website hosting, backup/active archiving, storage resource pools or backup storage for cloud services.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Storage Type	Scenario
SFS Turbo	SFS Turbo can be set only when a CCE cluster is bound to the component deployment environment. SFS Turbo file systems are fast, on-demand, and scalable, which are suitable for DevOps, containerized microservices, and enterprise office applications.
HostPath	The file directory of the host where the application component is located is mounted to the specified mounting point of the application. If the application component needs to access **/etc/hosts**, use **HostPath** to map **/etc/hosts**.
EmptyDir	Used for temporary storage. The lifecycle of temporary storage is the same as that of an application component instance. When an application component instance disappears, **EmptyDir** will be deleted and the data is permanently lost.
ConfigMap	Keys in a configuration item are mapped to an application component so that configuration files can be mounted to the specified application component directory.
Secret	Sensitive information such as application component authentication and keys is stored in a secret, and the secret is mounted to a specified path of the application component.
Network File System	Used to mount remote network files to containers for cross-instance and cross-node data sharing and persistent data storage and protection.

EVS Disks

EVS disks can be set only when a CCE cluster is bound to the component deployment environment.

EVS disks cannot be mounted across AZs and cannot be used by multiple workloads, multiple pods of the same workload, or multiple tasks.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **Cloud Storage** > **Add Cloud Storage** and set parameters by referring to **Table 5-14**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-14 EVS disks

Parameter	Description
Type	Select **EVS disk**. The method of using an EVS disk is the same as that of using a traditional disk. However, EVS disks have higher data reliability and I/O throughput and are easier to use. They apply to file systems, databases, or other system software or workloads that require block storage devices.
Allocation Mode	• Manual Select an EVS disk PersistentVolumeClaim (PVC). You need to statically create an EVS disk PVC and a storage volume in advance. For details, see **Using an Existing EVS Disk Through a Static PV**. The EVS disk PVC and the component instance container must be in the same namespace. • Automatic Storage is created automatically. You need to enter the storage capacity. 1. If the storage type is EVS disk, select an AZ for creating the EVS disk first. 2. Select a storage sub-type. High I/O: EVS disks that have high I/O and use SAS. Common I/O: EVS disks that use SATA. Ultra-high I/O: EVS disks that have ultra-high I/O and use SSD. 3. Enter the storage capacity, in the unit of GB. Ensure that the storage capacity quota is not exceeded; otherwise, creation will fail.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Add Container Path	1. Set **Sub-path** and **Mount Path** to the path to which the data storage is mounted. For example, if **Mount Path** is set to **/tmp** and **Sub-path** is set to **/app**, the data volume is mounted to **/tmp/app** of the application. – Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. – When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged. 2. Set **Permission**. – Read-only: allows you only to read data volumes in the application path. – Read/Write: allows you to modify data volumes in the application path. To prevent data loss, newly written data will not be migrated during application migration.

Step 3 Click **OK**.

----End

SFS File Systems

SFS can be set only when a CCE cluster is bound to the component deployment environment.

SFS applies to a wide range of scenarios, including media processing, content management, big data, and workload analysis.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **Cloud Storage** > **Add Cloud Storage** and set parameters by referring to **Table 5-15**.

Table 5-15 SFS file systems

Parameter	Description
Type	Select **SFS**. SFS applies to a wide range of scenarios, including media processing, content management, big data, and application analysis.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Allocation Mode	• Manual Select an SFS PVC. You need to statically create an SFS PVC and a storage volume in advance. For details, see **Using an Existing SFS File System Through a Static PV**. The SFS PVC and the component instance container must be in the same namespace. • Automatic Storage is created automatically. You need to enter the storage capacity. 1. Select a storage sub-type. Set the sub-type to NFS. 2. Enter the storage capacity, in the unit of GB. Ensure that the storage capacity quota is not exceeded; otherwise, creation will fail.
Add Container Path	1. Set **Sub-path** and **Mount Path** to the path to which the data storage is mounted. For example, if **Mount Path** is set to **/tmp** and **Sub-path** is set to **/app**, the data volume is mounted to **/tmp/app** of the application. – Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. – When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged. 2. Set **Permission**. – Read-only: allows you only to read data volumes in the application path. – Read/Write: allows you to modify data volumes in the application path. To prevent data loss, newly written data will not be migrated during application migration.

Step 3 Click **OK**.

----End

OBS Buckets

OBS can be set only when a CCE cluster is bound to the component deployment environment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **Cloud Storage** > **Add Cloud Storage** and set parameters by referring to **Table 5-16**.

Table 5-16 OBS buckets

Parameter	Description
Type	Select **OBS**. Standard and Infrequent Access OBS buckets are supported. OBS buckets apply to scenarios such as big data analytics, cloud native application data, static website hosting, and backup/active archiving.
Allocation Mode	• Manual Select an OBS PVC. You need to statically create an OBS PVC and a storage volume in advance. For details, see **Using an Existing OBS Bucket Through a Static PV**. The OBS volume and the component instance container must be in the same namespace. • Automatic 1. Set **Secret**. **Namespace** is the namespace of the container where the component instance is located and cannot be changed. Click **Use Existing Secret** to select the secret in the namespace of the container where the component instance is located. You can also create a secret: Enter a new secret name, click **Add Key File**, and upload the obtained local secret file. For details about how to obtain the secret file, see **Access Keys**. 2. Select a storage sub-type. You can select **Standard** or **Infrequent Access**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Add Container Path	1. Set **Sub-path** and **Mount Path** to the path to which the data storage is mounted. For example, if **Mount Path** is set to **/tmp** and **Sub-path** is set to **/app**, the data volume is mounted to **/tmp/app** of the application. – Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. – When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged. 2. Set **Permission**. – Read-only: allows you only to read data volumes in the application path. – Read/Write: allows you to modify data volumes in the application path. To prevent data loss, newly written data will not be migrated during application migration.

Step 3 Click **OK**.

----End

SFS Turbo

SFS Turbo can be set only when a CCE cluster is bound to the component deployment environment.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **Cloud Storage** > **Add Cloud Storage** and set parameters by referring to **Table 5-17**.

Table 5-17 SFS Turbo

Parameter	Description
Type	Select **SFS Turbo**. Standard and Infrequent Access OBS buckets are supported. OBS buckets apply to scenarios such as big data analytics, cloud native application data, static website hosting, and backup/active archiving.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Allocation Mode	• Manual Select an SFS Turbo volume. You need to statically create an SFS Turbo PVC and a storage volume in advance. For details, see **Using an Existing SFS Turbo File System Through a Static PV**. The SFS Turbo volume and the component instance container must be in the same namespace. • Automatic Select an available file system. You need to create a file system in advance. For details, see **Creating an SFS Turbo File System**.
Add Container Path	1. Set **Sub-path** and **Mount Path** to the path to which the data storage is mounted. For example, if **Mount Path** is set to **/tmp** and **Sub-path** is set to **/app**, the data volume is mounted to **/tmp/app** of the application. – Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. – When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged. 2. Set **Permission**. – Read-only: allows you only to read data volumes in the application path. – Read/Write: allows you to modify data volumes in the application path. To prevent data loss, newly written data will not be migrated during application migration.

Step 3 Click **OK**.

----End

HostPath

The file or directory of the host is mounted to the component.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **Local Disk** > **Add Local Disk** and set parameters by referring to **Table 5-18**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-18 HostPath

Parameter	Description
Storage Type	Select **HostPath**.
Host Path	Enter the host path, for example, **/etc/hosts**.
Container Mounting	1. Set **Sub-path** and **Mount Path** to the path to which the data storage is mounted. For example, if **Mount Path** is set to **/tmp** and **Sub-path** is set to **/app**, the data volume is mounted to **/tmp/app** of the application. – Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. – When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged. 2. Set **Permission**. – Read-only: allows you only to read data volumes in the application path. – Read/Write: allows you to modify data volumes in the application path. To prevent data loss, newly written data will not be migrated during application migration.

Step 3 Click **OK**.

----End

EmptyDir

EmptyDir applies to temporary data storage, disaster recovery, and shared running. It will be deleted upon deletion or transfer of application component instances.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **Local Disk** > **Add Local Disk** and set parameters by referring to **Table 5-19**.

Table 5-19 EmptyDir

Parameter	Description
Storage Type	Select **EmptyDir**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
EmptyDir	• If **Memory** is selected, the running speed is improved, but the storage capacity is limited by the memory size. This mode applies to a small amount of data with high requirements on reading and writing efficiency. • If **Memory** is not selected, data is stored in disks, which is applicable to a large amount of data with low requirements on reading and writing efficiency.
Container Mounting	1. Set **Sub-path** and **Mount Path** to the path to which the data storage is mounted. For example, if **Mount Path** is set to **/tmp** and **Sub-path** is set to **/app**, the data volume is mounted to **/tmp/app** of the application. – Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. – When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged. 2. Set **Permission**. – Read-only: allows you only to read data volumes in the application path. – Read/Write: allows you to modify data volumes in the application path. To prevent data loss, newly written data will not be migrated during application migration.

Step 3 Click **OK**.

----End

ConfigMap

ServiceStage separates the application codes from configuration files. **ConfigMap** is used to process application component configuration parameters.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **Local Disk** > **Add Local Disk** and set parameters by referring to **Table 5-20**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-20 ConfigMap

Parameter	Description
Storage Type	Select **ConfigMap**.
ConfigMap	Select the desired configuration item name. For details about how to create a configuration item, see **Creating a Configuration Item**. The configuration item and the component instance container must be in the same namespace.
Container Mounting	Set **Sub-path** and **Mount Path** to the path to which the data storage is mounted. For example, if **Mount Path** is set to **/tmp** and **Sub-path** is set to **/app**, the data volume is mounted to **/tmp/app** of the application. When you select **ConfigMap**, only **Read-only** is supported. You can only read the data volume in the container path. • Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. • When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged.

Step 3 Click **OK**.

----End

Secret

The data in the secret is mounted to the specified application component. The content of the secret is user-defined.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **Local Disk** > **Add Local Disk** and set parameters by referring to **Table 5-21**.

Table 5-21 Secret

Parameter	Description
Storage Type	Select **Secret**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Secret	Select the desired secret name. For details about how to create a secret, see **Creating a Secret**. The secret and the component instance container must be in the same namespace.
Container Mounting	Set **Sub-path** and **Mount Path** to the path to which the data storage is mounted. For example, if **Mount Path** is set to **/tmp** and **Sub-path** is set to **/app**, the data volume is mounted to **/tmp/app** of the application. When you select **Secret**, only **Read-only** is supported. You can only read the data volume in the container path. • Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. • When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged.

Step 3 Click **OK**.

----End

Network File System

A network file system (NFS) can be used to mount remote network files to containers for cross-instance and cross-node data sharing and persistent data storage and protection.

Step 1 Choose **Container Settings**.

Step 2 Choose **Data Storage** > **NFS** > **Add NFS** and set parameters by referring to **Table 5-22**.

Table 5-22 NFS

Parameter	Description
File Address	Address of the NFS server.
File Path	Shared path of remote network files. For example, **/data/nfs**.
Container Path	Mount point to which the data storage is mounted on the application instance. For example, **/mnt**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Sub-path	Specified file or directory mounted to the specified container path. For example, **data**. **Sub-path** and **Container Path** form the path for mounting the data storage to the application instance. For example, **/mnt/data**.

Step 3 Click **OK**.

----End

5.13.4 Configuring a Log Policy of an Application

ServiceStage allows you to configure a log policy for application components deployed in containers. If a CCE cluster is bound to the component deployment environment, you can view running logs on AOM or collect component running logs to LTS for processing. If no log policy is configured, the system collects standard application output logs by default.

You are advised to directly collect logs to LTS. The function of viewing logs in AOM will be removed soon.

Restrictions

If you want to collect the run logs of the container-deployed component to LTS for processing, there are the following restrictions:

- CCE cluster nodes whose container engine is Docker are supported.
- CCE cluster nodes whose container engine is Containerd are supported. You must be using ICAgent 5.12.130 or later.
- To collect container log directories mounted to host directories to LTS, you must configure the node file path.
- Currently, container file log collection supports only the Overlay2 storage driver, not Device Mapper. Run the following command to check the storage driver type:
  ```
  docker info | grep "Storage Driver"
  ```

Configuring a Log Policy of an Application

Step 1 Choose **Container Settings**.

Step 2 Choose **Log Collection** > **Add Log Policy**.

Step 3 Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Storage Type	Select a log storage type. • **Host Path**: Mount a host path to a specified container path. • **Container Path**: Logs are exported only to the container path. You do not need to mount the host path. The ICAgent needs to be upgraded to 5.10.79.
*Host Path	This parameter is mandatory when **Storage Type** is set to **Host Path**. Enter the log storage path on the host.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Container Mounting	1. Set **Mount Path** to the path to which the data storage is mounted. – Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created. – When the data volume is mounted to a high-risk directory, you are advised to use a low-permission account to start the application; otherwise, high-risk files on the host may be damaged. 2. Set **Extended Host Path**. This parameter is mandatory when **Storage Type** is set to **Host Path**. – **None**: No extended path is configured. – **PodUID**: Pod ID. – **PodName**: Pod name. – **PodUID/ContainerName**: Pod ID or container name. – **PodName/ContainerName**: Pod name or container name. 3. Set **Collection Path**. After specifying the collection path, you can specify the content to collect. The collection path can be specified if the ICAgent version of the collector is 5.12.22 or later. The following modes are supported: – If no collection path is specified, log files in **.log**, **.trace**, and **.out** formats will be collected from the current path by default. – If a collection path contains double asterisks (**), log files in **.log**, **.trace**, and **.out** formats will be collected from 5 levels of subdirectories. – If a collection path contains an asterisk (*), a fuzzy match is performed. Example: If the collection path is **/tmp/**/test*.log**, all **.log** files prefixed with **test** will be collected from **/tmp** and its 5 levels of subdirectories. 4. Set **Aging Period**. – **Hourly**: Log files are scanned every hour. If the size of a log file exceeds 20 MB, the system compresses the log file to a historical file, dumps the historical file to the directory where the log file is stored, and clears the original log file. – **Daily**: Log files are scanned once a day. If the size of a log file exceeds 20 MB, the system compresses the log file to a historical file, dumps the historical file to the directory where the log file is stored, and clears the original log file.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	– **Weekly**: Log files are scanned once a week. If the size of a log file exceeds 20 MB, the system compresses the log file to a historical file, dumps the historical file to the directory where the log file is stored, and clears the original log file. 5. Click to set **Log Format**. By default, the system collects and displays the logs printed by the program by line. If a complete log occupies multiple lines and you want to collect and display the entire log, you can set **Log Format** to enable multi-line logs. – **Single-line**: The system collects logs by line. – **Multi-line**: The system collects logs based on the configured matching rules. **Log Segmentation**: This parameter is mandatory when **Log Format** is set to **Multi-line**. It is used to match the beginning of each log. If you select **Log time**, the time matching mode is used. If you select **Regular expression**, the regular expression matching mode is used. **Time Wildcard**: Enter the time wildcard when **Log Segmentation** is set to **Log time**. For example, if the time format of each log is YYYY-MM-DD hh:mm:ss, set the time wildcard to YYYY-MM-DD hh:mm:ss. **Regular expression**: If **Log Segmentation** is set to **Regular expression**, enter the regular expression based on the format of the beginning of each log.

Step 4 Click **OK**.

Step 5 Determine whether to collect component run logs to LTS by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Whether to Collect Component Run Logs to LTS	Description
Yes	Select **Auto connection to LTS**. The system determines the log collection path based on **Step 3**: • If container mounting is configured, logs are delivered to the container file path of LTS. • If host mounting is configured, logs are delivered to the node file path of LTS. • If both container and node mounting are configured, only the node mounting path can be retained due to LTS restrictions, and logs are delivered only to the node file path of LTS. • If you do not configure log collection, the log path is **/var/log/*** or **/var/logs/***, which corresponds to the container file path of LTS. After a component is configured and deployed, the system automatically generates log group and log stream configurations, and generates access configurations and host groups on the **Ingestion Management** page of the LTS console. • **Log Group**: The naming rule is **servicestage-***{Application ID}*. Use the ID of the application to which the component belongs. • **Log Stream**: The naming rule is **lts-topic-***{Component name}*. Use the name of the component connected to LTS. • **Ingestion**: The naming rule is **access-config-***{Component name}*. Use the name of the component connected to LTS. • **Host Group**: The naming rule is **k8s-log-***{CCE cluster ID}*. Use the ID of the CCE cluster where the component connected to LTS is located.
No	Do not select **Auto connection to LTS**. • After the component is configured and deployed, you can view running logs on the AOM console. For details, see **Viewing Log Files**. • After the component is configured and deployed, you can collect component logs to LTS by referring to **Managing Component LTS Logs**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.13.5 Configuring Health Check

Health check periodically checks container health status during running of container-deployed components. If health check is not configured, a pod cannot detect application exceptions or automatically restart the application to restore it. This will result in a situation where the pod status is normal but the application in the pod is abnormal.

ServiceStage provides the following health check methods:

- **Component Liveness Probe**: checks whether an application component exists. It is similar to the **ps** command that checks whether a process exists. If the liveness check of an application component fails, the cluster restarts the application component. If the liveness check is successful, no operation is executed.
- **Component Service Probe**: checks whether an application component is ready to process user requests. It may take a long time for some applications to start before they can provide services. This is because that they need to load disk data or rely on startup of an external module. In this case, the application process exists, but the application cannot provide services. This check method is useful in this scenario. If the application component readiness check fails, the cluster masks all requests sent to the application component. If the application component readiness check is successful, the application component can be accessed.
- **Component Startup Probe**: checks whether an application has been started. After the control container is successfully started, the system checks the liveness and services to ensure that the liveness and service probes do not affect application startup. This can be used to perform liveness checks on slow starting containers to prevent them from getting terminated before they are started.

Health Check Modes

- HTTP request-based check

  This health check mode is applicable to application components that provide HTTP/HTTPS services. The cluster periodically sends an HTTP/HTTPS GET request to such application components. If the return code of the HTTP/HTTPS response is within 200–399, the check is successful. Otherwise, the check fails. In this health check mode, you must specify an application listening port and an HTTP/HTTPS request path.

  For example, if the application component provides the HTTP service, the port number is 80, and the HTTP check path is **/health-check**, the cluster periodically initiates the following request to the application:

  ```
  GET http://Host address:80/health-check
  ```

  If the host address is not set, the instance IP address is used by default.

- TCP port-based check

  For applications that provide a TCP communication service, the cluster periodically establishes a TCP connection to the application. If the connection is successful, the probe is successful. Otherwise, the probe fails. In this health check mode, you must specify an application listening port. For example, if you have a Nginx application component with service port 80, after you configure a TCP port-based check for the application component and specify

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

port 80 for the check, the cluster periodically establishes a TCP connection with port 80 of the application component. If the connection is successful, the check is successful. Otherwise, the check fails.

- CLI-based check

  In this mode, you must specify an executable command in an application component. The cluster will periodically execute the command in the application component. If the command output is **0**, the health check is successful. Otherwise, the health check fails.

  The CLI mode can be used to replace the following modes:

  – TCP port-based check: Write a program script to connect to an application component port. If the connection is successful, the script returns **0**. Otherwise, the script returns **–1**.

  – HTTP request-based check: Write a program script to run the **wget** command for an application component.

    wget http://127.0.0.1:80/health-check

    Check the return code of the response. If the return code is within 200–399, the script returns **0**. Otherwise, the script returns **–1**.

    ▪ Put the program to be executed in the application component image so that the program can be executed.

    ▪ If the command to be executed is a shell script, add a script interpreter instead of specifying the script as the command. For example, if the script is **/data/scripts/health_check.sh**, you must specify **sh/data/scripts/health_check.sh** for command execution. The reason is that the cluster is not in the terminal environment when executing programs in an application component.

- gRPC–based check

  gRPC checks can configure startup, liveness, and readiness probes for your gRPC application without exposing any HTTP endpoint, nor do you need an executable. Kubernetes can connect to your workload via gRPC and obtain its status.

  – The gRPC check is supported only in clusters of Kubernetes v1.25 or later.

  – To use gRPC for check, your application must support the **gRPC health checking protocol**.

  – Similar to HTTP and TCP probes, if the port is incorrect, the application does not support the health checking protocol, or there are another configuration error, the check will fail.

Common Parameter Description

Table 5-23 Common parameter description

Parameter	Description
Check Frequency(s)	Probe detection period, in seconds. Example: Value **30** indicates that the detection is performed every 30s.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Latency (s)	Check delay time. Unit: second. Set this parameter according to the normal startup time of services. For example, if this parameter is set to 30, the health check will be started 30 seconds after the application starts. The time is reserved for containerized services to start.
Timeout Period (s)	Timeout duration. Unit: second. If the time exceeds this value, the health check fails. For example, setting this parameter to 10 indicates that the health check timeout period is 10s. If the parameter is left blank or set to **0**, the default timeout time is 1s.
Success Threshold	Minimum consecutive successes for the probe to be considered successful after the detection fails. Example: Value **1** indicates that the workload is normal only when the health check is successful for one consecutive time after the health check fails. Default value: **1**. Minimum: **1**. The value must be **1** for liveness and startup probes.
Failure Thresholds	Number of retries after the detection fails. Giving up in case of liveness probe means to restart the container. In case of readiness probe the pod will be marked **Unready**.

Configuring Health Check

Step 1 Choose **Container Settings** > **Health Check**.

Step 2 Configure health check. For details about common parameters, see **Table 5-23**.

----End

5.14 Managing Application Settings of a Container-Deployed Component

ServiceStage provides **configuration management**. By creating a unified configuration file, you can fill the system information (such as the IP address, port number, database address, and application name associated with the environment) of the component environment and application in the variables corresponding to the configuration file. This implements multi-environment component release and upgrade with one-time configuration through file mounting.

This section describes how to manage component configuration files.

- **Binding a Configuration File**: Bind the created configuration file to the component. This implements multi-environment component release and

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

upgrade with one-time configuration through file mounting. Create the configuration file in advance. For details, see **Creating a Configuration File** or **Importing a Configuration File**.

- **Editing a Bound Configuration File**: Reset the parameters of the configuration file bound to the component.
- **Unbinding a Bound Configuration File**: Unbind a configuration file that is no longer used from the component. Unbinding a configuration file will not delete it. To delete it, see **Deleting a Configuration File**.

Binding a Configuration File

Step 1 Choose **Application Settings** > **Add Configuration File**.

Step 2 Set the configuration file parameters by referring to the following table.

Parameter	Description
Group	Group to which the configuration file belongs.
Configuration File Name	Name of the custom configuration file (for example, **config.yaml**) mounted to the container mounting path.
Configuration File Version	Version of the configuration file to be bound to the component. **latest** is selected by default, indicating the latest version.
ConfigMap Mount Path	Location of the operating system files. Enter a relative path to the working directory. You can get the working directory location from the environment variable **${APP_HOME}**. Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created.

Step 3 Click **OK**.

Step 4 (Optional) Select a created key from the **Associate Key** drop-down list to associate the key with the configuration file.

If sensitive information such as passwords needs to be encrypted in your configuration items, you can interconnect with DEW and use a custom key to encrypt and decrypt sensitive information.

This parameter is available only when Data Encryption Workshop (DEW) is deployed in the region where ServiceStage is deployed and a CCE cluster is bound to the component deployment environment.

For details about how to create a key, see **Creating a Key**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Editing a Bound Configuration File

Step 1 Choose **Application Settings**.

Step 2 Select the target configuration file in the configuration file list.

Step 3 Click **Edit** in the **Operation** column.

Step 4 Set the configuration file parameters by referring to the following table.

Parameter	Description
Group	Group to which the configuration file belongs.
Configuration File Name	Name of the custom configuration file (for example, **config.yaml**) mounted to the container mounting path.
Configuration File Version	Version of the configuration file to be bound to the component. **latest** is selected by default, indicating the latest version.
ConfigMap Mount Path	Location of the operating system files. Enter a relative path to the working directory. You can get the working directory location from the environment variable **${APP_HOME}**. Do not mount a data volume to a system directory such as **/** or **/var/run**. Otherwise, an exception occurs. An empty directory is recommended. If the directory is not empty, ensure that the directory does not contain any file that affects application startup. Otherwise, the files will be replaced, causing application startup exceptions. As a result, the application fails to be created.

Step 5 Click **OK**.

Step 6 (Optional) Select a created key from the **Associate Key** drop-down list to associate the key with the configuration file.

If sensitive information such as passwords needs to be encrypted in your configuration items, you can interconnect with DEW and use a custom key to encrypt and decrypt sensitive information.

This parameter is available only when Data Encryption Workshop (DEW) is deployed in the region where ServiceStage is deployed and a CCE cluster is bound to the component deployment environment.

For details about how to create a key, see **Creating a Key**.

----End

Unbinding a Bound Configuration File

Step 1 Choose **Application Settings**.

Step 2 Select the target configuration file in the configuration file list.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Remove** in the **Operation** column.

----End

5.15 Managing Advanced Settings of a Container-Deployed Component

5.15.1 Configuring a Component Upgrade Policy

In practice, component upgrade is a common scenario. Components deployed based on a container in ServiceStage support the following upgrade policies:

- RollingUpdate: New pods are created gradually, and then old pods are deleted. This policy is used by default when you **create and deploy a component based on a container**, **upgrade a component in rolling release mode**, **upgrade a component in dark launch mode**, and **upgrade components in batches**.
- Recreate: Old pods are deleted, and then new pods are created. This policy is used by default when you **upgrade a component in single-batch release mode**.

You can configure the component upgrade policy when configuring **Advanced Settings**.

Configuring a Component Upgrade Policy

Step 1 Choose **Advanced Settings** > **Upgrade Policy**.

Step 2 Configure the upgrade policy by referring to the following table.

Parameter	Description
Deployment Batch	Number of batches in which component instances are upgraded. The value range is [1, Total number of instances]. Total number of instances refers to the number of running instances of the component. For example, if there are 4 component instances and **Deployment Batch** is set to **2**, these component instances are upgraded in two batches, and each batch involves two component instances. • This parameter cannot be set when you perform this operation in **Creating a Component Based on a Container Using Manual Configurations** or **Upgrading a Component in Single-batch Release Mode**. • This parameter can be set when this operation can be performed in other scenarios.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Max. Unavailable Pods	Maximum number of unavailable pods allowed in each rolling upgrade. If the number is equal to the total number of pods, services may be interrupted. (Minimum number of alive pods = Total number of pods – Maximum number of unavailable pods) ● This parameter can be set when you perform this operation on components whose **Workload Type** is **Stateless** in **Creating a Component Based on a Container Using Manual Configurations**. The units and value ranges are described as follows: – Maximum number of unavailable pods compared with the number of instances. The value ranges from 0 to the number of pods. – Maximum percentage of unavailable pods to the total number of pods. The value ranges from 0 to 100. ● This parameter can be set when you perform this operation on stateless components in other scenarios. The value is determined by the number of instances and **Deployment Batch**. If there is one instance, the value is **1**. If there is more than one instance, the value depends on whether the number of instances can be exactly divided by the value of **Deployment Batch**. – If yes, use the quotient. – If no, round down the quotient and then plus 1.
Max. Surge	Maximum number of pods that can be created over the desired number of pods in each rolling upgrade. The units are described as follows: ● Maximum number of available pods compared with the number of instances. The value ranges from 0 to 2147483647. ● Maximum percentage of available pods to the total number of pods. The value ranges from 0 to 100. This parameter can be set when you perform this operation on stateless components.
Min. Ready Seconds	Minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is **0**, indicating that a component instance is considered available immediately after it is ready. The value ranges from 0 to 2147483647.
Revision History Limit	Number of old pods to retain to allow rollback. The default value is **10**. The value ranges from 0 to 2147483647.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Max. Upgrade Seconds	The time (in seconds) the system waits for before reporting an upgrade failure for components whose **Workload Type** is **Stateless**. If this parameter is specified, the value must be greater than that of **Min. Ready Seconds**. The value ranges from 0 to 2147483647.
Scale-in Time Window (s)	The time window for executing commands before a workload stops. The default value is 30s. The value ranges from 0 to 2147483647. You can set a graceful scale-in time window to save important data before a component instance stops. For example, if an application has two instances and only one instance will be kept after the scale-in operation, you can still perform certain operations on the instance to be stopped in the specified time window.
Upgrade Method	This parameter can be set when **Upgrading a Component in Rolling Release Mode** for components whose **Workload Type** is **Stateful**. • Rolling upgrade Gradually replaces old pods with new ones. During the upgrade, service traffic is evenly distributed to both old and new pods to ensure service continuity. • Automatic replacement upgrade You can set **First Batch of Instances to Be Upgraded** to select the first batch of pods to be upgraded based on the **Tag Key**, **Tag Value**, and **Operator**. – **In**: filters pods with the specified tag key in the tag value list. Multiple tag values are separated by semicolons (;). The tag value is a string. – **NotIn**: filters pods with the specified tag key not in the tag value list. Multiple tag values are separated by semicolons (;). The tag value is a string. – **Exists**: filters pods with the specified tag key. You do not need to enter the tag value. – **DoesNotExist**: filters pods without the specified tag key. You do not need to enter the tag value. The system will automatically use the latest configuration to start new pods to replace the first batch of pods.

----End

5.15.2 Configuring a Scheduling Policy of a Component Instance

Kubernetes supports node affinity and pod affinity/anti-affinity. You can configure custom rules for affinity and anti-affinity scheduling. For example, you can deploy

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

frontend pods and backend pods together, deploy the same type of applications onto specific nodes, or deploy applications onto different nodes.

Kubernetes affinity applies to nodes and pods.

- Node affinity: Kubernetes can schedule workloads to affinity nodes based on their labels and label values. For example, some nodes support GPU computing, and node affinity scheduling can guarantee that high-performance computing pods run on these GPU nodes.
- Workload affinity/anti-affinity: Kubernetes offers workload affinity and anti-affinity scheduling, which allows flexible scheduling of new workloads on either related or unrelated nodes. This results in improved cluster performance and utilization. For example, frontend pods and backend pods that frequently communicate with each other can be preferentially scheduled to the same node or AZ to minimize network latency.

  Workload affinity and anti-affinity require time for computing, which significantly slows down scheduling in large-scale clusters. Do not enable workload affinity and anti-affinity in a cluster that contains hundreds of nodes.

You can **Configuring an Affinity Scheduling Policy of a Node** and **Configuring an Affinity/Anti-affinity Scheduling Policy of a Workload** for container-deployed component instances when configuring **Advanced Settings**.

Configuring an Affinity Scheduling Policy of a Node

Step 1 Choose **Advanced Settings** > **Scheduling Policy**.

Step 2 Select a type of the node **affinity** scheduling rule.

- **Mandatory**: hard constraints that must be met. If multiple **Mandatory** rules are added, scheduling will be performed if one of the rules is met.
- **Best effort**: soft constraints that are met as much as possible. If multiple **Best effort** rules are added, scheduling will be performed if one or none of the rules is met.

Step 3 Click **Add**.

Step 4 Select **Node** for **Object Type**.

Step 5 Add objects as required.

- Add a policy: Click **Add Policy** and set the scheduling policy parameters by referring to the following table. You can customize a scheduling policy.
- Specify a node: Click **Specify Node** to quickly select the node to be scheduled. You do not need to manually enter the node label and label value. The kubernetes.io/hostname label is used when you specify a node.
- Specify an AZ: Click **Specify AZ** to quickly select the AZ to be scheduled. You do not need to manually enter the AZ label and label value. The failure-domain.beta.kubernetes.io/zone label is used when you specify an AZ.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Weight	This parameter is mandatory when the policy type selected in Step 2 is **Best effort**. During scheduling, the scheduler adds the weight to the scores of other priority functions and schedules pods to the node with the largest total score. The weight ranges from 1 to 100.
Label Name	When you click **Add Policy** to customize a scheduling policy, enter the node label to be matched.
Operators	• When you click **Add Policy** to customize a scheduling policy, set the following mapping relationships: – **In**: The label of the affinity object is in the label value list (**values** field). – **NotIn**: The label of the affinity object is not in the label value list (**values** field). – **Exists**: The affinity object has a specified label name. – **DoesNotExist**: The affinity object does not have a specified label name. – **Gt**: The label value of the scheduling node exceeds what is listed (character string comparison). – **Lt**: The label value of the scheduling node is under what is listed (character string comparison). • When configuring a **Specify Node** or **Specify AZ** scheduling policy, set the following matching relationships: – **In**: The label of the affinity object is in the label value list (**values** field). – **NotIn**: The label of the affinity object is not in the label value list (**values** field).
Label Value	When you click **Add Policy** to customize a scheduling policy, enter the label value corresponding to the label name of the node based on the selected operator. • If **Operators** is **In** or **NotIn**, enter characters for the label value. Separate multiple labels by semicolons (;). Example: `str1;str2;str3` • If **Operators** is **Exists** or **DoesNotExist**, the label value can be empty. • If **Operators** is **Gt** or **Lt**, enter an integer for the label value, for example, **5** or **-5**.

Step 6 Click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuring an Affinity/Anti-affinity Scheduling Policy of a Workload

Step 1 Choose **Advanced Settings** > **Scheduling Policy**.

Step 2 Select a type of the workload **affinity** or **anti-affinity** scheduling rule by referring to the following table.

Scheduling Policy	Rule Type	Description
Affinity	Mandatory	Hard constraints that must be met. Select pods that require affinity by label. If such pods already run on a node in the topology key, the scheduler will forcibly schedule the created pods to that topology key. **NOTE** If multiple affinity rules are configured, multiple labels will be used to filter pods that require affinity, and the newly created pods must have affinity with all pods that meet the label filtering conditions. In this way, all pods that meet the label filtering conditions locate in the same topology key for scheduling.
	Best effort	Soft constraints that are met as much as possible. Select pods that require affinity by label. If such pods already run on a node in the topology key, the scheduler will preferentially schedule the created pods to that topology key. **NOTE** If multiple affinity rules are configured, multiple labels will be used to filter pods that require affinity, and the newly created pods will be preferentially to have affinity with multiple pods that meet the label filtering conditions. However, even if no pod meets the label filter conditions, a topology domain will be selected for scheduling.
Anti-affinity	Mandatory	Hard constraints that must be met. Select one or more pods that require anti-affinity by label. If such pods already run on a node in the topology key, the scheduler will not schedule the created pods to that topology key. **NOTE** If multiple anti-affinity rules are configured, multiple labels will be used to filter pods that require anti-affinity, and the newly created pods must have anti-affinity with all pods that meet the label filtering conditions. In this way, all the topology keys where the pods that meet the label filtering conditions locate will not be scheduled.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Scheduling Policy	Rule Type	Description
	Best effort	Soft constraints that are met as much as possible. Select one or more pods that require anti-affinity by label. If such pods already run on a node in the topology key, the scheduler will preferentially schedule the created pods to other topology keys. **NOTE** If multiple anti-affinity rules are configured, multiple labels will be used to filter pods that require anti-affinity, and the newly created pods will be preferentially to have anti-affinity with multiple pods that meet the label filtering conditions. However, even if all topology keys involve the pods that require anti-affinity, a topology key will be selected for scheduling.

Step 3 Click **Add**.

Step 4 Select **Workload** for **Object Type**.

Step 5 Add objects as required.

- Click **Add Policy** to customize a scheduling policy. The scheduling policy is implemented based on node labels.
- Click **Specify Component**, select the component to be scheduled, and click **OK** to quickly select the component to be scheduled. You do not need to manually enter the component label and label value.

Parameter	Description
Weight	This parameter is mandatory when the policy type selected in Step 2 is **Best effort**. During scheduling, the scheduler adds the weight to the scores of other priority functions and schedules pods to the node with the largest total score. The weight ranges from 1 to 100.
Namespace	Select the namespace to which the object belongs. This parameter specifies the namespace in which the scheduling policy takes effect.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Topology Key	A topology key (**topologyKey**) determines the range of nodes to be scheduled based on node labels, identifies affinity/anti-affinity objects based on the labels and operators, and performs scheduling based on the topology key where the target object is located. – For example, if the node label is **kubernetes.io/hostname**, the label value will be a node name. Nodes with different names are assigned to different topology keys. This allows workload affinity scheduling on a single node, as each topology key contains only one node. – If the node label is **kubernetes.io/os**, the label value will be a node OS. Nodes running different OSs are assigned to different topology keys. This allows workload affinity scheduling on multiple nodes, as each topology key contains multiple nodes. For example, if a pod that meets the workload affinity rule runs on a node in a topology key, all nodes in the topology key can be scheduled.
Label Name	– When you click **Add Policy** to customize a scheduling policy, enter the workload label to be matched. – When you click **Specify Component** to set the scheduling policy for a specified component, you can use the default label or customize a label.
Operators	When you click **Add Policy** to customize a scheduling policy, set the matching relationship. – **In**: The label of the affinity or anti-affinity object is in the label value list (**values** field). – **NotIn**: The label of the affinity or anti-affinity object is not in the label value list (**values** field). – **Exists**: The affinity or anti-affinity object has a specified label name. – **DoesNotExist**: The affinity or anti-affinity object does not have a specified label name.
Label Value	Enter the label value corresponding to the workload label name based on the selected operator. – If **Operators** is **In** or **NotIn**, enter characters for the label value. Separate multiple labels by semicolons (;). Example: `str1;str2;str3` – If **Operators** is **Exists** or **DoesNotExist**, the label value can be empty.

Step 6 Click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.15.3 Configuring a Tolerance Policy of a Component Instance

For container-deployed components, tolerations allow the scheduler to schedule pods to nodes with target taints. Tolerances work with node taints. Each node allows one or more taints. If no tolerance is configured for a pod, the scheduler will schedule the pod based on node taint policies to prevent the pod from being scheduled to an inappropriate node.

As shown in **Table 5-24**, you can specify multiple effects for taints.

Table 5-24 Taint effects

Taint Effect	No Taint Tolerance Policy Configured	Taint Tolerance Policy Configured
NoExecute	• Pods running on the node will be evicted immediately. • Inactive pods will not be scheduled to the node.	• If the toleration time window is not specified, pods can always run on this node. • If the toleration time window is specified, pods still run on the node with taints within the time window. After the time expires, the pods will be evicted.
PreferNoSchedule	• Pods running on the node will not be evicted. • Inactive pods will not be scheduled to the node to the best extend.	Pods can always run on this node.
NoSchedule	• Pods running on the node will not be evicted. • Inactive pods will not be scheduled to the node.	Pods can always run on this node.

Kubernetes automatically adds tolerances for the node.kubernetes.io/not-ready and node.kubernetes.io/unreachable taints to pods, and sets the toleration time window to 300s. These default tolerance policies indicate that when either of the preceding taint is added to the node where pods are running, the pods can still run on the node for 5 minutes.

NOTE

When a DaemonSet pod is created, no toleration time window will be specified for the tolerations automatically added for the preceding taints. When either of the preceding taints is added to the node where the DaemonSet pod is running, the DaemonSet pod will never be evicted.

For details about common taints, see **Table 5-25**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-25 Common taints

Taint	Description
node.kubernetes.io/not-ready	Node not ready.
node.kubernetes.io/unreachable	Node controller cannot access node.
node.kubernetes.io/memory-pressure	Node has insufficient memory.
node.kubernetes.io/network-unavailable	Node network unavailable.
node.kubernetes.io/unschedulable	Node not schedulable.

You can configure the component instance tolerance policy when configuring **Advanced Settings**.

Prerequisites

You have added taints to the target cluster node. For details, see **Managing Node Taints**.

Configuring a Tolerance Policy of a Component Instance

Step 1 Choose **Advanced Settings** > **Tolerance Policy**.

Step 2 Click **Add Policy** to add a taint tolerance policy by referring to the following table.

Parameter	Description
Taint Key	Enter the taint key of the node.
Operators	• **Equal**: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. • **Exists**: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
Taint Value	This parameter is mandatory when **Operators** is set to **Equal**.
Taint Policy	• **All**: All taint policies are matched. • **NoSchedule**: Only the NoSchedule taint is matched. • **PreferNoSchedule**: Only the PreferNoSchedule taint is matched. • **NoExecute**: Only the NoExecute taint is matched.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Toleration Time Window (s)	This parameter can be set only when **Taint Policy** is set to **NoExecute**. Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

----End

5.15.4 Configuring Performance Management of a Container-Deployed Component

Performance management helps you quickly locate problems and identify performance bottlenecks to improve your experience. Selecting Java probe will start APM and install Java probes on the nodes deployed with APM, which consumes a small amount of resources. Java probes use the bytecode enhancement technology to trace Java application calls and generate topology and call chain data.

You can configure performance management of container-deployed components when configuring **Advanced Settings**.

Restrictions

- This function can be enabled only when APM of the corresponding version is deployed and enabled in the environment.
- JDK 8, 11, 17, and 21 are supported.
- Tomcat 8.x and 9.x are supported. For details, see **Usage Restrictions**.
- Performance management can be configured for Java applications (technology stack type being Java, Tomcat, or Java-based Docker) deployed using CCE clusters.

Configuring Application Performance Management

Step 1 Choose **Advanced Settings** > **Performance Management**.

Step 2 Select **Java probe** and set probe parameters based on your probe type by referring to the following table.

Probe Type	Description
APM 1.0	Select a probe version from the drop-down list.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Probe Type	Description
APM 2.0	1. Set **Application**. – Select the application that has the same name as the application where the application component is located from the drop-down list. – If the application does not exist, click **Create Application** to create it. After the application is created, you can log in to the APM console and view the new application in the application list. 2. Set **Probe Version**. Select a probe version from the drop-down list. You can select a probe version based on the CPU architecture of the master node in the CCE cluster where the component is deployed. For example, if the CPU architecture is x86_64, select **latest-x86_64**. 3. Set **Upgrade Policy**. Select a probe upgrade policy. By default, **Automatic upgrade upon restart** is selected. – **Automatic upgrade upon restart**: The system downloads the probe image each time the pod is restarted. – **Manual upgrade**: This policy means that if a local image is available, the local image will be used. The system downloads the probe image only when a local image is unavailable. 4. Set **Access Key**. The access key is automatically obtained. If the access key cannot be automatically obtained, manually enter it.

----End

5.15.5 Configuring Custom Monitoring of a Component

ServiceStage allows you to obtain custom metrics when components are deployed using CCE. You can use this method to report custom component running metrics.

You can configure custom monitoring of a component when configuring **Advanced Settings**.

Restrictions

- A CCE cluster must be bound to the component deployment environment.
- Currently, only **Gauge metrics** of Prometheus can be obtained.
- Before setting custom metric monitoring for an application component, you must understand **Prometheus** and provide the GET API for obtaining custom metric data in your application component so that ServiceStage can obtain custom metric data using this API.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuring Custom Monitoring of a Component

Step 1 Choose **Advanced Settings** > **O&M Policy**.

Step 2 Configure custom monitoring by referring to the following table.

Parameter	Description	Mandatory
Report Path	URL provided by the exporter for ServiceStage to obtain custom metric data. Example: **/metrics**	Yes
Report Port	Port provided by the exporter for ServiceStage to obtain custom metric data. Example: **8080**	Yes
Monitoring Metrics	Name of the custom metric provided by the exporter. Example: **["cpu_usage","mem_usage"]** • If this parameter is not set, ServiceStage collects data of all custom metrics. • If you set this parameter, for example, to **["cpu_usage","mem_usage"]**, ServiceStage collects the data of the specified cpu_usage and mem_usage metrics.	No

After the configuration and deployment are complete, you can view the monitoring metric data on the AOM console. For details, see **Metric Monitoring**.

----End

5.15.6 Configuring Component Job Settings

ServiceStage allows you to configure job settings of a container-deployed component whose **Workload Type** is **Job** to ensure efficient and reliable batch processing. You can configure job parameters to implement flexible task management and resource optimization.

You can configure job settings of a component whose **Workload Type** is **Job** when configuring **Advanced Settings**.

Configuring Component Job Settings

Step 1 Choose **Advanced Settings** > **Job**.

Step 2 Set job parameters by referring to the following table.

Parameter	Description
Parallel Pods	Maximum number of pods that can run in parallel during job execution. The value cannot be greater than the total number of pods in the job.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Timeout (s)	Once a job reaches this time, the job status becomes failed and all pods in this job will be deleted. If you leave this parameter blank, the job will never time out.
Completion Mode	• **Non-indexed**: A job is considered complete when all the pods are successfully executed. Each pod completion is homologous to each other. • **Indexed**: Each pod gets an associated completion index from 0 to the number of pods minus 1. The job is considered complete when every pod allocated with an index is successfully executed. For an indexed job, pods are named in the format of ${job-name}-${index}.
Suspend Job	By default, a job is executed immediately after being created. • Enabled: The job is in the suspended state after being created. • Disabled: The job is executed after being created.

----End

5.16 Managing VM-Deployed Component Configurations

5.16.1 Configuring VM Settings

- **Adding an Environment Variable of a Component**

  Environment variables are configured in the component running environment and can be modified during **Upgrading a Component in Rolling Release Mode** after component deployment, providing great flexibility. Environment variables set for a component are local environment variables and take effect only for this component.

  If you add an application environment variable to the application where the component is located and the name of the application environment variable is the same as that of the component environment variable in the application, the application environment variable is shielded by the component environment variable and does not take effect for the component. For details about how to add application environment variables, see **Managing Environment Variables of an Application**.

  Priority of environment variables of the component, application, and environment levels: component > application > environment.

  For details about environment variables of an environment, see **Managing Environment Variables of an Environment**.

- **Adding a Configuration Item**

  If the package type is package during **Creating a Component Based on a VM**, you can set the component configuration items by referring to this section.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Configuring Directory Settings**

  Manage the startup users, running directories, and log directories of VM-deployed components. You can set the component startup user, select the default directory rule, or customize directory rules as required.

- **Configuring the Lifecycle of a Component**

  If the component package type is JAR or WAR, you can configure specific commands as required to execute tasks after the component is started and before the component is stopped.

Adding an Environment Variable of a Component

Step 1 Choose **VM Settings** > **Environment Variable**.

Step 2 Click **Add Environment Variable**.

Step 3 Enter a key and value.

The key must be unique in a component. Enter 1 to 64 characters. Do not start with a digit or period (.). Only use letters, digits, underscores (_), hyphens (-), and periods (.).

----End

Adding a Configuration Item

Step 1 Choose **VM Settings** > **Environment Variable**.

Step 2 Click **Adding a Configuration Item**.

Step 3 Enter a key and value.

The key must be unique in a component. Enter 1 to 64 characters. Do not start with a digit or period (.). Only use letters, digits, underscores (_), hyphens (-), and periods (.).

----End

Configuring Directory Settings

Step 1 Choose **VM Settings** > **Directory Settings**.

Step 2 Set parameters based on the **Directory Configuration Mode**.

- Use default system directory

  The component deployment and log directories are the default system paths. The default component startup user and the user who can access the component deployment and log directories are **www**, and the default user group is **www**. For details about the default system directory configuration, see **Table 5-26**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-26 Default system directory configuration

Parameter	Description
Deployment Directory	– **Path**: parent directory of the component deployment path. The default value is **/opt/application**, and the absolute deployment path of the component is **/opt/application/***${Component instance name excluding the five random characters generated by the system}}/${Component version}/${Component instance ID}*. For example, if the component instance name is **comp-ecs-env-cce**_8DGO1, the component version number is 2025.0520.19070, and the component instance ID is 41a71313-46a4-411e-9eb5-388b16c9de3a, the absolute deployment path is **/opt/application/comp-ecs-env-cce/2025.0520.19070/41a71313-46a4-411e-9eb5-388b16c9de3a**. For details about the component instance name, see **Context**. – **User:User Group**: owner and group of the files in the component deployment directory. The default values are **www** and **www**, indicating that the file owner is the **www** user and the group to which the file belongs is the **www** group. – **Permissions**: access permissions on the files in the component deployment directory. The default value is **751**, indicating that the file owner (**www**) has the read, write, and execute permissions (**rwx**), the group to which the file belongs (**www**) has the read and execute permissions (**rx**), and other users have the execute permission (**x**).

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Log Directory	– **Path**: parent directory for storing component running logs. The default value is **/var/log/application**, and the absolute path is **/var/log/application/***${Component instance name excluding the five random characters generated by the system}/${Component instance ID}*. For example, if the component instance name is **comp-ecs-env-cce**_8DGO1, and the component instance ID is 41a71313-46a4-411e-9eb5-388b16c9de3a, the absolute storage path is **/opt/application/comp-ecs-env-cce/41a71313-46a4-411e-9eb5-388b16c9de3a**. For details about the component instance name, see **Context**. – **User:User Group**: owner and group of the files in the component log directory. The default values are **www** and **www**, indicating that the file owner is the **www** user and the group to which the file belongs is the **www** group. – **Permissions**: access permissions on the files in the component log directory. The default value is **751**, indicating that the file owner (**www**) has the read, write, and execute permissions (**rwx**), the group to which the file belongs (**www**) has the read and execute permissions (**rx**), and other users have the execute permission (**x**).
Running User	Component startup user. The default value is **www**.

- Use custom system directory

  The component deployment and log directories are the custom system paths. For details about how to customize a system directory, see **Customizing a System Directory**.

  – Click **Sync** to synchronize the latest directory configuration after **Customizing a System Directory** to the component.

  – Click **Roll Back** to roll back the directory configuration to the source configuration.

----End

Configuring the Lifecycle of a Component

Step 1 Choose **VM Settings** > **Lifecycle**.

Step 2 Click the **Post-start Processing** or **Pre-stop Processing** tab and set the parameters by referring to **Table 5-27**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 5-27 Lifecycle parameters

Processing Mode	Command
CLI mode	Command to be executed in the component instance. The command format is **Command** *Args[1] Args[2]*..... **Command** is a system command (**Bash mode**) or a user-defined executable program (**Binary mode**). If no path is specified, the system searches for an executable program in the default path. To run multiple commands, write them in different lines or into a script. Commands that are executed in the background or asynchronously are not supported.
HTTP request mode	HTTP call request. • **Protocol**: request protocol. **HTTP** and **HTTPS** are supported. • **Host Address**: (optional) IP address of the request. The default value is the IP address of the node where the application is located. Example: **127.0.0.1**. • **Port**: (mandatory) request port. Example: **80**. • **Path**: (optional) URL of a request. Example: **/nginx**.

----End

5.16.2 Managing Application Settings of a VM-Deployed Component

ServiceStage provides **configuration management**. By creating a unified configuration file, you can fill the system information (such as the IP address, port number, database address, and application name associated with the environment) of the component environment and application in the variables corresponding to the configuration file. This implements multi-environment component release and upgrade with one-time configuration through file mounting.

This section describes how to manage component configuration files.

- **Binding a Configuration File**: Bind the created configuration file to the component. This implements multi-environment component release and upgrade with one-time configuration through file mounting. Create the configuration file in advance. For details, see **Creating a Configuration File** or **Importing a Configuration File**.
- **Editing a Bound Configuration File**: Reset the parameters of the configuration file bound to the component.
- **Unbinding a Bound Configuration File**: Unbind a configuration file that is no longer used from the component. Unbinding a configuration file will not delete it. To delete it, see **Deleting a Configuration File**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Binding a Configuration File

Step 1 Choose **Application Settings** > **Add Configuration File**.

Step 2 Set the configuration file parameters by referring to the following table.

Parameter	Description
Group	Group to which the configuration file belongs.
Configuration File Name	Name of the custom configuration file (for example, **config.yaml**) mounted to the VM mounting path.
Configuration File Version	Version of the configuration file to be bound to the component. **latest** is selected by default, indicating the latest version.
ConfigMap Mount Path	Location of the operating system files. Enter a relative path to the working directory. You can get the working directory location from the environment variable **${APP_HOME}**. Relative path for mounting the configuration file. Example: **./config**. The upper-level directory is **/opt/application/*{CAS_INSTANCE_NAME}*/*{CAS_INSTANCE_VERSION}*/*{CAS_INSTANCE_ID}***, which stores the environment dependency, installation, startup, and stop scripts, and software packages required for component deployment. • **{CAS_INSTANCE_NAME}** is the component instance name. • **{CAS_INSTANCE_VERSION}** is the component instance version. • **{CAS_INSTANCE_ID}** is the component instance ID.

Step 3 Click **OK**.

Step 4 (Optional) Select a created key from the **Associate Key** drop-down list to associate the key with the configuration file.

If sensitive information such as passwords needs to be encrypted in your configuration items, you can interconnect with DEW and use a custom key to encrypt and decrypt sensitive information.

This parameter is available only when Data Encryption Workshop (DEW) is deployed in the region where ServiceStage is deployed and the component deployment environment manages ECSs.

For details about how to create a key, see **Creating a Key**.

----End

Editing a Bound Configuration File

Step 1 Choose **Application Settings**.

Step 2 Select the target configuration file in the configuration file list.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Edit** in the **Operation** column.

Step 4 Set the configuration file parameters by referring to the following table.

Parameter	Description
Group	Group to which the configuration file belongs.
Configuration File Name	Name of the custom configuration file (for example, **config.yaml**) mounted to the VM mounting path.
Configuration File Version	Version of the configuration file to be bound to the component. **latest** is selected by default, indicating the latest version.
ConfigMap Mount Path	Location of the operating system files. Enter a relative path to the working directory. You can get the working directory location from the environment variable **${APP_HOME}**. Relative path for mounting the configuration file. Example: **./config**. The upper-level directory is **/opt/application/*{CAS_INSTANCE_NAME}*/*{CAS_INSTANCE_VERSION}*/*{CAS_INSTANCE_ID}***, which stores the environment dependency, installation, startup, and stop scripts, and software packages required for component deployment. • **{CAS_INSTANCE_NAME}** is the component instance name. • **{CAS_INSTANCE_VERSION}** is the component instance version. • **{CAS_INSTANCE_ID}** is the component instance ID.

Step 5 Click **OK**.

Step 6 (Optional) Select a created key from the **Associate Key** drop-down list to associate the key with the configuration file.

If sensitive information such as passwords needs to be encrypted in your configuration items, you can interconnect with DEW and use a custom key to encrypt and decrypt sensitive information.

This parameter is available only when Data Encryption Workshop (DEW) is deployed in the region where ServiceStage is deployed and a CCE cluster is bound to the component deployment environment.

For details about how to create a key, see **Creating a Key**.

----End

Unbinding a Bound Configuration File

Step 1 Choose **Application Settings**.

Step 2 Select the target configuration file in the configuration file list.

Step 3 Click **Remove** in the **Operation** column.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.16.3 Configuring Application Performance Management of a VM-Deployed Component

Performance management helps you quickly locate problems and identify performance bottlenecks to improve your experience. Selecting Java probe will start APM and install Java probes on the nodes deployed with APM, which consumes a small amount of resources. Java probes use the bytecode enhancement technology to trace Java application calls and generate topology and call chain data.

Precautions

- This function can be enabled only when APM of the corresponding version is deployed and enabled in the environment.
- JDK 8, 11, 17, and 21 are supported.
- Tomcat 8.x and 9.x are supported. For details, see **Usage Restrictions**.
- Performance management can be configured for Java applications whose technology stack type is Java or Tomcat.

Configuring Application Performance Management of a VM-Deployed Component

Step 1 Choose **Advanced Settings** > **Performance Management**.

Step 2 Select **Java probe** and set probe parameters based on your probe type by referring to the following table.

Probe Type	Description
APM 2.0	1. Set **Application**. – Select the application that has the same name as the application where the application component is located from the drop-down list. – If the application does not exist, click **Create Application** to create it. After the application is created, you can log in to the APM console and view the new application in the application list. 2. Set **Probe Version**. The value is fixed to **latest**, indicating that the latest version is used. 3. Set **Access Key**. The access key is automatically obtained. If the access key cannot be automatically obtained, manually enter it.

----End

5.16.4 Configuring Log Settings of a VM-Deployed Component

For VM-deployed components, you can view run logs on the AOM console. ServiceStage also allows you to configure log settings of a component deployed

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

on an ECS bound to the environment using a VM and collect the component run logs to LTS for viewing and processing.

Configuring Log Settings of a VM-Deployed Component

Step 1 Choose **Log Settings**.

Step 2 Determine whether to collect component run logs to LTS by referring to the following table.

Whether to Collect Component Run Logs to LTS	Description
Yes	1. Select **Auto connection to LTS**. 2. Enter a **Log Collection Path**. – Click **Add Path** to set multiple log data collection sources. – If no log collection path is set, logs in the **/var/log/{instanceName}/*/start_app.log** directory are collected by default. **instanceName** indicates the name of the current component instance. After a component is configured and deployed, the system automatically generates log group and log stream configurations, and generates access configurations and host groups on the **Ingestion Management** page of the LTS console. • **Log Group**: The naming rule is **servicestage-***{Application ID}*. Use the ID of the application to which the component belongs. • **Log Stream**: The naming rule is **lts-topic-***{Component name}*. Use the name of the component connected to LTS. • **Ingestion**: The naming rule is **access-config-***{Component name}*. Use the name of the component connected to LTS. • **Host Group**: The naming rule is **vm-***{Instance ID}*. Use the ID of the component instance connected to LTS.
No	Do not select **Auto connection to LTS**. • After the component is configured and deployed, you can view running logs on the AOM console. For details, see **Viewing Log Files**. • After the component is configured and deployed, you can collect component logs to LTS by referring to **Managing Component LTS Logs**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5.17 Viewing Component Details

After the component is created and deployed, you can view the component details on its **Overview** page.

Prerequisites

- To view the YAML configurations of a CCE-deployed component, enable **View YAML Mode**. The parameters in the YAML configuration file are described in **Deployment**.
- Click **Synchronize Workload** to synchronize latest configurations after CCE-based component workload changes (such as workloads upgraded using CCE).

Viewing Component Details

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the component **Overview** page.

- On the **Application Management** page, click the application to which the component belongs, and click the target component in **Component List**.
- On the **Component Management** page, click the target component.

Step 3 On the component **Overview** page, view the component status and other details.

For details about component status description, see **Table 5-28**.

- Click a component in a state except **Draft** to view the deployment records.
- Click a component in the **Draft** state. You can continue to create and deploy components by referring to **Creating a Component Based on a Container Using Manual Configurations**, **Creating a Component Based on a Container Using YAML Configurations**, or **Creating a Component Based on a VM**.

Table 5-28 Component statuses

Status	Description
Initializing	When **Creating a Component Based on a Container Using Manual Configurations**, **Creating a Component Based on a Container Using YAML Configurations**, **Creating a Component Based on a VM**, or **Creating a Component Using a Component Template**, the component is in this state after you click **Create and Deploy**.
Upgrading/ Rolling back the component	The component is in this state when **Upgrading a Component in Single-batch Release Mode**, **Upgrading a Component in Rolling Release Mode**, **Upgrading Components in Batches**, **Rolling Back a Component**, **Redeploying a Component in Single-batch Release Mode**, **Upgrading a Component Using a Component Template**, or **Redeploying a Component in Rolling Release Mode**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Status	Description
Scaling	After **Configuring a Manual Scaling Policy**, the component is in this state during policy execution.
Failed	The operation on the component fails. For example, component deployment, upgrade, or rollback fails. You can click **Failed** to go to the **Deployment Records** list to view the failure cause.
Running	The operation on the component is successful, and the component status is normal. For example, component deployment, upgrade, or rollback is successful.
Deleting	The component is in this state when **Deleting a Component**.
Starting	The component is in this state during the start process by referring to **Starting or Stopping a Component**.
Stopping	The component is in this state during the stop process by referring to **Starting or Stopping a Component**.
Stopped	The component is in this state after being stopped by referring to **Starting or Stopping a Component**.
Restarting	The component is in this state during the restart process by referring to **Starting or Stopping a Component**.
Not ready	The component is in this state when the number of running component instances is less than the total number of component instances.
Unknown	A container-deployed component is in this state after **Unknown** is returned during synchronization from its CCE.
Abnormal	A container-deployed component is in this state after its CCE cluster or workload is deleted.
Releasing	The component is in this state when **Upgrading a Component in Dark Launch Mode** or **Redeploying a Component in Dark Launch Mode**.
Gray Release Failed	The component is in this state after **Upgrading a Component in Dark Launch Mode** or **Redeploying a Component in Dark Launch Mode** fails.
Draft	The component is being created and the parameter configuration is not complete. It is used to prevent the configured parameters from being lost due to factors such as page closing. When **Creating a Component Based on a Container Using Manual Configurations**, **Creating a Component Based on a Container Using YAML Configurations**, or **Creating a Component Based on a VM**, the component is in this state after you click **Save as Draft**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Status	Description
Imported and to be restarted	The component is waiting to restart after **Creating a Component by Importing a CCE Workload**.
Imported and restarting	The component is restarting after **Creating a Component by Importing a CCE Workload**.
Restart failed	The component is in this state if it fails to be restarted by referring to **Starting or Stopping a Component**.
Processing	The container-deployed component whose **Workload Type** is **Job** is in this state when each job is processed in sequence.
Completed	The container-deployed component whose **Workload Type** is **Job** is in this state when all jobs have been processed.

----End

5.18 Viewing the Component Running Environment

After a component is successfully deployed, you can view the resources (such as CCE clusters and microservice engines) on which the component depends and the resource status and usage on the component **Infrastructure** page.

Viewing the Component Running Environment

Step 1 Log in to ServiceStage.

Step 2 Use either of the following methods to go to the **Infrastructure** page.

- On the **Application Management** page, click the application to which the component belongs, click the target component in **Component List**, and choose **Infrastructure** in the left navigation pane.
- On the **Component Management** page, click the target component and choose **Infrastructure** in the left navigation pane.

Step 3 On the component **Infrastructure** page, select component running resources and view the resource status and usage.

----End

5.19 Deleting a Component

This section describes how to delete a component that is no longer used.

- Deleting a component will also delete the automatically created LTS access configuration and host group.
- After a component is deleted, the services provided by the component are unavailable.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

The component status is not **Initializing**, **Upgrading/Rolling back the component**, **Scaling**, **Deleting**, **Starting**, **Stopping**, **Releasing**, or **Imported and restarting**. For details about how to check the component status, see **Viewing Component Details**.

Deleting a Component

Step 1 Log in to ServiceStage.

Step 2 On the **Application Management** page, click the application to which the component belongs.

Step 3 In **Component List**:

- Single deletion: Locate the component to be deleted and choose **More** > **Delete** in the **Operation** column.
- Batch deletion: Select the components to be deleted and click **Bulk Delete**.

  Components created by **Creating a Component Using a Component Template** cannot be deleted in batches.

Step 4 In the displayed dialog box, perform the following operations based on the component creation method:

- For components created by **Creating a Component Based on a Container Using Manual Configurations**, **Creating a Component Based on a Container Using YAML Configurations**, **Creating a Component Based on a VM**, or **Creating a Component by Importing a CCE Workload**, go to **Step 5**.
- For container-deployed components created by **Creating a Component Using a Component Template**, determine whether to select **Delete configuration file** and **Delete Kubernetes resources** and go to **Step 5**.
- For VM-deployed components created by **Creating a Component Using a Component Template**, determine whether to select **Delete configuration file** and go to **Step 5**.

Step 5 In the displayed dialog box, click **OK**.

After a container-deployed component whose technology stack is Java, Tomcat, Node.js, Python, or PHP is deleted, the build job created by the system is also deleted. For details, see **Build Job Overview**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

6 Configuration Management

6.1 Configuration Overview

If multiple components of different applications need to be deployed in different environments, each component must be configured based on the environment and application when you create or upgrade them. This is complex and error-prone. If the component configuration is incorrect, the deployment or upgrade will fail, which severely affects services.

To solve the preceding problems, ServiceStage provides configuration management. By creating a unified configuration, you can fill the system variables (such as the IP address, port number, database address, and application name associated with the environment) of the component environment and application in the configuration to generate a configuration file. When a component is associated with a configuration file for deployment, the system variables are automatically replaced with the actual value. This implements multi-environment use with one-time configuration through file mounting.

System Variables of a Configuration File

For details about the system variables that can be written into a configuration file, see **Table 6-1**.

Table 6-1 System variables

Variable Name	Description
PAAS_PROJECT_ID	Project ID.
CAS_ENVIRONMENT_ID	Environment ID.
CAS_ENVIRONMENT_NAME	Environment name.
CAS_APPLICATION_ID	Application ID.
CAS_APPLICATION_NAME	Application name.
CAS_COMPONENT_ID	Component ID.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Variable Name	Description
CAS_INSTANCE_ID	Component instance ID.
CAS_INSTANCE_NAME	Component instance name.
CAS_INSTANCE_VERSION	Component instance version.
PAAS_PROJECT_NAME	Project name.
AOM_DISCOVERY_RULE_VERSION	AOM discovery rule name.
PAAS_REGION_NAME	Region name.
PAAS_CSE_CC_ENDPOINT	CSE configuration center address.
PAAS_CSE_SC_ENDPOINT	CSE registry center address.
DISTRIBUTED_CACHE_PASSWORD	Distributed cache password.
DISTRIBUTED_CACHE_TYPE	Distributed cache type.
DISTRIBUTED_CACHE_PORT	Distributed cache port.
DISTRIBUTED_CACHE_HOST	Distributed cache address.
DISTRIBUTED_CACHE_VERSION	Distributed cache version.
DISTRIBUTED_CACHE_CLUSTER	Distributed cache cluster type.
RELATIONAL_DATABASE_PASSWORD	Cloud database password.
RELATIONAL_DATABASE_DB_TYPE	Cloud database type.
RELATIONAL_DATABASE_JNDI_NAME	Cloud database JNDI name.
RELATIONAL_DATABASE_HOST	Cloud database address.
RELATIONAL_DATABASE_CONNECTION_TYPE	Cloud database connection type.
RELATIONAL_DATABASE_DB_NAME	Cloud database name.
RELATIONAL_DATABASE_DB_USER	Cloud database account.
RELATIONAL_DATABASE_PORT	Cloud database port.
RELATIONAL_DATABASE_NAME	Cloud database name.
RELATIONAL_DATABASE_VERSION	Cloud database version.

Configuration File Application Scenarios

The configuration file created through configuration management can be used in the following scenarios:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Creating a Component Based on a Container Using Manual Configurations
- Creating a Component Based on a VM
- Upgrading a Single Component
- Upgrading Components in Batches
- Cloning Components in Batches
- Creating a Release Task
- Editing a Release Task
- Cloning a Release Task
- Creating a Region Release Task Using Manual Configurations
- Creating a Region Release Task Using a Release Task Template
- Editing a Region Release Task Created Using Manual Configurations
- Editing a Region Release Task Created Using a Release Task Template
- Cloning a Region Release Task Created Using Manual Configurations
- Cloning a Region Release Task Created Using a Release Task Template

6.2 Managing Groups

Groups are used to manage configuration files by category.

Creating a Group

A maximum of 100 groups can be created in a project.

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **Group** > **Create Group**.

Step 3 Configure the group information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Group Name	Group name, which must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see Enabling the Enterprise Project Function. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see Creating an Enterprise Project

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Tag	Tags are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use tags to classify them based on usage, owner, or environment. If your organization has configured tag policies for ServiceStage, add tags to trackers based on the policies. If a tag does not comply with the tag policies, group creation may fail. Contact your administrator to learn more about tag policies. Each group can have a maximum of 20 labels. 1. Enter the key and value of the tag. – The key contains 1 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). – The value contains 0 to 63 characters. Start and end with a letter or digit. Only use digits, letters, underscores (_), hyphens (-), and periods (.). It is recommended that you **create a predefined tag** on TMS to add the same tag to different resources. 2. Click **OK**.

Step 4 Click **OK**.

----End

Deleting a Group

You can delete a group that is no longer used.

The group to be deleted is not associated with any configuration file. Otherwise, **Deleting a Configuration File** to disassociate the configuration group from the configuration file.

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **Group**.

Step 3 Click **Delete** in the **Operation** column of the target group.

----End

6.3 Managing Configuration Files

6.3.1 Creating a Configuration File

This section describes how to create a configuration file and fill the system variables of the component environment and application in the variables in the configuration file, to deploy and upgrade multiple components of different applications in different environments.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

A maximum of 1000 configuration files can be created in a project.

Creating a Configuration File

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List** > **Create Configuration**.

Step 3 In the **Basic Info** area, configure the configuration file by referring to the following table.

Parameter	Description
Configuration Name	Configuration file name, which cannot be modified after being created. Only use letters, digits, underscores (_), or hyphens (-) Enter 1 to 64 characters. The name extension is optional and can contain up to 17 characters, including dots (.).
Group	Groups are used to manage configuration files by category. The configuration file name must be unique in a group. • Select a group from the drop-down list. • Click **Create Group** to create a group by referring to **Creating a Group**.
Description	Configuration file description. 1. Click . 2. Enter the configuration file description, which can contain up to 128 characters. 3. Click .

Step 4 In the **Configuration Content** area, set configuration file parameters by referring to the following table.

Parameter	Description
Associate Key	If sensitive information such as passwords needs to be encrypted in your configuration items, you can interconnect with DEW and use a custom key to encrypt and decrypt sensitive information. 1. Click **Bind Associated Key**. 2. Select a created key from the **Key** drop-down list. To use a new key, click **Create a Key**. For details, see **Creating a Key**. 3. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Mount Mode	This parameter is available only for container-deployed components. • **secret**: If the configuration file contains sensitive data such as passwords, the configuration file can be mounted in this mode. • **configmap**: If the configuration file does not contain sensitive data, the configuration file can be mounted in this mode.

Step 5 In the **Configuration Content** area, set configuration file parameters by referring to the following table.

Method	Description
Manual enter	Manually enter the configuration file content in the editing area. Determine whether to encrypt variables (for example, **xxx**). • **xxx** indicates that encryption is disabled. During component deployment, actual values will be used as variables. • **#{xxx}** indicate that encryption is enabled. During component deployment, variables' actual values will be encrypted using a DEW key.
Modify the example	Modify the example provided by ServiceStage to set the configuration file. 1. Click **View Example**. The **Example** dialog box is displayed. 2. Click to copy the example. 3. Click to close the **Example** dialog box. 4. Paste the copied example to the editing box in **Configuration Content**, and edit the example. Determine whether to encrypt variables (for example, **${xxx}**). – **${xxx}** indicates that encryption is disabled. During component deployment, actual values will be used as variables. – **#{${xxx}}** indicate that encryption is enabled. During component deployment, variables' actual values will be encrypted using a DEW key.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Method	Description
Insert system variables	ServiceStage provides variables related to environments, applications, and components, as shown in Table 6-1. You can use these system variables in the configuration file. 1. Click **System Variables**. 2. Move the cursor to the variable to be written into the configuration file in the variable list, for example, **xxx**. – Click **Insert** to insert **${xxx}** into the editing area under **Configuration Content**. During component deployment, actual values will be used as variables. – Click **Insert and Encrypt** to insert **#{${xxx}}** to the editing area under **Configuration Content**. **#** indicates that encryption is enabled. During component deployment, variables' actual values will be encrypted using a DEW key.

Step 6 (Optional) Click **Verify Content** and modify the configuration as prompted.

Step 7 Click **Release**. The system automatically verifies the content correctness. Modify the content as prompted.

After the configuration file is created, you can view it in the **Configuration Management** list.

----End

6.3.2 Importing a Configuration File

This section describes how to import a local configuration file to deploy and upgrade multiple components of different applications in different environments.

Restrictions

- The configuration file to be imported must be compressed into a .zip package, and the package size cannot exceed 500 KB.

  The folder name in the compressed package is the configuration group name, and the file name in the folder is the configuration file name. The file can be in the YAML or Properties format.
- The package to be imported can contain up to 500 configuration files.

Importing a Configuration File

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List** > **Import**. The **Import Configuration** dialog box is displayed.

Step 3 Use either of the following methods to upload the configuration file:

- Click **click Upload** and select the target configuration file.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Drag the configuration file to the area in the **Import Configuration** dialog box.

You can click **Download Template** and prepare the configuration file based on the format of the downloaded template for import.

Step 4 Set **Policy for Processing Same Configuration**. This parameter is used to determine the policy for processing configuration files with the same name in the imported package and existing configuration files with the same name in the group.

- **Skip**: Skip the import.
- **Overwrite**: Replace the existing configuration file with the imported configuration file.

Step 5 Click **Import**.

- If the import is successful, you can view all the imported configuration files in the **Configuration Management** list.
- If the import failed, modify the configuration file as prompted and import it again.

----End

6.3.3 Exporting a Configuration File

This section describes how to export the current version of a configuration file so that you can view, edit, and share the configuration file locally.

Prerequisites

The configuration file to be exported is available.

- The configuration file has been created. For details, see **Creating a Configuration File**.
- The configuration file has been imported. For details, see **Importing a Configuration File**.

Exporting a Configuration File

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List**.

Step 3 Select a configuration file export method by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Export Method	Procedure	Result
Batch export	1. Select the configuration files. 2. Click **Export**.	• The exported configuration file is named *{Timestamp}*.**zip**, for example, **1743642593371.zip**. The timestamp (for example, **1743642593371**) is the time when you click **Export**. • The name of the level-1 file directory in the package is the name of the enterprise project to which the configuration group of the configuration file to be exported belongs. • The name of the level-2 file directory is the name of the configuration group to which the configuration file to be exported belongs. • The configuration file in the level-2 directory is the configuration file to be exported.
Single export	1. Click a configuration file. 2. In the **File Content** area, click **Download File**.	• The exported configuration file is named *{Timestamp}*.**zip**, for example, **1743642593371.zip**. The timestamp (for example, **1743642593371**) is the time when you click **Download File**. • The name of the file directory in the package is the name of the configuration group to which the configuration file to be exported belongs. • The configuration file in the directory is the configuration file to be exported.

----End

6.3.4 Editing a Configuration File

This section describes how to edit a configuration file.

Prerequisites

The configuration file to be edited is available.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- The configuration file has been created. For details, see **Creating a Configuration File**.
- The configuration file has been imported. For details, see **Importing a Configuration File**.

Editing a Configuration File

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List**.

Step 3 Use either of the following methods to go to the editing page:

- Directly edit: Click **Edit** in the **Operation** column of the target configuration file.
- Edit after viewing details: Click the target configuration file, view the details, and click **Edit**.

Step 4 In the **Basic Info** area, modify the configuration description.

1. Click .
2. Enter the configuration file description, which can contain up to 128 characters.
3. Click .

Step 5 In the **Configuration Content** area, set configuration file parameters by referring to the following table.

Parameter	Description
Associate Key	If sensitive information such as passwords needs to be encrypted in your configuration items, you can interconnect with DEW and use a custom key to encrypt and decrypt sensitive information. 1. Click **Bind Associated Key**. 2. Select a created key from the **Key** drop-down list. To use a new key, click **Create a Key**. For details, see **Creating a Key**. 3. Click **OK**.
Mount Mode	This parameter is available only for container-deployed components. • **secret**: If the configuration file contains sensitive data such as passwords, the configuration file can be mounted in this mode. • **configmap**: If the configuration file does not contain sensitive data, the configuration file can be mounted in this mode.

Step 6 In the **Configuration Content** area, set configuration file parameters by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Method	Description
Manual enter	Manually enter the configuration file content in the editing area. Determine whether to encrypt variables (for example, **xxx**). • **xxx** indicates that encryption is disabled. During component deployment, actual values will be used as variables. • **#{xxx}** indicate that encryption is enabled. During component deployment, variables' actual values will be encrypted using a DEW key.
Modify the example	Modify the example provided by ServiceStage to set the configuration file. 1. Click **View Example**. The **Example** dialog box is displayed. 2. Click to copy the example. 3. Click to close the **Example** dialog box. 4. Paste the copied example to the editing box in **Configuration Content**, and edit the example. Determine whether to encrypt variables (for example, **${xxx}**). – **${xxx}** indicates that encryption is disabled. During component deployment, actual values will be used as variables. – **#{${xxx}}** indicate that encryption is enabled. During component deployment, variables' actual values will be encrypted using a DEW key.
Insert system variables	ServiceStage provides variables related to environments, applications, and components, as shown in **Table 6-1**. You can use these system variables in the configuration file. 1. Click **System Variables**. 2. Move the cursor to the variable to be written into the configuration file in the variable list, for example, **xxx**. – Click **Insert** to insert **${xxx}** into the editing area under **Configuration Content**. During component deployment, actual values will be used as variables. – Click **Insert and Encrypt** to insert **#{${xxx}}** to the editing area under **Configuration Content**. **#** indicates that encryption is enabled. During component deployment, variables' actual values will be encrypted using a DEW key.

Step 7 (Optional) Click **Verify Content** and modify the configuration as prompted.

Step 8 Click **Release**. The system automatically verifies the content correctness. Modify the content as prompted.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

After the configuration file is edited, you can view the version details of the configuration file in the **Configuration Management** list. For details, see **Viewing Version Details of a Configuration File**.

----End

6.3.5 Viewing Version Details of a Configuration File

This section describes how to view the basic information and configuration details of a configuration file version.

Viewing Current Version Details of a Configuration File

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List**.

Step 3 Click the target configuration file to view the basic information and configuration details of the current version.

- Click the **Configuration Files** tab to view the configuration file details.
- Click the **Associated Components** tab to view the component name, application to which the component belongs, and environment to which the component belongs.

----End

Viewing Historical Version Details of a Configuration File

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List**.

Step 3 Use either of the following methods to open the **Historical Version** dialog box:

- Click **View History** in the **Operation** column of the target configuration file.
- Click the target configuration file and click **View History**.

Step 4 Move the cursor to a historical version.

Step 5 Click **View** to view the basic information and configuration details of the configuration file of the historical version.

----End

6.3.6 Comparing Configuration Files

This section describes how to compare the differences between the current and historical versions of a configuration file.

Prerequisites

Either of the following operations has been performed on the configuration files to be compared:

- **Editing a Configuration File**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Rolling Back a Configuration File**

Comparing Configuration Files

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List**.

Step 3 Use either of the following methods to open the **Historical Version** dialog box:

- Click **View History** in the **Operation** column of the target configuration file.
- Click the target configuration file and click **View History**.

Step 4 Move the cursor to a historical version.

Step 5 Use either of the following methods to open the **Compare Version** dialog box and compare the configuration files of the historical and current versions.

- Directly compare: Click **Compare**.
- Compare after viewing the historical version configuration: Click **View** to view the configuration details of the historical version, and click **Comparison with the Current Version**.

----End

6.3.7 Rolling Back a Configuration File

This section describes how to roll back a configuration file to a historical version.

Prerequisites

Editing a Configuration File has been performed on the configuration file to be rolled back.

Rolling Back a Configuration File

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List**.

Step 3 Use either of the following methods to open the **Historical Version** dialog box:

- Click **View History** in the **Operation** column of the target configuration file.
- Click the target configuration file and click **View History**.

Step 4 Move the cursor to a historical version.

Step 5 Use either of the following methods to roll back the configuration file:

- Roll back after version comparison: Click **Compare** to compare the current and historical versions, and click **Roll Back to Current Version**.
- Roll back after viewing the historical version configuration: Click **View** to view the configuration details of the historical version, and click **Roll Back to Current Version**.

Step 6 Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Compare the historical version selected in **Step 4** with the latest version by referring to **Comparing Configuration Files**. If the configurations are the same, the rollback is successful.

----End

6.3.8 Pushing a Configuration File Version to an Associated Component

Push the configuration file of a specified version to an associated component to update the component application configuration.

Prerequisites

- The configuration file to be pushed has been associated with a component.
  - For container-deployed components, see **Managing Application Settings of a Container-Deployed Component**.
  - For VM-deployed components, see **Managing Application Settings of a VM-Deployed Component**.
- The target component is in the **Running**, **Releasing**, or **Not ready** state. For details about how to check the component status, see **Viewing Component Details**.

Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List**.

Step 3 Click the target configuration file.

Step 4 Click the **Associated Components** tab.

Step 5 Select the target associated component and click **Push** in the **Operation** column.

Step 6 Select a **Target Version**.

The mount mode of the configuration file version must be the same as that of the associated component.

Step 7 Click **OK**.

Wait until the **Push Status** changes from **Pushing** to **Pushed**, indicating that the configuration file version has been successfully pushed to the associated component.

Step 8 Click the component. The **Overview** page is displayed.

Step 9 Click **Restart**.

Step 10 Click **OK**.

Wait until the component status changes from **Restarting** to **Running**, indicating that the component is restarted successfully.

Step 11 In the **Configurations** area, expand **Application Settings** and check **Current Component Version**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If the version is the same as the version selected in **Step 6**, the operation is successful.

----End

6.3.9 Deleting a Configuration File

This section describes how to delete a configuration file that is no longer used.

Deleted configuration files cannot be restored. Exercise caution when performing this operation. You can export the configuration file before deleting it. For details, see **Exporting a Configuration File**. When the configuration file is required, import it by referring to **Importing a Configuration File**.

Prerequisites

- The configuration file to be deleted is available.
  - The configuration file has been created. For details, see **Creating a Configuration File**.
  - The configuration file has been imported. For details, see **Importing a Configuration File**.
- If the configuration file to be deleted has been associated with a component, **Deleting a Component** to remove the association between the configuration file and the component before deleting it.

Deleting a Configuration File

Step 1 Log in to ServiceStage.

Step 2 Choose **Configuration Management** > **List**.

Step 3 Use either of the following methods to delete the configuration file:

- Directly delete: Click **Delete** in the **Operation** column of the target configuration file.
- Delete after viewing details: Click the target configuration file, view the details, and click **Delete**.

Step 4 Click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

7 Release Management

7.1 New Release Management

7.1.1 Overview

Description

In addition to component deployment, upgrade, and cloning provided by the **Release Management of the old version**, the region release tasks of the new version provide application management (application creation, component creation, component update, and build) and checkpoint tasks (manual approval and manual trigger).

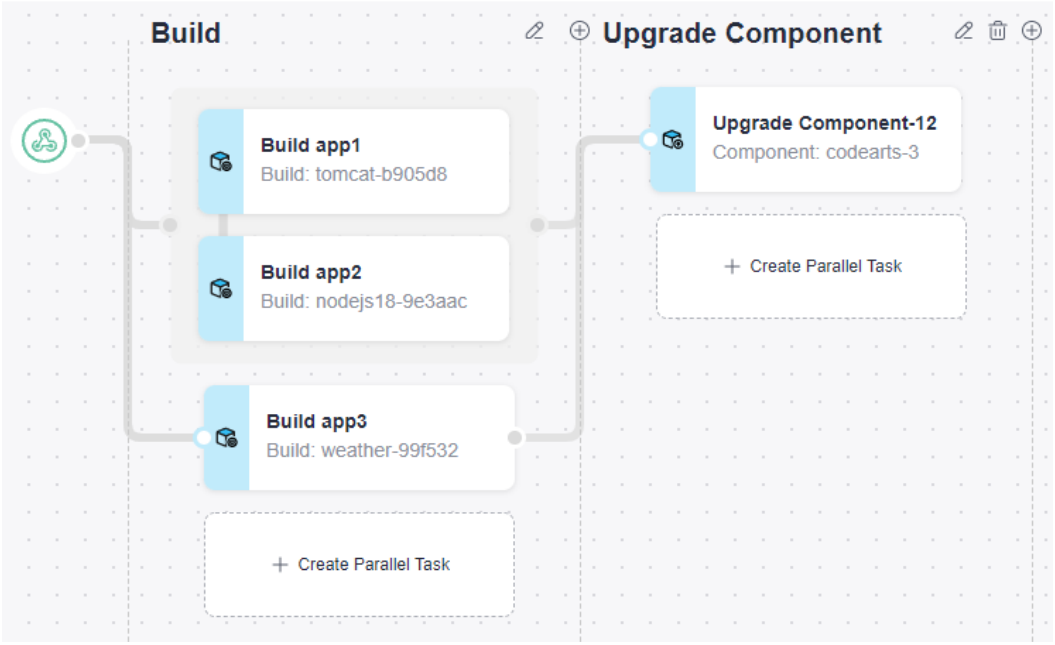
- Application Management
  - **Create Application**: Add an application creation task to the release task.
  - **Create Component**: Add a component creation task to the release task.
  - **Update Component**: Add a component update task to the release task.
  - **Build**: Add a component build task to the release task.
- Checkpoint Tasks
  - **Manual Approval**: If a manual approval checkpoint task is set, the release task process can continue only after being approved by related approvers.
  - **Manual Trigger**: If a manual trigger checkpoint task is set, the release task process can continue only after you manually confirm it.

In addition to manual release task creation on the UI, release management of the new version also supports automatic release task creation based on the release task template. For details, see **Creating a Region Release Task Using a Release Task Template**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release Task Structure

Figure 7-1 Example of the region release task structure



As shown in **Figure 7-1**, a region release task consists of multiple phases (for example, building and upgrading a component). A phase is a set of tasks that need to be executed in sequence in a release task.

A phase can contain multiple groups. A group combines a series of tasks that need to be executed in sequence in the same phase. Tasks in the same group are executed in sequence, and tasks in different groups in the same phase are executed concurrently. For example, the **Build app1** and **Build app2** tasks in the **Build** stage form a group and are executed in sequence. The **Build app3** task forms another group and is executed concurrently with the former group.

ServiceStage supports up to 1000 region release tasks in the same project.

A region release task can contain up to 20 phases. A phase can contain up to 20 groups. A group can contain up to 20 tasks.

7.1.2 Creating a Region Release Task

7.1.2.1 Creating a Region Release Task Using Manual Configurations

ServiceStage supports up to 1000 region release tasks in the same project.

Create a region release task using manual configurations to combine applications, components, and build tasks in the same region in different phases as required and orchestrate and release them.

Procedure

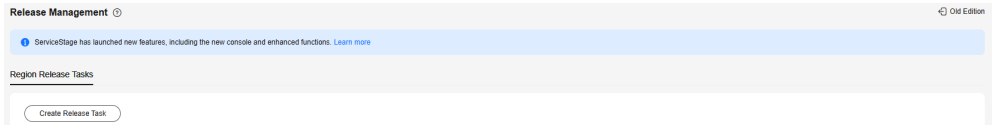
Step 1 Log in to ServiceStage.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Click **Create Release Task** and set release task parameters by referring to the following table.

Parameter	Description
Release Task	The release task name must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**
Description	Description of a release task. 1. Click . 2. Enter up to 128 characters. 3. Click .

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Release Scenario	Fixed to **Custom Orchestration**, indicating to customize the orchestration of tasks under a release task. This is used for heavy or stable traffic.
Example	Whether to use the example to create a release task. • If it is not selected, the example is not used to create a release task. • If it is selected, the example is used to create a release task. – **Simple Example**: Automatically add a build model and select an existing build task created by the system. – **Common Example**: Automatically add the build-upgrade component model, select an existing build task created by the system, and use the generated software package or image package as the source of the software package or image package of the component to be upgraded in the component upgrade task.

Step 6 Click **OK**.

Step 7 Configure the release task phase and task.

- If **Example** is not selected in **Step 5**, go to **Step 8**.
- If you select **Example** and **Simple Example** in **Step 5**, go to **Step 9**.
- If you select **Example** and **Common Example** in **Step 5**, go to **Step 10**.

Step 8 If **Example** is not selected in **Step 5**, perform the following operations:

1. Set **Best Effort Policy**.
   - Enable it to continue releasing other components when a component fails to be released.
   - If it is disabled, the release task fails when a component fails to be released.
2. Click next to the phase name and enter a **Phase Name**, for example, Create Application.
3. Click **Create Task** and create a task by referring to **Table 7-1**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 7-1 Creating a task

Task Type	Description
Create Application	Add an application creation task to the release task. 1. Use either of the following methods to go to the **Create Application** page: ▪ Choose **All** > **Create Application** > **Create Now**. ▪ Choose **Application Management** > **Create Application** > **Create Now**. 2. Enter a task name containing 1 to 64 characters. 3. Enter an application name. The application name must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). 4. Select an enterprise project. Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. ▪ Click the drop-down list to select an existing enterprise project. ▪ Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project** 5. Click , enter up to 128 characters as the description, and click . 6. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
Create Component	Add a component creation task to the release task. 1. Use either of the following methods to go to the **Create Component** page: ■ Choose **All** > **Create Component** > **Create Now**. ■ Choose **Application Management** > **Create Component** > **Create Now**. 2. Select the application to which the component belongs from the **Application** drop-down list. 3. Select a component creation method. ■ **Create Component**: Go to **Step 8.3.d**. ■ **Clone Component**: Go to **Step 8.3.e**. 4. Click **Create Component** and perform the following operations to create a component: a. Enter a task name containing 1 to 64 characters. b. (Optional) Select a task in the previous phase that can be associated with the current task from the **Associated Task** drop-down list. For example, select the **Create Application** task created in the previous phase. c. Set component parameters. For container-deployed components, see **Creating a Component Based on a Container Using Manual Configurations**. For VM-deployed components, see **Creating a Component Based on a VM**. If you specify **Associated Task**, for example, the **Create Application** task created in the previous phase, the **Application** of the component is fixed to the application set in the selected **Create Application** task. d. Click **Deploy Now**. Select a created component from the component list and click **Set** to reset component parameters. Select a created component from the component list and click **Remove** to remove the component from the component creation task. e. Go to **Step 8.3.f**. 5. Click **Clone Component** and perform the following operations to configure the component: a. Select the component to be cloned from the component list and click **OK**. b. Click the component to be cloned in the component list and click **Set** to reset component parameters.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
	For container-deployed components, see **Creating a Component Based on a Container Using Manual Configurations**. For VM-deployed components, see **Creating a Component Based on a VM**. c. Go to **Step 8.3.f**. 6. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
Update Component	Add a component upgrade task to the release task. 1. Use either of the following methods to go to the **Update Component** page: ■ Choose **All** > **Update Component** > **Create Now**. ■ Choose **Application Management** > **Update Component** > **Create Now**. 2. Select the application to which the component belongs from the **Application** drop-down list. 3. Set the component image source. ■ If you enable **Image from Build**, you need to create a build task in the phase before the current task phase. Go to **Step 8.3.d**. ■ If you disable **Image from Build**, go to **Step 8.3.e**. 4. If you enable **Image from Build**, perform the following operations: a. Click **Add Component** and select only one container-deployed component to add to the upgrade task. The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**. b. Click **OK**. You can click **Change** to reselect a component. c. Select a build task from the **Build** drop-down list as the component image source for component upgrade. By default, the archive address of the image package output during component upgrade is the archive address set in the selected build task displayed in the **Build Output** drop-down list. d. Go to **Step 8.3.f**. 5. If you disable **Image from Build**, perform the following operations: a. Click **Select Components to Upgrade** and select the component to be added to the upgrade task. The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**. b. Click **OK**. You can click **Remove** on the card of a component in the list of selected components to be upgraded to remove the component from the list. Click **Set** on the card of the specified component in the list of selected components to be upgraded, enter a **Task Name** containing 1 to 64 characters, set

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
	component upgrade parameters by referring to **Step 5** to **Step 6** in **Upgrading Components in Batches**, and click **OK**. c. Go to **Step 8.3.f**. 6. Click **OK**.
Build	Add a job created for building images from source code, or a JAR, or WAR package to the release task. 1. Use either of the following methods to go to the **Build** page: ▪ Choose **All** > **Build** > **Create Now**. ▪ Choose **Application Management** > **Build** > **Create Now**. 2. Enter a task name containing 1 to 64 characters. 3. Select a build task from the **Build** drop-down list. 4. Click **OK**.
Manual Approval	Add a manual approval checkpoint task to the release task. If manual approval is set, the release task process can continue only after being approved by related approvers. 1. Use either of the following methods to go to the **Manual Approval** page: ▪ Choose **All** > **Manual Approval** > **Create Now**. ▪ Choose **Checkpoint Tasks** > **Manual Approval** > **Create Now**. 2. Enter a task name containing 1 to 64 characters. 3. Select the approver of the release task process from the **Approved By** drop-down list. If the approver list cannot be obtained, contact the administrator to create a custom policy that has the **iam:users.listUsers** permission of IAM, and bind the policy to the user group to which the current account belongs. 4. Set **Approval Mode**. ▪ **Approved by all**: The release task process can continue only after all selected approvers approve the task. Otherwise, the release fails. ▪ **Approved by any person**: The release task process can continue after being approved by any of the selected approvers. 5. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
Manual Trigger	Add a manual triggering task to the release task. If manual triggering is set, the release task process can continue only after you manually confirm it. Use either of the following methods to create a manually triggered task: – Choose **All** > **Manual Trigger** > **Create Now**. – Choose **Checkpoint Tasks** > **Manual Trigger** > **Create Now**.

- After a task is created, you can click **Create Parallel Task** to create a group and task in the current phase by referring to **Table 7-1**.
- Move the cursor to a specified task card in a group, click ⊕, and insert a new sequence to execute the tasks in the group by referring to **Table 7-1**.
- Move the cursor to a specified task card and click 🗑 to delete the task.

4. Click ⊕ next to the phase name to create a phase and create a group and task by referring to **Step 8**.

   To delete a specified phase, click 🗑 next to the phase name and click **OK**. Deleting a phase will delete all groups and tasks in the phase.
5. Go to **Step 11**.

Step 9 If you select **Example** and **Simple Example** in **Step 5**, perform the following operations:

1. Set **Best Effort Policy**.
   - Enable it to continue releasing other components when a component fails to be released.
   - If it is disabled, the release task fails when a component fails to be released.
2. (Optional) Click ✎ and enter a phase name.
3. Click **Create Build Task** and add a job created for building images from source code, or a JAR, or WAR package to the release task.
   a. Enter a task name containing 1 to 64 characters.
   b. Select a build task from the **Build** drop-down list.
   c. Click **OK**.
4. Go to **Step 11**.

Step 10 If you select **Example** and **Common Example** in **Step 5**, perform the following operations:

1. Set **Best Effort Policy**.
   - Enable it to continue releasing other components when a component fails to be released.
   - If it is disabled, the release task fails when a component fails to be released.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

2. (Optional) Click next to the **Build** phase and enter a phase name.
3. Click **Create Build Task** and add a job created for building images from source code, or a JAR, or WAR package to the release task.
   a. Enter a task name containing 1 to 64 characters.
   b. Select a build task from the **Build** drop-down list.
   c. Click **OK**.
4. (Optional) Click next to the **Upgrade Component** phase and enter a phase name.
5. Click **Create Component Upgrade Task** to create a component upgrade task.
   a. Select the application to which the component belongs from the **Application** drop-down list.
   b. Click **Add Component**.
   c. Select the component to be upgraded under the selected application.

      Only one container-deployed component whose status is **Running**, **Not ready**, or **Abnormal** can be added to the upgrade task. For details about how to check the component status, see **Viewing Component Details**.
   d. Click **OK**.
   e. Select the build task created in the previous step from the **Build** drop-down list as the component image source for component upgrade.

      By default, the archive address of the image package output during component upgrade is the archive address set in the selected build task displayed in the **Build Output** drop-down list.
   f. Click **OK**.
6. Go to **Step 11**.

Step 11 Perform subsequent operations based on whether to release the release task.

- If yes, click **Save and Execute**, select the task to be executed, and click **Execute**.

  If a checkpoint task is set in the release task, approve the task based on the checkpoint task type by referring to the following table.

Checkpoint Task Type	Approval Procedure
Manual Approval	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual approval task to be approved. 3. Click **Approve**. The **Manual Approval** dialog box is displayed. 4. Enter **Comments** to determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Checkpoint Task Type	Approval Procedure
Manual Trigger	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual triggering task to be approved. 3. Click **Approve**. The **Manual Trigger** dialog box is displayed. 4. Determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.

  - On the **Release Process View** page, you can view the execution status of the release process and wait until the release task is complete.
  - After the release task is saved, its **Release Status** is **Initialized**. For details about how to check the release task status, see **Viewing Region Release Task Details**.
  - After the release task is saved, you can manage the release task by referring to **Managing Region Release Tasks**.
- If no, click **Save**.
  - After the release task is saved, its **Release Status** is **Initialized**. For details about how to check the release task status, see **Viewing Region Release Task Details**.
  - After the release task is saved, you can manage the release task by referring to **Managing Region Release Tasks**.

----End

7.1.2.2 Creating a Region Release Task Using a Release Task Template

ServiceStage supports up to 1000 region release tasks in the same project.

Create a region release task using a release task template to combine applications, components, and build tasks in the same region in different phases as required and orchestrate and release them.

Release Task Template Description

The created template file needs to be compressed into a .zip package. A release task template package can contain multiple applications. An application can contain multiple components. Example directory structure of a release task template package:

```
release/
|-- app-1
|   |-- app.yaml
|   |-- component-1/
```

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

```
|   |   |-- spec.yaml
|   |   |-- variables.yaml
|   |   |-- values.yaml
|   |   |-- component/
|   |   |   |-- demo_component.yaml
|   |   |   |-- demo_config.yaml
|   |   |   |-- deployment.yaml
|   |   |   |-- configmap.yaml
|   |   |   |-- service.yaml
|   |   |--...(Other Kubernetes resource files)
|   |   |-- README.md
|   |-- component-2/
......
|-- app-2
......
```

release indicates the name of the released template package, **app-1** and **app-2** indicate the application directories in the template package, and **component-1** and **component-2** indicate the component directories of the application.

- Description of files in the application directory
  - **app.yaml**: application description file, which specifies the application name and the enterprise project to which the application belongs. Example file content:

    ```
    kind: ServiceStageApplication
    spec:
      name: test-application-zyh
      enterprise_project_id: 0
    ```

  - **spec.yaml**: application configuration information file, which contains the model version and model type. Example file content:

    ```
    # Model version number. Currently, only v1.0.0 is available.
    model_version: v1.0.0

    # Model type. Application indicates an application model.
    model_type: Application
    ```

  - **values.yaml**: file that describes the default values of variables. Example file content:

    ```
    namespace: demo
    dep_name: demo-dep
    svc_port: 8080
    value: World
    image: swr.example.com/ss/demo-app:latest
    ```

  - **variables.yaml**: supported variable declaration file. All variables are declared in this file, including the variable name, type, description, and default value. Example file content:

    ```
    variable:
      namespace:
        type: string
        default: demo
        description: namespace
      dep_name:
        type: string
        default: demo-dep
        description: workload name
      svc_port:
        type: integer
        default: 8080
        description: service port number
      value:
        type: string
        default: World
        description: Hello interface response data
      cce_resource:
        type: cce
    ```

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

```
  description: CCE resource
image:
  type: string
  description: workload image
```

- Description of files in the component directory

  See **Component Template Description**.

Prerequisites

- You have compressed the prepared release task template files into a ZIP package and uploaded the package to the repository. For details, see **Release Task Template Description**.
  - Upload the template package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Select the corresponding template package from OBS. Upload the template package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**.
  - If you need to use JFrog (example) as the repository for storing the template package, you can use an HTTP/HTTPS custom file download address as the template package download address. Upload the template package to the custom file address in advance.
- If the release task template contains environment creation and basic resource tasks, contact the administrator to grant the RFS permissions to the user group to which the operation account belongs by referring to the following table. In addition, grant the permission to create basic resources to the user group to which the operation account belongs. For details, see **Assigning Permissions to a User Group**.

ServiceStage Permission of the User Group to Which the Operation Account Belongs	Required RFS Permission of the User Group to Which the Operation Account Belongs
ServiceStage FullAccess	RF FullAccess
ServiceStage Administrator	
ServiceStage Developer	

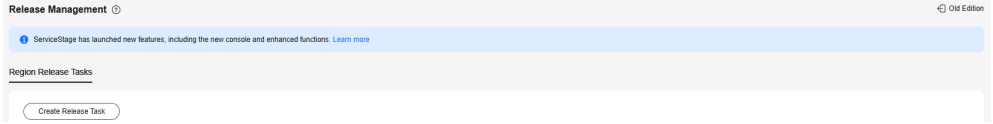
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.



 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Click **Use Template** and set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	The release task name must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
*Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**
Description	Description of a release task. 1. Click . 2. Enter up to 128 characters. 3. Click .

Step 6 Set **Set Template Package**. Select the uploaded release task template package based on the template package storage mode by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Package Repository	Procedure
CodeArts release repository	If PKIToken is disabled, the CodeArts software release repository is not supported. 1. Click **Select Software Package**. 2. Select the uploaded template package from the CodeArts release repository. 3. Click **OK**.
OBS	1. Click **Select Software Package**. 2. Select the uploaded template package from the OBS bucket. 3. Click **OK**.
SWR	If PKIToken is disabled, the SWR repository is not supported. 1. Click **Select Software Package**. 2. Select the uploaded template package from the SWR repository. 3. Click **OK**.
Custom file address	If you need to use JFrog (example) as the repository for storing the template package, perform the following operations to obtain the template package that has been uploaded to the custom file address: 1. Enter the HTTP/HTTPS custom file download address where the template package is located. The custom file address must be the complete path of the release task template package file, for example, **https://172.16.0.1:8082/artifactory/template-release.zip**. Click **Add Custom File Address** and enter the HTTP/HTTPS custom file download address where multiple template packages are located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

Step 7 Click **Next**.

- In the **Preview Template Package** area, you can view the details of the release task template file.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- In the **Set Parameters** area, you can view and modify the release task parameters.

Step 8 Click **Create and Deploy**.

- On the **Release Process View** page, you can view the release task process details.
- After the release task is created, its **Release Status** is **Initialized**. For details about how to check the release task status, see **Viewing Region Release Task Details**.
- After a release task is created, you can manage the release task by referring to **Managing Region Release Tasks**.

----End

7.1.3 Managing Region Release Tasks

7.1.3.1 Releasing a Region Release Task

The release process starts from the first task of the release task.

Prerequisites

The region release task cannot be in the **Running** state. For details about how to check the region release task status, see **Viewing Region Release Task Details**.

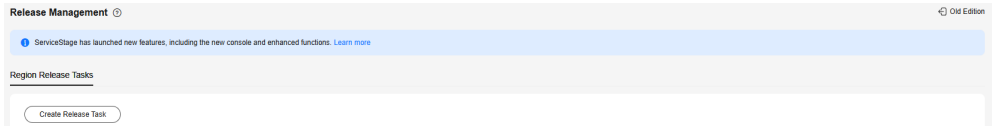
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 5 Use either of the following methods to release a release task:

- Directly release: Click **Execute** in the **Operation** column of the release task, select the task to be executed, and click **Execute Now**.
- Release after viewing details: Click the release task. After **Viewing Region Release Task Details**, click **Execute** in the upper right corner, select the task to be executed, and click **Execute Now**.

Step 6 If a checkpoint task is set in the release task, approve the task based on the checkpoint task type by referring to the following table.

Checkpoint Task Type	Approval Procedure
Manual Approval	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual approval task to be approved. 3. Click **Approve**. The **Manual Approval** dialog box is displayed. 4. Enter **Comments** to determine whether to pass the approval. – Click **Reject** to terminate the release task. The release task fails to be released. – Click **Agree** to continue the release.
Manual Trigger	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual triggering task to be approved. 3. Click **Approve**. The **Manual Trigger** dialog box is displayed. 4. Determine whether to pass the approval. – Click **Reject** to terminate the release task. The release task fails to be released. – Click **Agree** to continue the release.

- Wait until the release status changes from **Running** to **Successful**, which indicates that the release task is successfully released.
- For release tasks in the **Releasing** state, you can **Pausing the Release of a Region Release Task**.
- If the release fails, you can **Retrying a Region Release Task** or **Rolling Back a Region Release Task**.

----End

7.1.3.2 Pausing the Release of a Region Release Task

Pause a released region release task.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

The region release task is in the **Running** state. For details about how to check the region release task status, see **Viewing Region Release Task Details**.

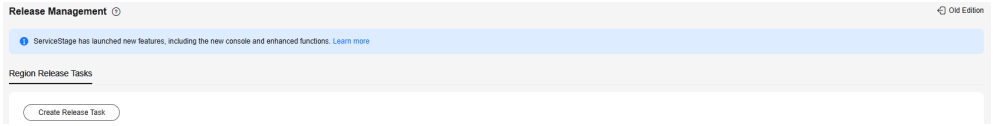
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Click the release task to be paused. The **Release Process View** page is displayed.

Step 6 Click **Stop** in the upper right corner.

Wait until the release status changes from **Running** to **Suspend**, which indicates that the release task is successfully stopped.

----End

7.1.3.3 Continuing to Release a Region Release Task

Continue to release a paused region release task.

Prerequisites

The region release task is in the **Suspend** state. For details about how to check the region release task status, see **Viewing Region Release Task Details**.

Procedure

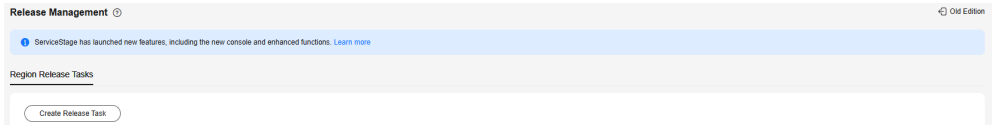
Step 1 Log in to ServiceStage.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Click the release task to be released. The **Release Process View** page is displayed.

Step 6 Click **Continue** in the upper right corner.

- If a checkpoint task is set in the release task, approve the task based on the checkpoint task type by referring to the following table.

Checkpoint Task Type	Approval Procedure
Manual Approval	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual approval task to be approved. 3. Click **Approve**. The **Manual Approval** dialog box is displayed. 4. Enter **Comments** to determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Checkpoint Task Type	Approval Procedure
Manual Trigger	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual triggering task to be approved. 3. Click **Approve**. The **Manual Trigger** dialog box is displayed. 4. Determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.

- Wait until the release status changes from **Suspend** to **Running**, which indicates that the release task is successfully continued.

----End

7.1.3.4 Retrying a Region Release Task

Retry a failed region release task. The release process continues from the failed task.

Prerequisites

The region release task is in the **Release failed** state. For details about how to check the region release task status, see **Viewing Region Release Task Details**.

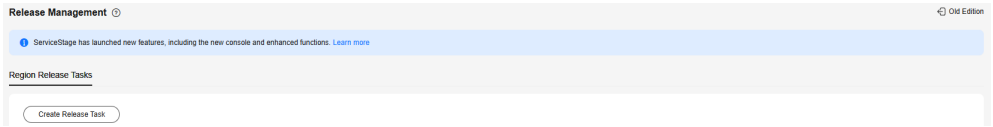
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Click the release task to be retried. The **Release Process View** page is displayed.

Step 6 Click **Retry** in the upper right corner.

- If a checkpoint task is set in the release task, approve the task based on the checkpoint task type by referring to the following table.

Checkpoint Task Type	Approval Procedure
Manual Approval	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual approval task to be approved. 3. Click **Approve**. The **Manual Approval** dialog box is displayed. 4. Enter **Comments** to determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.
Manual Trigger	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual triggering task to be approved. 3. Click **Approve**. The **Manual Trigger** dialog box is displayed. 4. Determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.

- Wait until the release status changes from **Running** to **Successful**, which indicates that the release task is successfully released.

----End

7.1.3.5 Rolling Back a Region Release Task

Roll back a released region release task to the status before execution.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

The region release task is in the **Successful** or **Release failed** state. For details about how to check the region release task status, see **Viewing Region Release Task Details**.

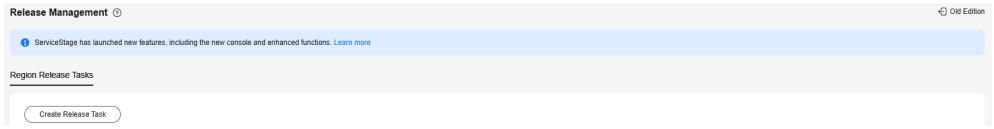
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Use either of the following methods to roll back a release task:

- Roll back directly: Click **Roll Back** in the **Operation** column of the release task in the **Successful** state.
- Roll back after viewing details: Click the release task. After **Viewing Region Release Task Details**, click **Roll Back** in the upper right corner.

Step 6 Click **OK**.

Wait until the release status changes from **Successful** or **Release failed** to **Rolled back**, which indicates that the release task is successfully rolled back to the status before execution.

----End

7.1.3.6 Editing a Region Release Task Created Using Manual Configurations

This section describes how to edit a region release task created by **Creating a Region Release Task Using Manual Configurations**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

The region release task created using manual configurations is not in the **Running** state. For details about how to check the region release task status, see **Viewing Region Release Task Details**.

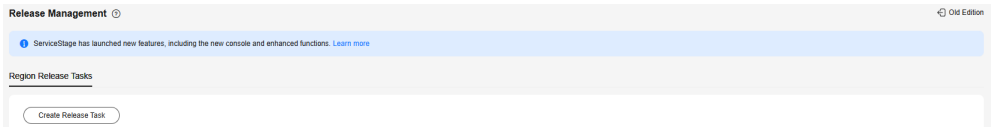
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Use either of the following methods to go to the page for editing a release task:

- Directly edit: Click **Edit** in the **Operation** column of the target release task.
- Edit after viewing details: Click the target release task. After **Viewing Region Release Task Details**, click **Orchestrate Release Task** in the upper right corner.

Step 6 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

Step 7 Click **Set Basic Info** in the upper right corner and set basic release task parameters by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Release Task	The release task name must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**
Description	Description of a release task. 1. Click . 2. Enter up to 128 characters. 3. Click .

Step 8 (Optional) Select another release task example.

Selecting another sample will clear all parameter settings of the release task. Exercise caution when performing this operation.

1. Click **Select Example** in the upper right corner and determine whether to use the example to create a release task.
   - If it is not selected, the example is not used to create a release task.
   - If it is selected, the example is used to create a release task.
     - **Simple Example**: Automatically add a build model and select an existing build task created by the system.
     - **Common Example**: Automatically add the build-upgrade component model, select an existing build task created by the system, and use the generated software package or image package as the source of the software package or image package of the component to be upgraded in the component upgrade task.
2. Click **OK**.
   - If **Example** is not selected, go to **Step 9**.
   - If you select **Example** and **Simple Example**, go to **Step 10**.
   - If you select **Example** and **Common Example**, go to **Step 11**.

Step 9 If **Example** is not selected, perform the following editing operations in the specified phase and then go to **Step 12**.

- Modify the phase name: Click next to the phase name and set **Phase Name**, for example, Create Application.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Delete a group task: Move the pointer to a task card in a group and click .
- Edit a group task: Click a task card in a group and edit the task by referring to **Table 7-2**.

Table 7-2 Editing a region release task

Operation	Description
Edit application creation	1. Enter a task name containing 1 to 64 characters. 2. Set the application name. The application name must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). 3. Select an enterprise project. Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. ▪ Click the drop-down list to select an existing enterprise project. ▪ Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project** 4. Click , enter up to 128 characters as the description, and click . 5. Click **OK**.
Edit component creation	1. Enter a task name containing 1 to 64 characters. 2. (Optional) Select a task in the previous phase that can be associated with the current task from the **Associated Task** drop-down list. For example, select the **Create Application** task created in the previous phase. 3. Set component parameters. For container-deployed components, see **Creating a Component Based on a Container Using Manual Configurations**. For VM-deployed components, see **Creating a Component Based on a VM**. If you specify **Associated Task**, for example, the **Create Application** task created in the previous phase, the **Application** of the component is fixed to the application set in the selected **Create Application** task. 4. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Edit component upgrade	– **Image from Build** is enabled when the **Update Component** task is created. 1. Select the application to which the component belongs from the **Application** drop-down list. 2. Click **Change**, select the component to be upgraded of the selected application again, and click **OK**. Only one container-deployed component can be added to the upgrade task. The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**. 3. Select a build task from the **Build** drop-down list as the component image source for component upgrade. By default, the archive address of the image package output during component upgrade is the archive address set in the selected build task displayed in the **Build Output** drop-down list. 4. Click **OK**. – **Image from Build** is disabled when the **Update Component** task is created. 1. Enter a task name containing 1 to 64 characters. 2. Set component upgrade parameters by referring to **Step 5** to **Step 6** in **Upgrading Components in Batches**. 3. Click **OK**.
Edit build	1. Enter a task name containing 1 to 64 characters. 2. Select a build task from the **Build** drop-down list. 3. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Edit manual approval	1. Enter a task name containing 1 to 64 characters. 2. Set **Approved By**. ▪ Add an approver: Select the approver of the release task process from the **Approved By** drop-down list. If the approver list cannot be obtained, contact the administrator to create a custom policy that has the **iam:users.listUsers** permission of IAM, and bind the policy to the user group to which the current account belongs. ▪ Delete an approver: Click × next to an approver. 3. Set **Approval Mode**. ▪ **Approved by all**: The release task process can continue only after all selected approvers approve the task. If any approver rejects the release, the release fails. ▪ **Approved by any person**: The release task process can continue after being approved by any of the selected approvers. 4. Click **OK**.

- Insert a task to be executed in a new sequence under a task group: Move the cursor to the task card in the group, click ⊕, and insert a task by referring to **Table 7-3**.
- Create a group and task: Click **Create Parallel Task**. In this phase, create a group and task by referring to **Table 7-3**.
- Create a phase and group: Click ⊕ next to a phase to create a phase and group. Click **Create Task** and create a group and task by referring to **Table 7-3**.
- Delete a phase: Click 🗑 next to the phase name and click **OK**.

  Deleting a phase will delete all tasks in the phase.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 7-3 Creating a task

Task Type	Description
Create Application	Add an application creation task to the release task. 1. Use either of the following methods to go to the **Create Application** page: ▪ Choose **All** > **Create Application** > **Create Now**. ▪ Choose **Application Management** > **Create Application** > **Create Now**. 2. Enter a task name containing 1 to 64 characters. 3. Enter an application name. The application name must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). 4. Select an enterprise project. Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. ▪ Click the drop-down list to select an existing enterprise project. ▪ Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project** 5. Click , enter up to 128 characters as the description, and click . 6. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
Create Component	Add a component creation task to the release task. 1. Use either of the following methods to go to the **Create Component** page: ■ Choose **All** > **Create Component** > **Create Now**. ■ Choose **Application Management** > **Create Component** > **Create Now**. 2. Select the application to which the component belongs from the **Application** drop-down list. 3. Select a component creation method. ■ **Create Component**: Go to **Step 9.d**. ■ **Clone Component**: Go to **Step 9.e**. 4. Click **Create Component** and perform the following operations to create a component: a. Enter a task name containing 1 to 64 characters. b. (Optional) Select a task in the previous phase that can be associated with the current task from the **Associated Task** drop-down list. For example, select the **Create Application** task created in the previous phase. c. Set component parameters. For container-deployed components, see **Creating a Component Based on a Container Using Manual Configurations**. For VM-deployed components, see **Creating a Component Based on a VM**. If you specify **Associated Task**, for example, the **Create Application** task created in the previous phase, the **Application** of the component is fixed to the application set in the selected **Create Application** task. d. Click **Deploy Now**. Select a created component from the component list and click **Set** to reset component parameters. Select a created component from the component list and click **Remove** to remove the component from the component creation task. e. Go to **Step 9.f**. 5. Click **Clone Component** and perform the following operations to configure the component: a. Select the component to be cloned from the component list and click **OK**. b. Click the component to be cloned in the component list and click **Set** to reset component parameters.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
	For container-deployed components, see **Creating a Component Based on a Container Using Manual Configurations**. For VM-deployed components, see **Creating a Component Based on a VM**. c. Go to **Step 9.f**. 6. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
Update Component	Add a component upgrade task to the release task. 1. Use either of the following methods to go to the **Update Component** page: ■ Choose **All** > **Update Component** > **Create Now**. ■ Choose **Application Management** > **Update Component** > **Create Now**. 2. Select the application to which the component belongs from the **Application** drop-down list. 3. Set the component image source. ■ If you enable **Image from Build**, you need to create a build task in the phase before the current task phase. Go to **Step 9.d**. ■ If you disable **Image from Build**, go to **Step 9.e**. 4. If you enable **Image from Build**, perform the following operations: a. Click **Add Component** and select only one container-deployed component to add to the upgrade task. The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**. b. Click **OK**. You can click **Change** to reselect a component. c. Select a build task from the **Build** drop-down list as the component image source for component upgrade. By default, the archive address of the image package output during component upgrade is the archive address set in the selected build task displayed in the **Build Output** drop-down list. d. Go to **Step 9.f**. 5. If you disable **Image from Build**, perform the following operations: a. Click **Select Components to Upgrade** and select the component to be added to the upgrade task. The component status is **Running**, **Not ready**, or **Abnormal**. For details about how to check the component status, see **Viewing Component Details**. b. Click **OK**. You can click **Remove** on the card of a component in the list of selected components to be upgraded to remove the component from the list. Click **Set** on the card of the specified component in the list of selected components to be upgraded, enter a **Task Name** containing 1 to 64 characters, set

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
	component upgrade parameters by referring to **Step 5** to **Step 6** in **Upgrading Components in Batches**, and click **OK**. c. Go to **Step 9.f**. 6. Click **OK**.
Build	Add a job created for building images from source code, or a JAR, or WAR package to the release task. 1. Use either of the following methods to go to the **Build** page: ▪ Choose **All** > **Build** > **Create Now**. ▪ Choose **Application Management** > **Build** > **Create Now**. 2. Enter a task name containing 1 to 64 characters. 3. Select a build task from the **Build** drop-down list. 4. Click **OK**.
Manual Approval	Add a manual approval checkpoint task to the release task. If manual approval is set, the release task process can continue only after being approved by related approvers. 1. Use either of the following methods to go to the **Manual Approval** page: ▪ Choose **All** > **Manual Approval** > **Create Now**. ▪ Choose **Checkpoint Tasks** > **Manual Approval** > **Create Now**. 2. Enter a task name containing 1 to 64 characters. 3. Select the approver of the release task process from the **Approved By** drop-down list. If the approver list cannot be obtained, contact the administrator to create a custom policy that has the **iam:users.listUsers** permission of IAM, and bind the policy to the user group to which the current account belongs. 4. Set **Approval Mode**. ▪ **Approved by all**: The release task process can continue only after all selected approvers approve the task. Otherwise, the release fails. ▪ **Approved by any person**: The release task process can continue after being approved by any of the selected approvers. 5. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Task Type	Description
Manual Trigger	Add a manual triggering task to the release task. If manual triggering is set, the release task process can continue only after you manually confirm it. Use either of the following methods to create a manually triggered task: – Choose **All** > **Manual Trigger** > **Create Now**. – Choose **Checkpoint Tasks** > **Manual Trigger** > **Create Now**.

Step 10 If you select **Example** and **Simple Example**, perform the following operations:

1. Click next to the phase name, set **Phase Name**, and click **OK**.
2. Click **Create Build Task** and set build task parameters.
   a. Enter a task name containing 1 to 64 characters.
   b. Select a build task from the **Build** drop-down list.
3. Click **OK**.
4. Go to **Step 12**.

Step 11 If you select **Example** and **Common Example**, perform the following operations:

1. Click next to the **Build** phase, set **Phase Name**, and click **OK**.
2. Click **Create Build Task** and set build task parameters.
   a. Enter a task name containing 1 to 64 characters.
   b. Select a build task from the **Build** drop-down list.
   c. Click **OK**.
3. Click next to the **Upgrade Component** phase, set **Phase Name**, and click **OK**.
4. Click **Create Component Upgrade Task** and set component upgrade task parameters.
   a. Select the application to which the component belongs from the **Application** drop-down list.
   b. Click **Add Component**. Only one container-deployed component whose status is **Running**, **Not ready**, or **Abnormal** can be added to the upgrade task.
      For details about how to check the component status, see **Viewing Component Details**.
   c. Click **OK**.
   d. Select a build task from the **Build** drop-down list as the component image source for component upgrade.
      By default, the archive address of the image package output during component upgrade is the archive address set in the selected build task displayed in the **Build Output** drop-down list.
   e. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

5. Go to **Step 12**.

Step 12 Perform subsequent operations based on whether to release the release task.

- If yes, click **Save and Execute**, select the task to be executed, and click **Execute**.

  If a checkpoint task is set in the release task, approve the task based on the checkpoint task type by referring to the following table.

Checkpoint Task Type	Approval Procedure
Manual Approval	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual approval task to be approved. 3. Click **Approve**. The **Manual Approval** dialog box is displayed. 4. Enter **Comments** to determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.
Manual Trigger	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual triggering task to be approved. 3. Click **Approve**. The **Manual Trigger** dialog box is displayed. 4. Determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.

  - On the **Release Process View** page, you can view the execution status of the release process and wait until the release task is complete.
  - After the release task is saved, its **Release Status** is **Initialized**. For details about how to check the release task status, see **Viewing Region Release Task Details**.
  - After the release task is saved, you can manage the release task by referring to **Managing Region Release Tasks**.
- If no, click **Save**.
  - After the release task is saved, its **Release Status** is **Initialized**. For details about how to check the release task status, see **Viewing Region Release Task Details**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- After the release task is saved, you can manage the release task by referring to **Managing Region Release Tasks**.

----End

7.1.3.7 Editing a Region Release Task Created Using a Release Task Template

This section describes how to edit a region release task created by **Creating a Region Release Task Using a Release Task Template**.

Prerequisites

- The region release task created using a template is not in the **Running** state. For details about how to check the region release task status, see **Viewing Region Release Task Details**.
- You have compressed the prepared release task template files into a ZIP package and uploaded the package to the repository. For details, see **Release Task Template Description**.
  - Upload the template package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Select the corresponding template package from OBS. Upload the template package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**.
  - If you need to use JFrog (example) as the repository for storing the template package, you can use an HTTP/HTTPS custom file download address as the template package download address. Upload the template package to the custom file address in advance.

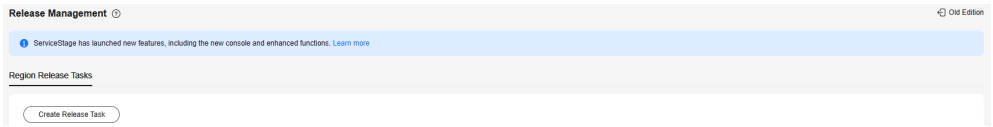
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

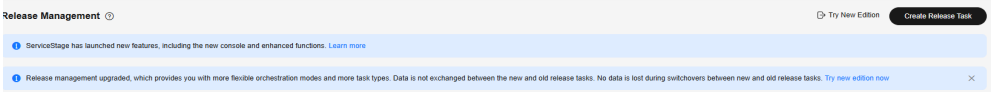
- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Use either of the following methods to go to the page for editing a release task:

- Directly edit: Click **Edit** in the **Operation** column of the target release task.
- Edit after viewing details: Click the target release task. After **Viewing Region Release Task Details**, click **Edit Release Task** in the upper right corner.

Step 6 Set the basic parameters of the release task by referring to the following table.

Parameter	Description
Release Task	The release task name must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**
Description	Description of a release task. 1. Click . 2. Enter up to 128 characters. 3. Click .

Step 7 Set **Set Template Package** by referring to the following table.

Operation	Procedure
Removing a template package	Perform the following operations based on the selected template package storage mode: • **CodeArts release repository**: Move the cursor to the added template package card and click . • **OBS**: Move the cursor to the added template package card and click . • **SWR repository**: Move the cursor to the added template package card and click . • **Custom file address**: Select the path of the template package file to be removed and click **Delete**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Procedure
Adding a template package	Select the uploaded release task template package based on the template package storage mode: • **CodeArts release repository**: Click **Select Software Package**, select the uploaded template package from the CodeArts release repository, and click **OK**. • **OBS**: Click **Select Software Package**, select the uploaded template package from the OBS bucket, and click **OK**. • **SWR repository**: Click **Select Software Package**, select the uploaded template package from the SWR repository, and click **OK**. • **Custom file address**: If you need to use JFrog (example) as the repository for storing the template package, perform the following operations to obtain the template package that has been uploaded to the custom file address: 1. Click **Add Custom File Address**. 2. Enter the HTTP/HTTPS custom file download address where the software package is located. The custom file address must be the complete path of the release task template package file, for example, **https://172.16.0.1:8082/artifactory/template-release.zip**. 3. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

Step 8 Click **Next**.

- In the **Preview Template Package** area, you can view the details of the release task template file.
- In the **Set Parameters** area, you can view and modify the release task parameters.

Step 9 Click **Upgrade**.

- On the **Release Process View** page, you can view the release task process details.
- After a release task is edited, you can manage the release task by referring to **Managing Region Release Tasks**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

7.1.3.8 Cloning a Region Release Task Created Using Manual Configurations

This section describes how to clone a region release task created by **Creating a Region Release Task Using Manual Configurations**.

Prerequisites

The region release task has been created using manual configurations. For details, see **Creating a Region Release Task Using Manual Configurations**.

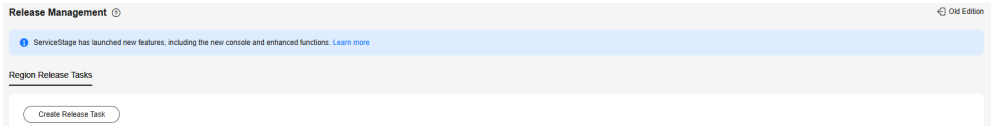
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Use either of the following methods to go to the page for cloning a release task:

- Directly clone: Click **More** > **Clone** in the **Operation** column of the target release task.
- Clone after viewing details: Click the target release task. After **Viewing Region Release Task Details**, choose **...** > **Clone** in the upper right corner.

Step 6 Edit the release task. For details, see **Step 6** to **Step 11** in **Editing a Region Release Task Created Using Manual Configurations**.

Step 7 Perform subsequent operations based on whether to release the release task.

- If yes, click **Save and Execute**, select the task to be executed, and click **Execute**.

  If a checkpoint task is set in the release task, approve the task based on the checkpoint task type by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Checkpoint Task Type	Approval Procedure
Manual Approval	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual approval task to be approved. 3. Click **Approve**. The **Manual Approval** dialog box is displayed. 4. Enter **Comments** to determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.
Manual Trigger	1. Click the release task to be executed. The **Release Process View** page is displayed. 2. Move the cursor to the card of the manual triggering task to be approved. 3. Click **Approve**. The **Manual Trigger** dialog box is displayed. 4. Determine whether to pass the approval. ▪ Click **Reject** to terminate the release task. The release task fails to be released. ▪ Click **Agree** to continue the release.

  - On the **Release Process View** page, you can view the execution status of the release process and wait until the release task is complete.
  - After the release task is saved, its **Release Status** is **Initialized**. For details about how to check the release task status, see **Viewing Region Release Task Details**.
  - After the release task is saved, you can manage the release task by referring to **Managing Region Release Tasks**.
- If no, click **Save**.
  - After the release task is saved, its **Release Status** is **Initialized**. For details about how to check the release task status, see **Viewing Region Release Task Details**.
  - After the release task is saved, you can manage the release task by referring to **Managing Region Release Tasks**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

7.1.3.9 Cloning a Region Release Task Created Using a Release Task Template

This section describes how to clone a region release task created by **Creating a Region Release Task Using a Release Task Template**.

Prerequisites

- The region release task has been created using a template. For details, see **Creating a Region Release Task Using a Release Task Template**.
- You have compressed the prepared release task template files into a ZIP package and uploaded the package to the repository. For details, see **Release Task Template Description**.
  - Upload the template package to the SWR repository. For details, see **Uploading a Software Package**.

    If PKIToken is disabled, the SWR repository is not supported.
  - Select the corresponding template package from OBS. Upload the template package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**.
  - If you need to use JFrog (example) as the repository for storing the template package, you can use an HTTP/HTTPS custom file download address as the template package download address. Upload the template package to the custom file address in advance.

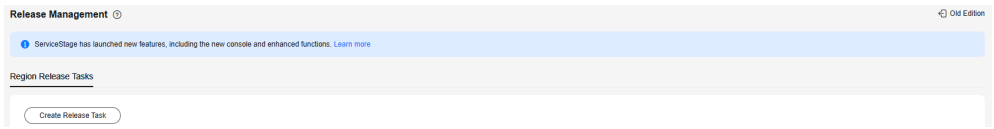
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 5 Use either of the following methods to go to the page for cloning a release task:

- Directly clone: Click **More** > **Clone** in the **Operation** column of the target release task.
- Clone after viewing details: Click the target release task. After **Viewing Region Release Task Details**, choose **...** > **Clone** in the upper right corner.

Step 6 Set the basic parameters of the release task by referring to the following table.

Parameter	Description
Release Task	The release task name must be unique in the same enterprise project. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after the function is enabled. For details, see **Enabling the Enterprise Project Function**. • Click the drop-down list to select an existing enterprise project. • Click **Create Enterprise Project** to create an enterprise project and select it. For details, see **Creating an Enterprise Project**
Description	Description of a release task. 1. Click . 2. Enter up to 128 characters. 3. Click .

Step 7 Set **Set Template Package** by referring to the following table.

Operation	Procedure
Removing a template package	Perform the following operations based on the selected template package storage mode: • **CodeArts release repository**: Move the cursor to the added template package card and click . • **OBS**: Move the cursor to the added template package card and click . • **SWR repository**: Move the cursor to the added template package card and click . • **Custom file address**: Select the path of the template package file to be removed and click **Delete**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Procedure
Adding a template package	Select the uploaded release task template package based on the template package storage mode: • **CodeArts release repository**: Click **Select Software Package**, select the uploaded template package from the CodeArts release repository, and click **OK**. • **OBS**: Click **Select Software Package**, select the uploaded template package from the OBS bucket, and click **OK**. • **SWR repository**: Click **Select Software Package**, select the uploaded template package from the SWR repository, and click **OK**. • **Custom file address**: If you need to use JFrog (example) as the repository for storing the template package, perform the following operations to obtain the template package that has been uploaded to the custom file address: 1. Click **Add Custom File Address**. 2. Enter the HTTP/HTTPS custom file download address where the software package is located. The custom file address must be the complete path of the release task template package file, for example, **https://172.16.0.1:8082/artifactory/template-release.zip**. 3. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located.

Step 8 Click **Next**.

- In the **Preview Template Package** area, you can view the details of the release task template file.
- In the **Set Parameters** area, you can view and modify the release task parameters.

Step 9 Click **Create and Deploy**.

- On the **Release Process View** page, you can view the release task process details.
- After the release task is created, its **Release Status** is **Initialized**. For details about how to check the release task status, see **Viewing Region Release Task Details**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- After a release task is created, you can manage the release task by referring to **Managing Region Release Tasks**.

----End

7.1.3.10 Configuring a Triggering Policy of a Region Release Task

Set a triggering policy for a region release task so that the release task can be started when the configured triggering policy is met.

Prerequisites

The region release task has been created. For details, see **Creating a Region Release Task**.

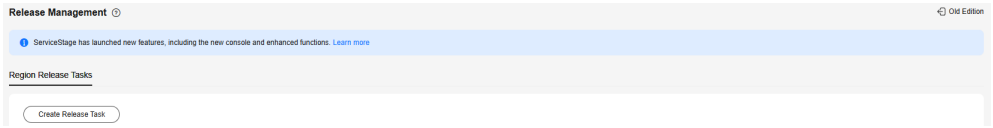
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Use either of the following methods to go to the page for configuring the triggering policy:

- Directly access: Click **More** > **Set Trigger Policy** in the **Operation** column of the target release task.
- Access after viewing details: Click the target release task. After **Viewing Region Release Task Details**, choose **...** > **Configure Trigger** in the upper right corner.

Step 6 Select the task for which you want to set a triggering policy and click **Next**.

Step 7 Set the triggering policy by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Trigger Type	Description
Manual	Manually start the release task. 1. Set **Type** to **Manual**. 2. Click **OK**. After the manual triggering policy is set, you can manually start the release task by referring to **Releasing a Region Release Task**.
Scheduled	Triggers a pipeline at a scheduled time. 1. Set **Type** to **Scheduled**. 2. Disable **Periodic Triggering**. 3. Set **Triggered**. 4. Click **OK**.
	Triggers the pipeline repeatedly. 1. Set **Type** to **Scheduled**. 2. Enable **Periodic Triggering**. 3. Set **Schedule**, **Triggered**, **Effective Time**, and **Days**. 4. Click **OK**.

Step 8 Click **OK**.

----End

7.1.3.11 Viewing the Running Records of a Region Release Task

View the running records of a region release task.

Prerequisites

The region release task has been released. For details, see **Releasing a Region Release Task**.

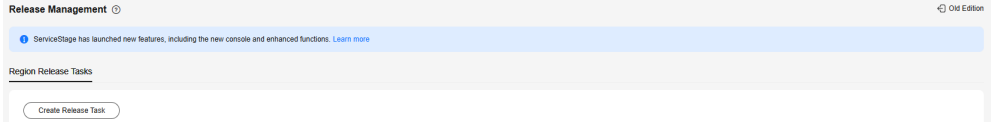
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Click **More** > **View Execution Records** in the **Operation** column of the target release task.

Step 6 Click **View Details** in the **Operation** column of the record to view release details.

----End

7.1.3.12 Viewing Region Release Task Details

View the details about a region release task.

Prerequisites

The region release task has been created. For details, see **Creating a Region Release Task**.

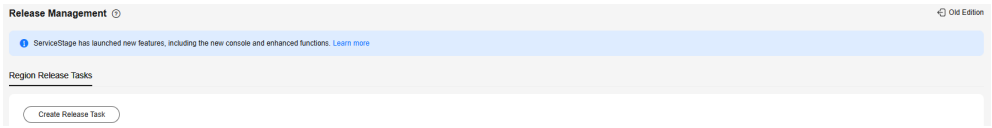
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

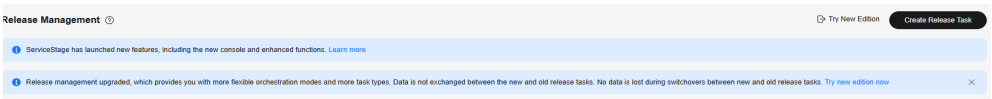
- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 5 Select a release task and view the status in the following table.

Name.	Description
Release Task	Name of a release task.
Release Status	Release task status. • **Initialized**: The release task is not released. • **Running**: The release task is being released. • **Successful**: The release task is released successfully. • **Release failed**: The release task fails to be released. • **Stopping**: The release task is being stopped. • **Suspend**: The release task has been suspended. • **Restarting**: The release task that has been suspended or fails to be released is being restarted. • **Rolling back**: The release task is being rolled back. • **Rolled back**: The release task is successfully rolled back to the status before execution. • **Rollback failed**: The release task fails to be rolled back to the status before execution. • **Skipped**: A task that is not selected in the release task is being skipped.
Enterprise Project	Enterprise project to which the release task belongs.
Released By	Account of the user who releases the release task.
Created	Creation time of a release task.
Executed	Time when the release task is last executed.

Step 6 Click the release task to view the release task process and execution history.

- Click **Release Process View** to view each phase of the release task and the tasks in each phase.
- Click **Execution History**, and click **View Details** in the **Operation** column of the record to view release details.

----End

7.1.3.13 Deleting a Region Release Task

Delete a region release task that is no longer used. Deleted release tasks cannot be restored. Exercise caution when performing this operation.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

The region release task is not in the **Running** state. For details about how to check the region release task status, see **Viewing Region Release Task Details**.

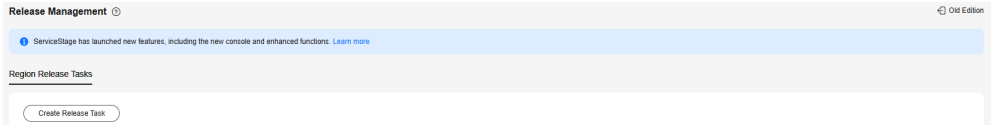
Procedure

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

- If the **Release Management** page of the new version is displayed by default, go to **Step 4**.

  Overview describes release management of the new version.

- If the **Release Management** page of the old version is displayed by default, go to **Step 3**.

  Overview describes release management of the old version.

Step 3 Click **Try new edition now** or **Try New Edition** to switch to the **Release Management** page of the new version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Choose **Region Release Tasks**.

Step 5 Use either of the following methods to delete a release task:

- Directly delete: Click **More** > **Delete** in the **Operation** column of the target release task.
- Delete after viewing details: Click the target release task. After **Viewing Region Release Task Details**, choose **...** > **Delete** in the upper right corner.

Step 6 Click **OK**.

----End

7.2 Old Release Management

7.2.1 Overview

Components of different technology stacks may need to be deployed in different deployment environments of different applications. If you create these components by referring to **Creating and Deploying a Component**, each component must be configured as required. This process is complex and faults may occur. In addition, the components need to be deployed and run immediately

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

after being created, and do not support on-demand deployment. If the component configurations are incorrect, the deployment fails. In this case, you need to delete the components, create them again, and then deploy them.

To upgrade component versions in batches, ServiceStage allows you to **upgrade components in batches**. However, only components in the same application can be upgraded in batches. In addition, all components are deployed in the same batch, and you cannot specify the deployment sequence. If the components to be upgraded depend on each other, for example, the startup of a component depends on other components, services may be affected.

To deploy multiple components with almost the same configurations, ServiceStage allows you to **clone components in batches**. However, only components in the same application can be cloned in batches.

To solve the preceding problems, you can use the release management function of ServiceStage to create different release tasks. A release task provides functions for releasing applications: single-component release, batch operations (release, upgrade, and clone), and dependency-based orchestration. This feature enables you to flexibly deploy components in batches, improving service efficiency and user experience.

- Create a batch deployment release task to deploy components of different technology stacks in different deployment environments of different applications.
- Create a batch upgrade release task to upgrade components of different applications in batches and specify the component upgrade sequence to ensure that services are not affected during the upgrade.
- Create a batch clone release task to clone components of different applications and quickly configure and deploy components in batches.

7.2.2 Creating a Release Task

7.2.2.1 Creating a Batch Deployment Release Task

Create a batch deployment release task to create multiple components in batches for deployment and release.

Creating a Batch Deployment Release Task

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

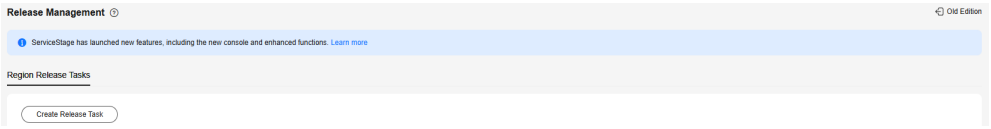
- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.



Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Click **Create Release Task**.

Step 5 Configure the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Select **Bulk Deploy** for **Operation Type**.

Step 7 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

Step 8 Click **Create Component**, set the parameters of the component to be added to the release task by referring to **Creating a Component Based on a Container Using Manual Configurations** or **Creating a Component Based on a VM**, and click **Create**.

A batch deployment release task can contain up to 64 components.

Step 9 After the target components are created, reset the version configurations of each component by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Component	Component name. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). For container-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments. For VM-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different environments.
Component Version	Version number of a component. • By default, the version number is the time when you start **Step 8**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. You can perform the following operations to synchronize component versions in batches: 1. Move the cursor to the **Component Version** text box of the specified component and click . 2. Select other components whose versions need to be synchronized. 3. Click **OK**.
Application	Select the application to which the component belongs. You can perform the following operations to synchronize the applications to which components belong in batches: 1. Move the cursor to the **Application** drop-down list of the specified component and click . 2. Select other components whose applications need to be synchronized. 3. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Environment	Select the component deployment environment. You can perform the following operations to synchronize the environments to which components belong in batches: 1. Move the cursor to the **Environment** drop-down list of the specified component and click . 2. Select other components whose environments need to be synchronized. 3. Click **OK**. After the environment is reset, the cluster to which the container-deployed components belong is changed to the cluster bound to the new environment, and the namespace to which the component belongs is changed to the default namespace of the cluster bound to the new environment.
Cluster	Select a CCE cluster for deploying and running the container-deployed component. You can perform the following operations to synchronize the clusters to which components belong in batches: 1. Move the cursor to the **Cluster** drop-down list of the specified component and click . 2. Select other components whose clusters need to be synchronized. 3. Click **OK**. After the cluster is set, the cluster is changed only when the selected cluster is bound to the environment to which the component belongs. In addition, the namespace to which the component belongs is changed to the default namespace of the new cluster.
Namespace	Select the namespace of the CCE cluster in the build environment for the container-deployed component to isolate build data. For details, see **Managing Namespaces**. You can perform the following operations to synchronize the namespaces to which components belong in batches: 1. Move the cursor to the **Namespace** drop-down list of the specified component and click . 2. Select other components whose namespaces need to be synchronized. 3. Click **OK**. After the namespace is set, the namespace is changed only when the selected namespace exists in the cluster to which the component belongs.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click , and select the component source again. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Step 8**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Instances	Set the number of running instances of the component. • For container-deployed components, the value ranges from 1 to 200. You can perform the following operations to synchronize the number of component instances in batches: 1. Move the cursor to the **Instances** text box of the specified component and click . 2. Select other components whose number of component instances need to be synchronized. 3. Click **OK**. • For VM-deployed components, you can perform the following operations to reset the number of component instances: 1. Click **Select**. 2. Reselect the number of servers for component deployment. The servers to be selected must have the VM agent installed and be running properly. If the status of the target server agent indicates that the agent is missing and to install it first, click **Install**. For details, see **Step 6** to **Step 8** in **Installing a VM Agent**. 3. Click **OK**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 10 Set the component build parameters.

This parameter is available when the component is deployed based on a container and the technology stack type is Java, Tomcat, Node.js, Python, or PHP.

1. Locate the target component and click **Build** in the **Operation** column.
2. Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. – **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. – **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. ▪ If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. ▪ To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The build environment must be a Kubernetes environment, and can access the Internet. You are advised to select **Use current environment**. If the CCE cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. – **Use independent environment**: Use an independent build environment to build an image. The CCE clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. – **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the CCE cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	– If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. – If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Select Cluster	Select a cluster for build if **Environment** is set to **HA Environment**. For details about the HA environment, see **Environment Overview**.
*Namespace	Select the namespace of the CCE cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Node Label	In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: – If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. – If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.

3. Click **OK**.

Step 11 Locate the target component and click **Advanced Settings** in the **Operation** column.

Step 12 Select a component technology stack and version. For details, see **Technology Stack**.

Step 13 Set **Headless Service Settings** for container-deployed stateful components.

It provides a fixed access domain name for every pod for mutual pod access in a StatefulSet.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 14 Set **Public Network Access**.

- For container-deployed components, go to **Step 15**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For VM-deployed components, go to **Step 16**.

Step 15 Click to enable public network access for container-deployed components.

Public network access can be configured if **Workload Type** of the component is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the component deployment environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If you need to upgrade and maintain components in ELB dark launch mode, enable public network access for the components. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

  a. Set **Public Network Load Balancer**.

  An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

  - Select a load balancer that has been bound to an EIP in the selected environment.
  - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
  - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	○ **Creating a Shared Load Balancer**. ○ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

  b. Set **Client Protocol**.

  - **HTTP** has security risks. You are advised to select **HTTPS**.
  - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

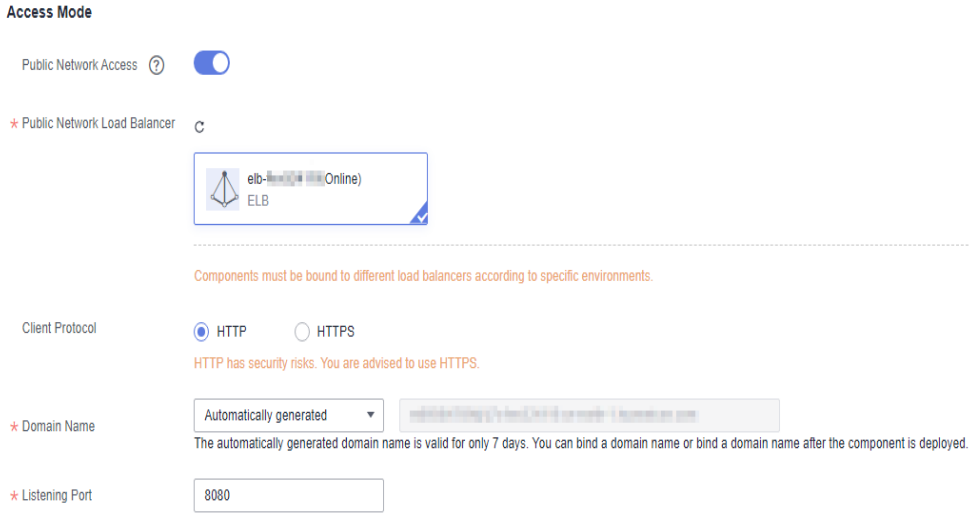
c. Set **Domain Name**.

- If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
- If **Bound** is selected, enter a domain name.

d. Set **Listening Port**.

Enter the listening port number of the application process.

Figure 7-2 Configuring public access



Step 16 Click to enable public network access for VM-deployed components.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources:
     - Create a shared load balancer by referring to **Creating a Shared Load Balancer**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Create a dedicated application load balancer by referring to **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.

   After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

Step 17 Set other component configurations.

- For container-deployed components, see **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components, see **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.

Step 18 Click **OK**.

Step 19 Perform subsequent operations based on whether to release the release task by referring to the following table.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Deployment Release Task**. – You can view the release records and release task information, and **Cloning a Batch Deployment Release Task** and **Deleting a Release Task**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Deployment Release Task**, **Editing a Batch Deployment Release Task**, and **Deleting a Release Task**.

----End

7.2.2.2 Creating a Batch Upgrade Release Task

Create a batch upgrade release task to upgrade the version configurations of multiple components at the same time.

Prerequisites

You have created and deployed the components to be added to the batch upgrade release task. For details, see **Creating and Deploying a Component**.

Creating a Batch Upgrade Release Task

Step 1 Log in to ServiceStage.

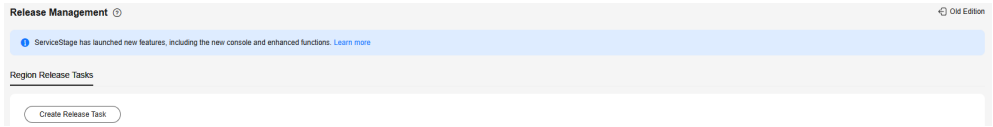
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Click **Create Release Task**.

Step 5 Configure the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Select **Bulk Upgrade** for **Operation Type**.

Step 7 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

Step 8 Click **Add Component**.

Step 9 Select the components to be added to the release task and click **OK**.

A batch upgrade release task supports up to 64 components in the **Running**, **Not ready**, **Unknown**, or **Abnormal** state. For details about how to check the component status, see **Viewing Component Details**.

Step 10 Reset the version configurations of each component by referring to the following table.

Parameter	Description
Component Version	Version number of a component. • By default, the version number is the time when you finish **Step 9**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click and select another package type. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • Set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you can retain the default settings or do not set this parameter.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Deployment Batch	Number of batches in which component instances are upgraded. The value range is [1, Total number of instances]. Total number of instances refers to the number of running instances of the component. For example, if there are 4 component instances and **Deployment Batch** is set to **2**, these component instances are upgraded in two batches, and each batch involves two component instances. For container-deployed components, you can reset this parameter when **Configuring a Component Upgrade Policy** in **Step 11**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 11 Click **Advanced Settings** in the **Operation** column of the target component, set advanced settings, and click **OK**.

- Select a component technology stack and version. For details, see **Technology Stack**.
- For container-deployed components, set other parameters by referring to **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components:
  - Set other parameters by referring to **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.
  - Select **Upgrade Policy** and set **Pause Before Stop**.

    The Rx traffic will be processed during this period of time. The unit is second and the value ranges from 0 to 600.

Step 12 Perform subsequent operations based on whether to release the release task by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 11**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Upgrade Release Task**. – You can view the release records and release task information, and **Rolling Back a Released Batch Upgrade Release Task**, **Cloning a Batch Upgrade Release Task**, and **Deleting a Release Task**.
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 11**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Upgrade Release Task**, **Editing a Batch Upgrade Release Task**, and **Deleting a Release Task**.

----End

7.2.2.3 Creating a Batch Clone Release Task

This avoids complex and repeated configurations because multiple microservice components may have same configurations, for example, scheduling policy, AS policy, data storage, log storage, and lifecycle configurations, and only minor differences need to be modified.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

ServiceStage allows you to create a batch clone release task, so that you can add similar components that already exist in the current service environment to this release task, improving component deployment efficiency and user experience.

Prerequisites

You have created and deployed the components to be added to the batch clone release task. For details, see **Creating and Deploying a Component**.

Creating a Batch Clone Release Task

Step 1 Log in to ServiceStage.

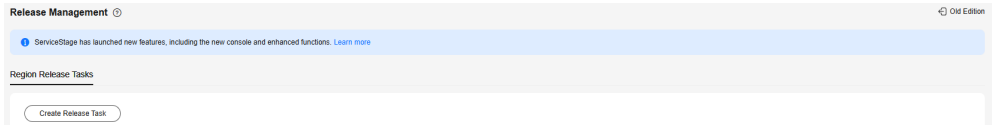
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Click **Create Release Task**.

Step 5 Configure the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Select **Bulk Clone** for **Operation Type**.

Step 7 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If it is disabled, the release task fails when a component fails to be released.

Step 8 Click **Add Component**.

Step 9 Select the components to be added to the release task and click **OK**.

A batch clone release task supports up to 64 components in the **Running**, **Not ready**, **Unknown**, or **Abnormal** state. For details about how to check the component status, see **Viewing Component Details**.

Step 10 Reset the version configurations of each component by referring to the following table.

Parameter	Description
Component	Component name. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). For container-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments. For VM-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different environments.
Component Version	Version number of a component. • By default, the version number is the time when you finish **Step 9**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. You can perform the following operations to synchronize component versions in batches: 1. Move the cursor to the **Component Version** text box of the specified component and click . 2. Select other components whose versions need to be synchronized. 3. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Application	Select the application to which the component belongs. You can perform the following operations to synchronize the applications to which components belong in batches: 1. Move the cursor to the **Application** drop-down list of the specified component and click . 2. Select other components whose applications need to be synchronized. 3. Click **OK**.
Environment	Select the component deployment environment. You can perform the following operations to synchronize the environments to which components belong in batches: 1. Move the cursor to the **Environment** drop-down list of the specified component and click . 2. Select other components whose environments need to be synchronized. 3. Click **OK**. After the environment is reset, the cluster to which the container-deployed components belong is changed to the cluster bound to the new environment, and the namespace to which the component belongs is changed to the default namespace of the cluster bound to the new environment.
Cluster	Select a CCE cluster for deploying and running the container-deployed component. You can perform the following operations to synchronize the clusters to which components belong in batches: 1. Move the cursor to the **Cluster** drop-down list of the specified component and click . 2. Select other components whose clusters need to be synchronized. 3. Click **OK**. After the cluster is set, the cluster is changed only when the selected cluster is bound to the environment to which the component belongs. In addition, the namespace to which the component belongs is changed to the default namespace of the new cluster.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Namespace	Select the namespace of the CCE cluster in the build environment for the container-deployed component to isolate build data. For details, see **Managing Namespaces**. You can perform the following operations to synchronize the namespaces to which components belong in batches: 1. Move the cursor to the **Namespace** drop-down list of the specified component and click . 2. Select other components whose namespaces need to be synchronized. 3. Click **OK**. After the namespace is set, the namespace is changed only when the selected namespace exists in the cluster to which the component belongs.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click , and select the component source again. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Instances	Set the number of running instances of the component. • For container-deployed components, the value ranges from 1 to 200. You can perform the following operations to synchronize the number of component instances in batches: 1. Move the cursor to the **Instances** text box of the specified component and click . 2. Select other components whose number of component instances need to be synchronized. 3. Click **OK**. • For VM-deployed components, you can perform the following operations to reset the number of component instances: 1. Click **Select**. 2. Reselect the number of servers for component deployment. The servers to be selected must have the VM agent installed and be running properly. If the status of the target server agent indicates that the agent is missing and to install it first, click **Install**. For details, see **Step 6** to **Step 8** in **Installing a VM Agent**. 3. Click **OK**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 11 Set the component build parameters.

This parameter is available when the component is deployed based on a container and the technology stack type is Java, Tomcat, Node.js, Python, or PHP.

1. Locate the target component and click **Build** in the **Operation** column.
2. Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. – **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. – **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. ▪ If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. ▪ To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The build environment must be a Kubernetes environment, and can access the Internet. You are advised to select **Use current environment**. If the CCE cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. – **Use independent environment**: Use an independent build environment to build an image. The CCE clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. – **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the CCE cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	– If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. – If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Select Cluster	Select a cluster for build if **Environment** is set to **HA Environment**. For details about the HA environment, see **Environment Overview**.
*Namespace	Select the namespace of the CCE cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Node Label	In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: – If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. – If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.

3. Click **OK**.

Step 12 Locate the target component and click **Advanced Settings** in the **Operation** column.

Step 13 Select a component technology stack and version. For details, see **Technology Stack**.

Step 14 Set **Headless Service Settings** for container-deployed stateful components.

It provides a fixed access domain name for every pod for mutual pod access in a StatefulSet.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 15 Set **Public Network Access**.

- For container-deployed components, go to **Step 16**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For VM-deployed components, go to **Step 17**.

Step 16 Click to enable public network access for container-deployed components.

Public network access can be configured if **Workload Type** of the component is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the component deployment environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If you need to upgrade and maintain components in ELB dark launch mode, enable public network access for the components. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

  a. Set **Public Network Load Balancer**.

  An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

  - Select a load balancer that has been bound to an EIP in the selected environment.
  - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
  - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	○ **Creating a Shared Load Balancer**. ○ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

  b. Set **Client Protocol**.

  - **HTTP** has security risks. You are advised to select **HTTPS**.
  - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

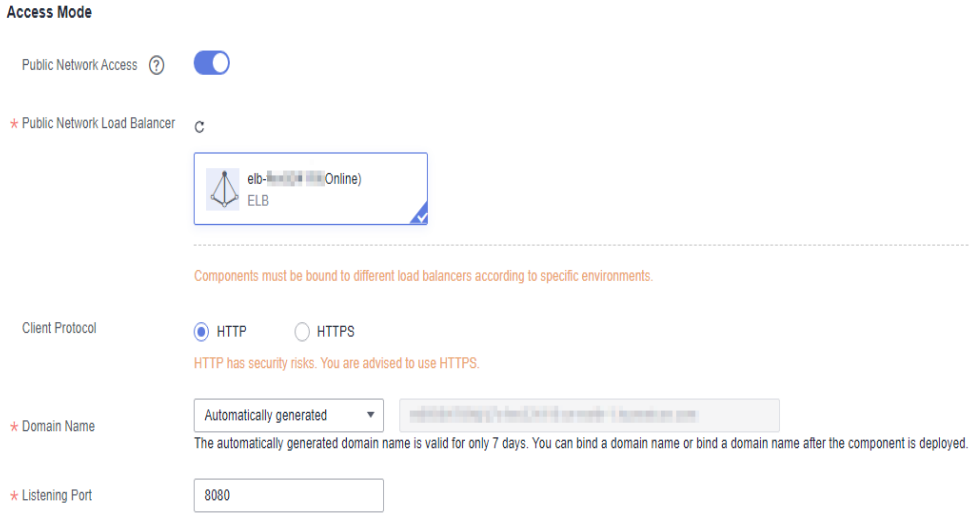
c. Set **Domain Name**.

- If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
- If **Bound** is selected, enter a domain name.

d. Set **Listening Port**.

Enter the listening port number of the application process.

Figure 7-3 Configuring public access



Step 17 Click to enable public network access for VM-deployed components.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources:
     - Create a shared load balancer by referring to **Creating a Shared Load Balancer**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Create a dedicated application load balancer by referring to **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.

   After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

Step 18 Set other component configurations.

- For container-deployed components, see **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components, see **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.

Step 19 Click **OK**.

Step 20 Perform subsequent operations based on whether to release the release task by referring to the following table.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 18**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Clone Release Task**. – You can view the release records and release task information, and **Cloning a Batch Clone Release Task** and **Deleting a Release Task**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 18**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Clone Release Task**, **Editing a Batch Clone Release Task**, and **Deleting a Release Task**.

----End

7.2.3 Releasing a Release Task

This topic describes how to release a release task in the **To release** state to deploy, upgrade, or clone components in batches.

Prerequisites

You have created a release task but not released it. For details, see **Creating a Release Task**.

Releasing a Release Task

Step 1 Log in to ServiceStage.

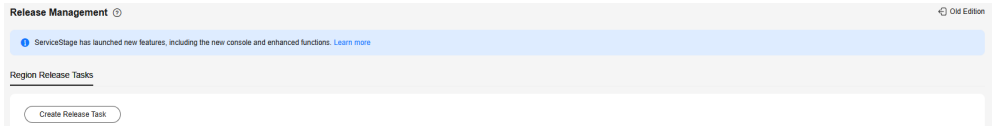
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task in the **To release** state.

- Release directly: Click **Release** in the **Operation** column.
- Release after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the configurations of each component, and click **Release**.

Step 5 Click **OK**.

- For batch deployment release tasks, you can view the release records and release task information, and **Cloning a Batch Deployment Release Task**, **Deleting a Release Task**, and **Suspending the Release of a Release Task**.
- For batch upgrade release tasks, you can view the release records and release task information, and **Rolling Back a Released Batch Upgrade Release Task**, **Cloning a Batch Upgrade Release Task**, **Deleting a Release Task**, and **Suspending the Release of a Release Task**.
- For batch clone release tasks, you can view the release records and release task information, and **Cloning a Batch Upgrade Release Task**, **Deleting a Release Task**, and **Suspending the Release of a Release Task**.

----End

7.2.4 Suspending the Release of a Release Task

During release task release, if the component configuration is incorrect, you can click **Abort** to stop the release.

Prerequisites

The release task is in the **Releasing** state. For details about how to check the release task status, see **Viewing a Release Task**.

Procedure

Step 1 Log in to ServiceStage.

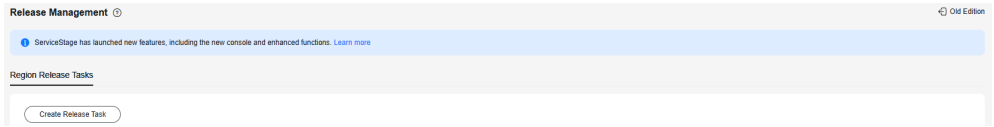
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task in the **Releasing** state.

- Directly suspend: Click **More** > **Abort** in the **Operation** column.
- Suspend after viewing details: Click the target release task to go to its **Release Management** page, view the details, and click **Abort**.

Step 5 Click **OK**.

Wait until release task status changes from **Suspending** to **Suspended**, which indicates that the release task is suspended.

----End

7.2.5 Editing a Release Task

7.2.5.1 Editing a Batch Deployment Release Task

Edit a batch deployment release task that has not been released based on service changes.

Prerequisites

You have created a batch deployment release task but not released it. For details, see **Creating a Batch Deployment Release Task**.

Editing a Batch Deployment Release Task

Step 1 Log in to ServiceStage.

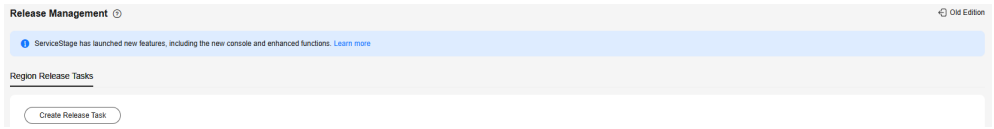
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 Select the target release task in the **To release** state.

- Edit directly: Click **More** > **Edit** in the **Operation** column.
- Edit after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the configurations of each component, and click **Edit**.

Step 5 Modify the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

Step 7 To delete a component from the release task, click **Delete** in the **Operation** column of the target component.

Step 8 To add a component, click **Create Component**, set the parameters of the component to be added to the release task by referring to **Creating a Component Based on a Container Using Manual Configurations** or **Creating a Component Based on a VM**, and click **Create**.

A batch deployment release task can contain up to 64 components.

Step 9 Reset the version configurations of each component by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Component	Component name. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). For container-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments. For VM-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different environments.
Component Version	Version number of a component. • By default, the version number of the original components in the release task is the original version number of the components, and the version number of the new components in the release task is the time when you perform **Step 8**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. You can perform the following operations to synchronize component versions in batches: 1. Move the cursor to the **Component Version** text box of the specified component and click . 2. Select other components whose versions need to be synchronized. 3. Click **OK**.
Application	Select the application to which the component belongs. You can perform the following operations to synchronize the applications to which components belong in batches: 1. Move the cursor to the **Application** drop-down list of the specified component and click . 2. Select other components whose applications need to be synchronized. 3. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Environment	Select the component deployment environment. You can perform the following operations to synchronize the environments to which components belong in batches: 1. Move the cursor to the **Environment** drop-down list of the specified component and click . 2. Select other components whose environments need to be synchronized. 3. Click **OK**. After the environment is reset, the cluster to which the container-deployed components belong is changed to the cluster bound to the new environment, and the namespace to which the component belongs is changed to the default namespace of the cluster bound to the new environment.
Cluster	Select a CCE cluster for deploying and running the container-deployed component. You can perform the following operations to synchronize the clusters to which components belong in batches: 1. Move the cursor to the **Cluster** drop-down list of the specified component and click . 2. Select other components whose clusters need to be synchronized. 3. Click **OK**. After the cluster is set, the cluster is changed only when the selected cluster is bound to the environment to which the component belongs. In addition, the namespace to which the component belongs is changed to the default namespace of the new cluster.
Namespace	Select the namespace of the CCE cluster in the build environment for the container-deployed component to isolate build data. For details, see **Managing Namespaces**. You can perform the following operations to synchronize the namespaces to which components belong in batches: 1. Move the cursor to the **Namespace** drop-down list of the specified component and click . 2. Select other components whose namespaces need to be synchronized. 3. Click **OK**. After the namespace is set, the namespace is changed only when the selected namespace exists in the cluster to which the component belongs.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click , and select the component source again. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Instances	Set the number of running instances of the component. • For container-deployed components, the value ranges from 1 to 200. You can perform the following operations to synchronize the number of component instances in batches: 1. Move the cursor to the **Instances** text box of the specified component and click . 2. Select other components whose number of component instances need to be synchronized. 3. Click **OK**. • For VM-deployed components, you can perform the following operations to reset the number of component instances: 1. Click **Select**. 2. Reselect the number of servers for component deployment. The servers to be selected must have the VM agent installed and be running properly. If the status of the target server agent indicates that the agent is missing and to install it first, click **Install**. For details, see **Step 6** to **Step 8** in **Installing a VM Agent**. 3. Click **OK**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 10 Reset the component build parameters.

This parameter is available when the component is deployed based on a container and the technology stack type is Java, Tomcat, Node.js, Python, or PHP.

1. Locate the target component and click **Build** in the **Operation** column.
2. Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. – **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. – **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. ▪ If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. ▪ To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The build environment must be a Kubernetes environment, and can access the Internet. You are advised to select **Use current environment**. If the CCE cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. – **Use independent environment**: Use an independent build environment to build an image. The CCE clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. – **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the CCE cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	– If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. – If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Select Cluster	Select a cluster for build if **Environment** is set to **HA Environment**. For details about the HA environment, see **Environment Overview**.
*Namespace	Select the namespace of the CCE cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Node Label	In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: – If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. – If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.

3. Click **OK**.

Step 11 Locate the target component and click **Advanced Settings** in the **Operation** column.

Step 12 Select a component technology stack and version. For details, see **Technology Stack**.

Step 13 Set **Headless Service Settings** for container-deployed stateful components.

It provides a fixed access domain name for every pod for mutual pod access in a StatefulSet.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 14 Set **Public Network Access**.

- For container-deployed components, go to **Step 15**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For VM-deployed components, go to **Step 16**.

Step 15 Click to enable public network access for container-deployed components.

Public network access can be configured if **Workload Type** of the component is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the component deployment environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If you need to upgrade and maintain components in ELB dark launch mode, enable public network access for the components. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

  a. Set **Public Network Load Balancer**.

  An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

  - Select a load balancer that has been bound to an EIP in the selected environment.
  - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
  - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	○ **Creating a Shared Load Balancer**. ○ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

  b. Set **Client Protocol**.

  - **HTTP** has security risks. You are advised to select **HTTPS**.
  - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

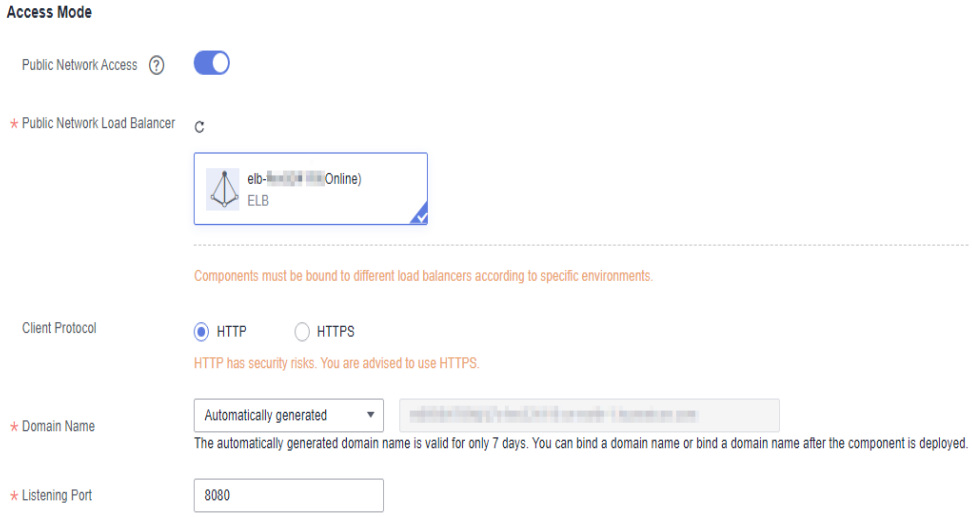
c. Set **Domain Name**.

- If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
- If **Bound** is selected, enter a domain name.

d. Set **Listening Port**.

Enter the listening port number of the application process.

Figure 7-4 Configuring public access



Step 16 Click to enable public network access for VM-deployed components.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources:
     - Create a shared load balancer by referring to **Creating a Shared Load Balancer**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Create a dedicated application load balancer by referring to **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.

   After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

Step 17 Set other component configurations.

- For container-deployed components, see **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components, see **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.

Step 18 Click **OK**.

Step 19 Perform subsequent operations based on whether to release the release task by referring to the following table.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Deployment Release Task**. – You can view the release records and release task information, and **Cloning a Batch Deployment Release Task** and **Deleting a Release Task**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Deployment Release Task**, **Editing a Batch Deployment Release Task**, and **Deleting a Release Task**.

----End

7.2.5.2 Editing a Batch Upgrade Release Task

Edit a batch upgrade release task that has not been released based on service changes.

Prerequisites

You have created a batch upgrade release task but not released it. For details, see **Creating a Batch Upgrade Release Task**.

Editing a Batch Upgrade Release Task

Step 1 Log in to ServiceStage.

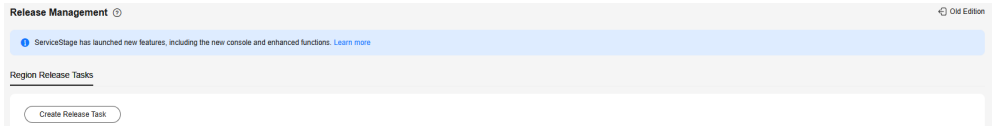
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task in the **To release** state.

- Edit directly: Click **More** > **Edit** in the **Operation** column.
- Edit after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the configurations of each component, and click **Edit**.

Step 5 Modify the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

Step 7 To delete a component from the release task, click **Delete** in the **Operation** column of the target component.

Step 8 To add components, click **Add Component**, select the components to be added to the release task, and click **OK**.

A batch upgrade release task supports up to 64 components in the **Running**, **Not ready**, **Unknown**, or **Abnormal** state. For details about how to check the component status, see **Viewing Component Details**.

Step 9 Reset the version configurations of each component by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Component Version	Version number of a component. • By default, the version number of original components in the release task is the time when you start **Step 4**, and the version number of new components in the release task is the time when you finish **Step 8**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click , and select the component source again. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Deployment Batch	Number of batches in which component instances are upgraded. The value range is [1, Total number of instances]. Total number of instances refers to the number of running instances of the component. For example, if there are 4 component instances and **Deployment Batch** is set to **2**, these component instances are upgraded in two batches, and each batch involves two component instances. For container-deployed components, you can reset this parameter when **Configuring a Component Upgrade Policy** in **Step 10**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 10 Click **Advanced Settings** in the **Operation** column of the target component, set advanced settings, and click **OK**.

- Select a component technology stack and version. For details, see **Technology Stack**.
- For container-deployed components, set other parameters by referring to **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components:
  - Set other parameters by referring to **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.
  - Select **Upgrade Policy** and set **Pause Before Stop**.

    The Rx traffic will be processed during this period of time. The unit is second and the value ranges from 0 to 600.

Step 11 Perform subsequent operations based on whether to release the release task by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 10**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Upgrade Release Task**. – You can view the release records and release task information, and **Rolling Back a Released Batch Upgrade Release Task**, **Cloning a Batch Upgrade Release Task**, and **Deleting a Release Task**.
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 10**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Upgrade Release Task**, **Editing a Batch Upgrade Release Task**, and **Deleting a Release Task**.

----End

7.2.5.3 Editing a Batch Clone Release Task

Edit a batch clone release task that has not been released based on service changes.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

You have created a batch clone release task but not released it. For details, see **Creating a Batch Clone Release Task**.

Editing a Batch Clone Release Task

Step 1 Log in to ServiceStage.

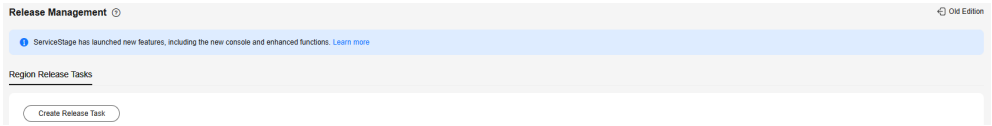
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task in the **To release** state.

- Edit directly: Click **More** > **Edit** in the **Operation** column.
- Edit after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the configurations of each component, and click **Edit**.

Step 5 Modify the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 7 To delete a component from the release task, click **Delete** in the **Operation** column of the target component.

Step 8 To add components, click **Add Component**, select the components to be added to the release task, and click **OK**.

A batch clone release task supports up to 64 components in the **Running**, **Not ready**, **Unknown**, or **Abnormal** state. For details about how to check the component status, see **Viewing Component Details**.

Step 9 Reset the version configurations of each component by referring to the following table.

Parameter	Description
Component	Component name. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). For container-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments. For VM-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different environments.
Component Version	Version number of a component. • By default, the version number of the original components in the release task is the original version number of the components, and the version number of the new components in the release task is the time when you finish **Step 8**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. You can perform the following operations to synchronize component versions in batches: 1. Move the cursor to the **Component Version** text box of the specified component and click . 2. Select other components whose versions need to be synchronized. 3. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Application	Select the application to which the component belongs. You can perform the following operations to synchronize the applications to which components belong in batches: 1. Move the cursor to the **Application** drop-down list of the specified component and click . 2. Select other components whose applications need to be synchronized. 3. Click **OK**.
Environment	Select the component deployment environment. You can perform the following operations to synchronize the environments to which components belong in batches: 1. Move the cursor to the **Environment** drop-down list of the specified component and click . 2. Select other components whose environments need to be synchronized. 3. Click **OK**. After the environment is reset, the cluster to which the container-deployed components belong is changed to the cluster bound to the new environment, and the namespace to which the component belongs is changed to the default namespace of the cluster bound to the new environment.
Cluster	Select a CCE cluster for deploying and running the container-deployed component. You can perform the following operations to synchronize the clusters to which components belong in batches: 1. Move the cursor to the **Cluster** drop-down list of the specified component and click . 2. Select other components whose clusters need to be synchronized. 3. Click **OK**. After the cluster is set, the cluster is changed only when the selected cluster is bound to the environment to which the component belongs. In addition, the namespace to which the component belongs is changed to the default namespace of the new cluster.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Namespace	Select the namespace of the CCE cluster in the build environment for the container-deployed component to isolate build data. For details, see **Managing Namespaces**. You can perform the following operations to synchronize the namespaces to which components belong in batches: 1. Move the cursor to the **Namespace** drop-down list of the specified component and click . 2. Select other components whose namespaces need to be synchronized. 3. Click **OK**. After the namespace is set, the namespace is changed only when the selected namespace exists in the cluster to which the component belongs.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click , and select the component source again. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Instances	Set the number of running instances of the component. • For container-deployed components, the value ranges from 1 to 200. You can perform the following operations to synchronize the number of component instances in batches: 1. Move the cursor to the **Instances** text box of the specified component and click . 2. Select other components whose number of component instances need to be synchronized. 3. Click **OK**. • For VM-deployed components, you can perform the following operations to reset the number of component instances: 1. Click **Select**. 2. Reselect the number of servers for component deployment. The servers to be selected must have the VM agent installed and be running properly. If the status of the target server agent indicates that the agent is missing and to install it first, click **Install**. For details, see **Step 6** to **Step 8** in **Installing a VM Agent**. 3. Click **OK**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 10 Reset the component build parameters.

This parameter is available when the component is deployed based on a container and the technology stack type is Java, Tomcat, Node.js, Python, or PHP.

1. Locate the target component and click **Build** in the **Operation** column.
2. Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. – **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. – **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. ▪ If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. ▪ To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The build environment must be a Kubernetes environment, and can access the Internet. You are advised to select **Use current environment**. If the CCE cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. – **Use independent environment**: Use an independent build environment to build an image. The CCE clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. – **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the CCE cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	– If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. – If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Select Cluster	Select a cluster for build if **Environment** is set to **HA Environment**. For details about the HA environment, see **Environment Overview**.
*Namespace	Select the namespace of the CCE cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Node Label	In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: – If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. – If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.

3. Click **OK**.

Step 11 Locate the target component and click **Advanced Settings** in the **Operation** column.

Step 12 Select a component technology stack and version. For details, see **Technology Stack**.

Step 13 Set **Headless Service Settings** for container-deployed stateful components.

It provides a fixed access domain name for every pod for mutual pod access in a StatefulSet.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 14 Set **Public Network Access**.

- For container-deployed components, go to **Step 15**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For VM-deployed components, go to **Step 16**.

Step 15 Click to enable public network access for container-deployed components.

Public network access can be configured if **Workload Type** of the component is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the component deployment environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If you need to upgrade and maintain components in ELB dark launch mode, enable public network access for the components. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

  a. Set **Public Network Load Balancer**.

  An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

  - Select a load balancer that has been bound to an EIP in the selected environment.
  - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
  - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	○ **Creating a Shared Load Balancer**. ○ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

  b. Set **Client Protocol**.

  - **HTTP** has security risks. You are advised to select **HTTPS**.
  - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

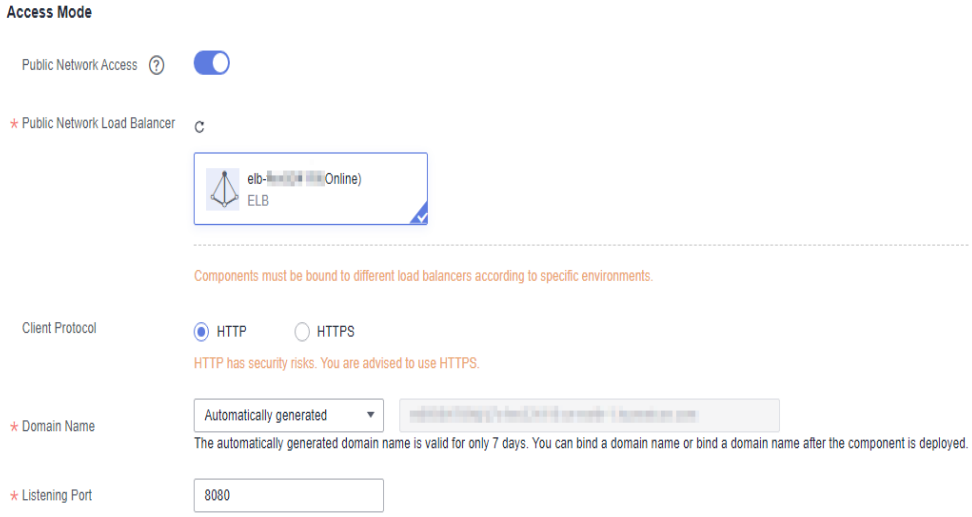
c. Set **Domain Name**.

- If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
- If **Bound** is selected, enter a domain name.

d. Set **Listening Port**.

Enter the listening port number of the application process.

Figure 7-5 Configuring public access



Step 16 Click to enable public network access for VM-deployed components.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources:
     - Create a shared load balancer by referring to **Creating a Shared Load Balancer**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Create a dedicated application load balancer by referring to **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.

   After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

Step 17 Set other component configurations.

- For container-deployed components, see **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components, see **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.

Step 18 Click **OK**.

Step 19 Perform subsequent operations based on whether to release the release task by referring to the following table.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Clone Release Task**. – You can view the release records and release task information, and **Cloning a Batch Clone Release Task** and **Deleting a Release Task**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Clone Release Task**, **Editing a Batch Clone Release Task**, and **Deleting a Release Task**.

----End

7.2.6 Cloning a Release Task

7.2.6.1 Cloning a Batch Deployment Release Task

If you need to deploy components in batches and a batch deployment release task already exists, you can clone this release task and modify it to create another batch deployment release task to improve service efficiency.

Prerequisites

You have created the batch deployment release task to be cloned. For details, see **Creating a Batch Deployment Release Task**.

Cloning a Batch Deployment Release Task

Step 1 Log in to ServiceStage.

Step 2 Choose **Release Management**.

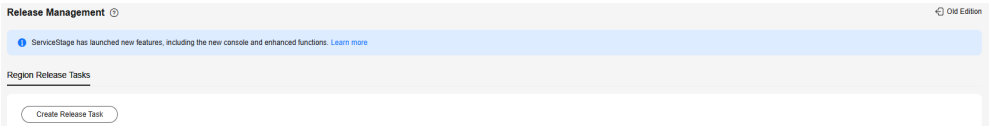
- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.



Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task.

- Clone directly: Click **Clone** in the **Operation** column.
- Clone after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the configurations of each component, and click **Clone**.

Step 5 Modify the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

Step 7 To delete a component from the release task, click **Delete** in the **Operation** column of the target component.

Step 8 To add components, click **Create Component** to create components to be added to the release task in batches.

A batch deployment release task can contain up to 64 components. For details about how to create a component, see **Creating a Component Based on a Container Using Manual Configurations** or **Creating a Component Based on a VM**.

Step 9 Reset the version configurations of each component by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Component	Component name. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). For container-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments. For VM-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different environments.
Component Version	Version number of a component. • By default, the version number of the original components in the release task is the original version number of the components, and the version number of the new components in the release task is the time when you finish **Step 8**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. You can perform the following operations to synchronize component versions in batches: 1. Move the cursor to the **Component Version** text box of the specified component and click . 2. Select other components whose versions need to be synchronized. 3. Click **OK**.
Application	Select the application to which the component belongs. You can perform the following operations to synchronize the applications to which components belong in batches: 1. Move the cursor to the **Application** drop-down list of the specified component and click . 2. Select other components whose applications need to be synchronized. 3. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Environment	Select the component deployment environment. You can perform the following operations to synchronize the environments to which components belong in batches: 1. Move the cursor to the **Environment** drop-down list of the specified component and click . 2. Select other components whose environments need to be synchronized. 3. Click **OK**. After the environment is reset, the cluster to which the container-deployed components belong is changed to the cluster bound to the new environment, and the namespace to which the component belongs is changed to the default namespace of the cluster bound to the new environment.
Cluster	Select a CCE cluster for deploying and running the container-deployed component. You can perform the following operations to synchronize the clusters to which components belong in batches: 1. Move the cursor to the **Cluster** drop-down list of the specified component and click . 2. Select other components whose clusters need to be synchronized. 3. Click **OK**. After the cluster is set, the cluster is changed only when the selected cluster is bound to the environment to which the component belongs. In addition, the namespace to which the component belongs is changed to the default namespace of the new cluster.
Namespace	Select the namespace of the CCE cluster in the build environment for the container-deployed component to isolate build data. For details, see **Managing Namespaces**. You can perform the following operations to synchronize the namespaces to which components belong in batches: 1. Move the cursor to the **Namespace** drop-down list of the specified component and click . 2. Select other components whose namespaces need to be synchronized. 3. Click **OK**. After the namespace is set, the namespace is changed only when the selected namespace exists in the cluster to which the component belongs.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click , and select the component source again. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Instances	Set the number of running instances of the component. • For container-deployed components, the value ranges from 1 to 200. You can perform the following operations to synchronize the number of component instances in batches: 1. Move the cursor to the **Instances** text box of the specified component and click . 2. Select other components whose number of component instances need to be synchronized. 3. Click **OK**. • For VM-deployed components, you can perform the following operations to reset the number of component instances: 1. Click **Select**. 2. Reselect the number of servers for component deployment. The servers to be selected must have the VM agent installed and be running properly. If the status of the target server agent indicates that the agent is missing and to install it first, click **Install**. For details, see **Step 6** to **Step 8** in **Installing a VM Agent**. 3. Click **OK**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 10 Reset the component build parameters.

This parameter is available when the component is deployed based on a container and the technology stack type is Java, Tomcat, Node.js, Python, or PHP.

1. Locate the target component and click **Build** in the **Operation** column.
2. Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. – **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. – **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. ▪ If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. ▪ To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The build environment must be a Kubernetes environment, and can access the Internet. You are advised to select **Use current environment**. If the CCE cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. – **Use independent environment**: Use an independent build environment to build an image. The CCE clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. – **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the CCE cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	– If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. – If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Select Cluster	Select a cluster for build if **Environment** is set to **HA Environment**. For details about the HA environment, see **Environment Overview**.
*Namespace	Select the namespace of the CCE cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Node Label	In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: – If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. – If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.

3. Click **OK**.

Step 11 Locate the target component and click **Advanced Settings** in the **Operation** column.

Step 12 Select a component technology stack and version. For details, see **Technology Stack**.

Step 13 Set **Headless Service Settings** for container-deployed stateful components.

It provides a fixed access domain name for every pod for mutual pod access in a StatefulSet.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 14 Set **Public Network Access**.

- For container-deployed components, go to **Step 15**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For VM-deployed components, go to **Step 16**.

Step 15 Click to enable public network access for container-deployed components.

Public network access can be configured if **Workload Type** of the component is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the component deployment environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If you need to upgrade and maintain components in ELB dark launch mode, enable public network access for the components. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

  a. Set **Public Network Load Balancer**.

  An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

  - Select a load balancer that has been bound to an EIP in the selected environment.
  - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
  - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	○ **Creating a Shared Load Balancer**. ○ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

  b. Set **Client Protocol**.

  - **HTTP** has security risks. You are advised to select **HTTPS**.
  - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

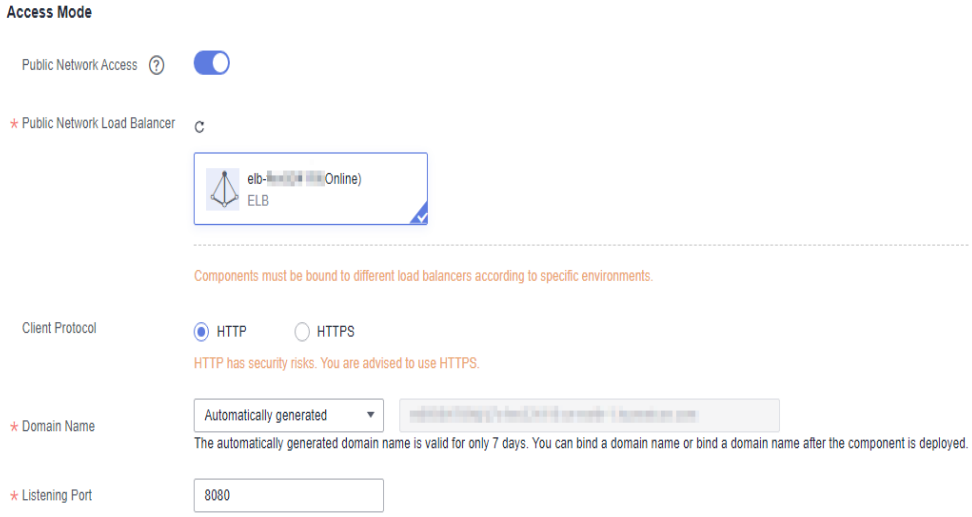
c. Set **Domain Name**.

- If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
- If **Bound** is selected, enter a domain name.

d. Set **Listening Port**.

Enter the listening port number of the application process.

Figure 7-6 Configuring public access



Step 16 Click to enable public network access for VM-deployed components.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources:
     - Create a shared load balancer by referring to **Creating a Shared Load Balancer**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Create a dedicated application load balancer by referring to **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.

   After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

Step 17 Set other component configurations.

- For container-deployed components, see **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components, see **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.

Step 18 Click **OK**.

Step 19 Perform subsequent operations based on whether to release the release task by referring to the following table.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Deployment Release Task**. – You can view the release records and release task information, and **Cloning a Batch Deployment Release Task** and **Deleting a Release Task**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Deployment Release Task**, **Editing a Batch Deployment Release Task**, and **Deleting a Release Task**.

----End

7.2.6.2 Cloning a Batch Upgrade Release Task

If you need to upgrade components in batches and a batch upgrade release task already exists, you can clone this release task and modify it to create another batch upgrade release task to improve service efficiency.

Prerequisites

You have created the batch upgrade release task to be cloned. For details, see **Creating a Batch Upgrade Release Task**.

Cloning a Batch Upgrade Release Task

Step 1 Log in to ServiceStage.

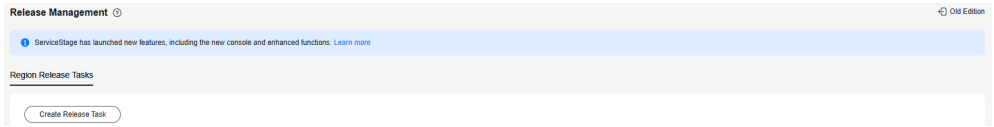
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task.

- Clone directly: Click **Clone** in the **Operation** column.
- Clone after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the configurations of each component, and click **Clone**.

Step 5 Modify the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

Step 7 To delete a component from the release task, click **Delete** in the **Operation** column of the target component.

Step 8 To add components, click **Add Component**, select the components to be added to the release task, and click **OK**.

A batch upgrade release task supports up to 64 components in the **Running**, **Not ready**, **Unknown**, or **Abnormal** state. For details about how to check the component status, see **Viewing Component Details**.

Step 9 Reset the version configurations of each component by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Component Version	Version number of a component. • By default, the version number of original components in the release task is the time when you finish **Step 4**, and the version number of new components in the release task is the time when you finish **Step 8**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. The customized version number must be unique and cannot be the same as any historical version number of the component. Otherwise, the current deployment record will overwrite the historical deployment record of the same version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click , and select the component source again. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Deployment Batch	Number of batches in which component instances are upgraded. The value range is [1, Total number of instances]. Total number of instances refers to the number of running instances of the component. For example, if there are 4 component instances and **Deployment Batch** is set to **2**, these component instances are upgraded in two batches, and each batch involves two component instances. For container-deployed components, you can reset this parameter when **Configuring a Component Upgrade Policy** in **Step 10**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 10 Click **Advanced Settings** in the **Operation** column of the target component, set advanced settings, and click **OK**.

- Select a component technology stack and version. For details, see **Technology Stack**.
- For container-deployed components, set other parameters by referring to **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components:
  - Set other parameters by referring to **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.
  - Select **Upgrade Policy** and set **Pause Before Stop**.

    The Rx traffic will be processed during this period of time. The unit is second and the value ranges from 0 to 600.

Step 11 Perform subsequent operations based on whether to release the release task by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 10**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Upgrade Release Task**. – You can view the release records and release task information, and **Rolling Back a Released Batch Upgrade Release Task**, **Cloning a Batch Upgrade Release Task**, and **Deleting a Release Task**.
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 10**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Upgrade Release Task**, **Editing a Batch Upgrade Release Task**, and **Deleting a Release Task**.

----End

7.2.6.3 Cloning a Batch Clone Release Task

If you need to clone components in batches and a batch clone release task already exists, you can clone this release task and modify it to create another batch clone release task to improve service efficiency.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

You have created the batch clone release task to be cloned. For details, see **Creating a Batch Clone Release Task**.

Cloning a Batch Clone Release Task

Step 1 Log in to ServiceStage.

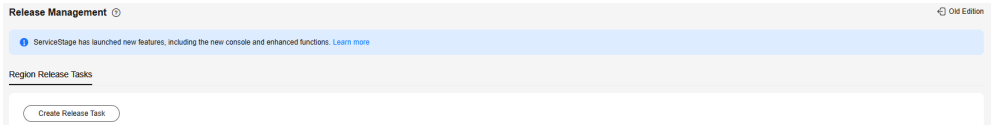
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task.

- Clone directly: Click **Clone** in the **Operation** column.
- Clone after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the configurations of each component, and click **Clone**.

Step 5 Modify the release task by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Release Task	Name of a release task. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
Description	Description of a release task. Enter up to 128 characters.

Step 6 Set **Best Effort Policy**.

- Enable it to continue releasing other components when a component fails to be released.
- If it is disabled, the release task fails when a component fails to be released.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 7 To delete a component from the release task, click **Delete** in the **Operation** column of the target component.

Step 8 To add components, click **Add Component**, select the components to be added to the release task, and click **OK**.

A batch clone release task supports up to 64 components in the **Running**, **Not ready**, **Unknown**, or **Abnormal** state. For details about how to check the component status, see **Viewing Component Details**.

Step 9 Reset the version configurations of each component by referring to the following table.

Parameter	Description
Component	Component name. Enter 2 to 64 characters. Start with a letter and end with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-). For container-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different CCE clusters in the same HA environment. • Components with the same name in the same application can be deployed in different environments. For VM-deployed components: • Components with the same name in different applications can be deployed in the same environment. • Components with the same name in the same application can be deployed in different environments.
Component Version	Version number of a component. • By default, the version number of the original components in the release task is the original version number of the components, and the version number of the new components in the release task is the time when you finish **Step 8**. The format is yyyy.mmdd.hhmms, where **s** is the ones place of the second in the timestamp. For example, if the timestamp is 2022.0803.104321, the version number is 2022.0803.10431. • You can also customize the version number in the format of A.B.C, or A.B.C.D. A. B, C, and D are natural numbers, for example, 1.0.0 or 1.0.0.0. You can perform the following operations to synchronize component versions in batches: 1. Move the cursor to the **Component Version** text box of the specified component and click . 2. Select other components whose versions need to be synchronized. 3. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Application	Select the application to which the component belongs. You can perform the following operations to synchronize the applications to which components belong in batches: 1. Move the cursor to the **Application** drop-down list of the specified component and click . 2. Select other components whose applications need to be synchronized. 3. Click **OK**.
Environment	Select the component deployment environment. You can perform the following operations to synchronize the environments to which components belong in batches: 1. Move the cursor to the **Environment** drop-down list of the specified component and click . 2. Select other components whose environments need to be synchronized. 3. Click **OK**. After the environment is reset, the cluster to which the component belongs is changed to the cluster bound to the new environment, and the namespace to which the component belongs is changed to the default namespace of the cluster bound to the new environment.
Cluster	Select a CCE cluster for deploying and running the container-deployed component. You can perform the following operations to synchronize the clusters to which components belong in batches: 1. Move the cursor to the **Cluster** drop-down list of the specified component and click . 2. Select other components whose clusters need to be synchronized. 3. Click **OK**. After the cluster is set, the cluster is changed only when the selected cluster is bound to the environment to which the component belongs. In addition, the namespace to which the component belongs is changed to the default namespace of the new cluster.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Namespace	Select the namespace of the CCE cluster in the build environment for the container-deployed component to isolate build data. For details, see **Managing Namespaces**. You can perform the following operations to synchronize the namespaces to which components belong in batches: 1. Move the cursor to the **Namespace** drop-down list of the specified component and click . 2. Select other components whose namespaces need to be synchronized. 3. Click **OK**. After the namespace is set, the namespace is changed only when the selected namespace exists in the cluster to which the component belongs.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Software Package/ Image Package/ Source Code Repository	Click , and select the component source again. For details, see **Component Source**. • If you select an image package for a container-deployed component: – If **Multi-container** is selected in **Creating and Deploying a Component**, select a container and click to specify it as the initialization container for installing tools or scripts before the service container starts. At least one container must be specified as the service container (non-initialization container). For details about container initialization, see **Init Containers**. – You can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. – Click in the upper right corner of the selected container to change the container image package source. • If you select a software package for a container-deployed component, you can customize the container name. The name contains 1 to 63 characters, including lowercase letters, digits, and hyphens (-), and must start with a lowercase letter and end with a lowercase letter or digit. • If the package type is software package and you need to use JFrog (example) as the software package repository, you can select **Custom file address**. 1. Enter the HTTP/HTTPS custom file address where the software package is located. 2. Determine whether to enable authentication. If authentication is disabled, any user can download the software package in the custom file address by default. Click to enable authentication. Only authenticated users can download the software package in the custom file address. **Authentication mode** can be **User name and password authentication** or **User-defined Header Authentication**. The authentication mode and the corresponding authentication parameters are determined by the authentication mode supported by the server where the custom file directory is located. • For container-deployed components, you can set **Image Access Credential**. You can select up to 16 image access credentials. Click **Create Secret** to create an image access credential. **Secret Type** must be kubernetes.io/dockerconfigjson. For details, see **Creating a Secret**. An image access credential is a secret, which is used to store the authentication information required for pulling images from a private repository. For a public repository image, you

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
	can retain the default settings or do not set this parameter. For a private repository image, this parameter is mandatory. Otherwise, the image fails to be pulled.
Instances	Set the number of running instances of the component. • For container-deployed components, the value ranges from 1 to 200. You can perform the following operations to synchronize the number of component instances in batches: 1. Move the cursor to the **Instances** text box of the specified component and click . 2. Select other components whose number of component instances need to be synchronized. 3. Click **OK**. • For VM-deployed components, you can perform the following operations to reset the number of component instances: 1. Click **Select**. 2. Reselect the number of servers for component deployment. The servers to be selected must have the VM agent installed and be running properly. If the status of the target server agent indicates that the agent is missing and to install it first, click **Install**. For details, see **Step 6** to **Step 8** in **Installing a VM Agent**. 3. Click **OK**.
Deployment Sequence	Set the deployment sequence of all components to be added to the release task. • If the selected components depend on each other, for example, the startup of a component depends on other components, set **Deployment Sequence** so that the depended components are deployed first. For example, for components A, B, and C to be added to the release task, the startup of component A depends on components B and C. Therefore, set **Deployment Sequence** of components B and C to **1**, and set **Deployment Sequence** of component A to **2**. • If the components do not depend on each other, retain the default value of **Deployment Sequence**. All components will be deployed in the same batch.

Step 10 Reset the component build parameters.

This parameter is available when the component is deployed based on a container and the technology stack type is Java, Tomcat, Node.js, Python, or PHP.

1. Locate the target component and click **Build** in the **Operation** column.
2. Set the parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Command	If the package type is **Source code repository**, set **Command** based on service requirements. – **Default command or script**: preferentially executes **build.sh** in the **root** directory. If **build.sh** does not exist, the code will be compiled using the common method of the selected language. Example: **mvn clean package** for Java. – **Custom command**: Commands are customized using the selected language. Alternatively, the default command or script is used after **build.sh** is modified. ▪ If **Custom command** is selected, exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. ▪ To run the compilation command in the project subdirectory, you need to go to the project subdirectory and then run other script commands. Example: **cd ./weather**/ **mvn clean package**
*Dockerfile Address	If the package type is **Source code repository**, set **Dockerfile Address** based on service requirements. **Dockerfile Address** is the directory where the Dockerfile is located relative to the root directory (./) of the project. The Dockerfile is used to build an image. If **Dockerfile Address** is not specified, the system searches for the Dockerfile in the root directory of the project by default. If the Dockerfile does not exist in the root directory, the system automatically generates the Dockerfile based on the selected operating environment.
*Organization	An organization is used to manage images generated during component build.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Build	Select the type of the environment used to build an image. The build environment must be a Kubernetes environment, and can access the Internet. You are advised to select **Use current environment**. If the CCE cluster in the current environment cannot access the Internet and you have planned an independent build environment, you can select **Use independent environment**. – **Use independent environment**: Use an independent build environment to build an image. The CCE clusters in the independent build environment and in the current component deployment environment must have the same CPU architecture. Otherwise, the component deployment fails. – **Use current environment**: Use the deployment environment to which the component belongs to build an image. In the current environment, masters and nodes in the CCE cluster must have the same CPU architecture. Otherwise, the component build fails.
*Environment	– If **Use independent environment** is selected for **Build**, select an independent build environment different from that to which the component belongs. – If **Use current environment** is selected for **Build**, select the deployment environment to which the component belongs.
*Select Cluster	Select a cluster for build if **Environment** is set to **HA Environment**. For details about the HA environment, see **Environment Overview**.
*Namespace	Select the namespace of the CCE cluster in the environment where the build is executed, which is used to isolate build data. For details, see **Managing Namespaces**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Node Label	In the following cases, set a node label to deliver the build job to the fixed node bound with an EIP to ensure that the component can be successfully built and deployed: – If **Use independent environment** is selected for **Build**, set a node label to deliver the build job to the node in the independent environment, to ensure that the node CPU architecture is the same as that of the node in the current component deployment environment. – If **Use current environment** is selected for **Build** and the CPU architecture of a node is different from that of the master node in the environment, set a node label to deliver the build job to the node that has the same CPU architecture as the master node. For details about how to add a label, see **Managing Node Labels**.

3. Click **OK**.

Step 11 Locate the target component and click **Advanced Settings** in the **Operation** column.

Step 12 Select a component technology stack and version. For details, see **Technology Stack**.

Step 13 Set **Headless Service Settings** for container-deployed stateful components.

It provides a fixed access domain name for every pod for mutual pod access in a StatefulSet.

Parameter	Description
*Service Name	Enter the name of the service that can be accessed by external services. A service is an abstract method that exposes a group of applications running on a pod as network services. Service names of components deployed in the same cluster must be unique. Otherwise, component deployment will fail.
*Port Settings	Set the following port parameters: • **Port Name**: name of the port for communication between pods. • **Container Port**: port used by an application for listening in a container. • **Access Port**: port for accessing the service. To add port settings, click **Add Setting**.

Step 14 Set **Public Network Access**.

- For container-deployed components, go to **Step 15**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For VM-deployed components, go to **Step 16**.

Step 15 Click to enable public network access for container-deployed components.

Public network access can be configured if **Workload Type** of the component is **Stateless**, **Stateful**, or **DaemonSet** and a CCE cluster is bound to the component deployment environment.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

- If you need to upgrade and maintain components in ELB dark launch mode, enable public network access for the components. For details about ELB dark launch, see **Upgrading a Component in Dark Launch Mode**.
- After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.
- By default, public network access is disabled for a component. After a component is created and deployed, you can also configure the component access mode. For details, see **Configuring the Component Access Mode**.

  a. Set **Public Network Load Balancer**.

  An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

  - Select a load balancer that has been bound to an EIP in the selected environment.
  - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
  - To create ELB resources, refer to the following table.

Scenario	Reference
Use the domain name to access an application.	○ **Creating a Shared Load Balancer**. ○ See **Creating a Dedicated Load Balancer**.
ELB dark launch	See **Creating a Dedicated Load Balancer**.

  b. Set **Client Protocol**.

  - **HTTP** has security risks. You are advised to select **HTTPS**.
  - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.

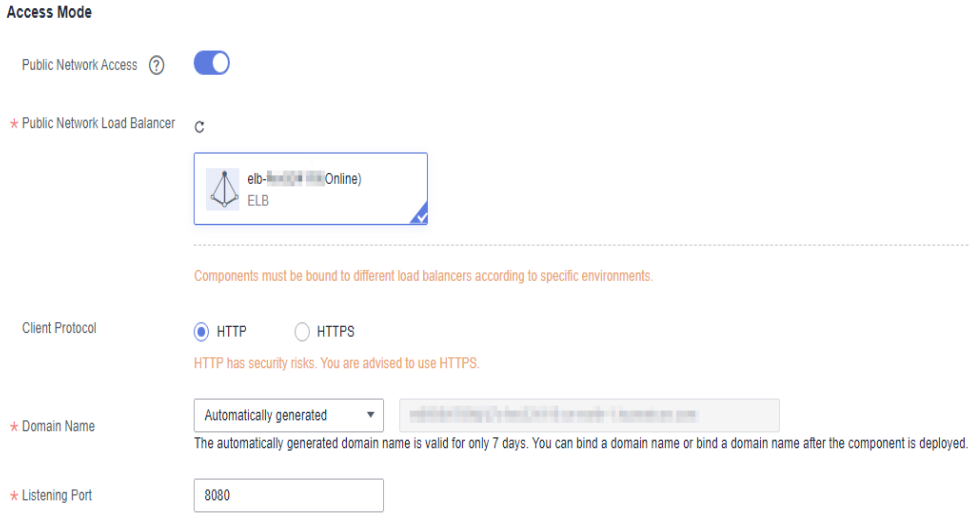
c. Set **Domain Name**.

- If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
- If **Bound** is selected, enter a domain name.

d. Set **Listening Port**.

Enter the listening port number of the application process.

Figure 7-7 Configuring public access



Step 16 Click to enable public network access for VM-deployed components.

After public network access is enabled for a component, you can use a public network domain name to access the component through an ELB bound with an EIP to use services provided by the component.

After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

1. Set **Public Network Load Balancer**.

   An EIP has been bound to the load balancer and must be in the same VPC and subnet as the compute resources managed in the current component deployment environment. Components must be bound with different load balancers in different deployment environments to avoid route errors.

   - Select a load balancer that has been bound to an EIP in the selected environment.
   - If no load balancer exists, click **Add One**. On the **Edit Environment** page that is displayed, click **Add Optional Resource** to add created load balancers to the environment.
   - To create ELB resources:
     - Create a shared load balancer by referring to **Creating a Shared Load Balancer**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Create a dedicated application load balancer by referring to **Creating a Dedicated Load Balancer**.

2. Set **Client Protocol**.
   - **HTTP** has security risks. You are advised to select **HTTPS**.
   - If **HTTPS** is selected, click **Use existing** to select an existing certificate.

     If no certificate exists, click **Create new** to create a server certificate. For details, see **Adding a Certificate**.
3. Set **Domain Name**.
   - If **Automatically generated** is selected, the automatically generated domain name is valid only for seven days.
   - If **Bound** is selected, enter a domain name.

   After a component with public network access enabled is created and deployed, you can change the configured component access domain name by referring to **Changing the Component Access Domain Name**.

Step 17 Set other component configurations.

- For container-deployed components, see **Managing Cloud Service Settings of a Component** to **Managing Advanced Settings of a Container-Deployed Component**.
- For VM-deployed components, see **Managing Cloud Service Settings of a Component** and **Managing VM-Deployed Component Configurations**.

Step 18 Click **OK**.

Step 19 Perform subsequent operations based on whether to release the release task by referring to the following table.

Release or Not	Operation
Yes	1. Click **Complete and Execute**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, the components in the release task will be deployed in batches as configured. – During the release task release, if the component configuration is incorrect, you can click **Suspend** to stop the release. Then, click **Clone**, reconfigure the component, and release the release task again by referring to **Cloning a Batch Clone Release Task**. – You can view the release records and release task information, and **Cloning a Batch Clone Release Task** and **Deleting a Release Task**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Release or Not	Operation
No	1. Click **Finish**. The system automatically checks whether the advanced settings of each component are correct. 2. If an error is reported, perform the following operations and then continue: a. In the displayed dialog box, confirm the information and click **OK**. b. Click **Advanced Settings** in the **Operation** column of the abnormal component and set the parameter by referring to **Step 17**. 3. If the advanced settings pass the pre-check, a release task in the **To release** state will be generated. On the **Release Management** page, you can view release task information, and **Releasing a Release Task**, **Cloning a Batch Clone Release Task**, **Editing a Batch Clone Release Task**, and **Deleting a Release Task**.

----End

7.2.7 Rolling Back a Released Batch Upgrade Release Task

This section describes how to roll back a released batch upgrade release task to the status before the release.

Prerequisites

- You have created a batch upgrade release task and released it. For details, see **Creating a Batch Upgrade Release Task** and **Releasing a Release Task**.
- The batch upgrade release task has not been rolled back.

Rolling Back a Released Batch Upgrade Release Task

Step 1 Log in to ServiceStage.

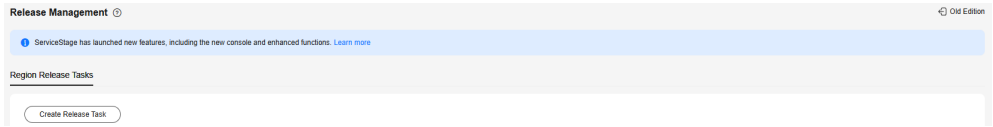
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task.

- Roll back directly: Click **Roll Back** in the **Operation** column.
- Roll back after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the details, and click **Roll Back**.

Step 5 Click **OK**.

You can view the release records and release task information, and **Cloning a Batch Upgrade Release Task** and **Deleting a Release Task**.

----End

7.2.8 Viewing a Release Task

This section describes how to view a created release task

Prerequisites

You have created a release task. For details, see **Creating a Release Task**.

Viewing a Release Task

Step 1 Log in to ServiceStage.

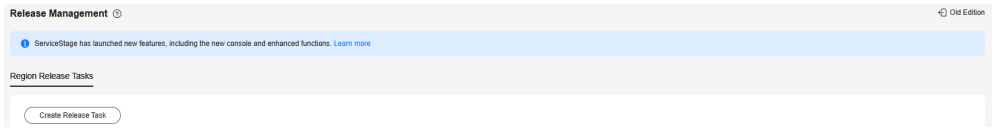
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Click the target release task. The **Release Management** page is displayed.

- View the release task information by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Item	Description
Release Status	View the release task status next to the release task name. – **To release**: The release task has been created but has not been released. – **Released**: The release task has been released. – **Releasing**: The release task is being released. – **Release failed**: The release task fails to be released. – **Rolling back**: The released batch upgrade release task is being rolled back. – **Rolled back**: The released batch upgrade release task has been rolled back. – **Rollback failed**: The released batch upgrade release task fails to be rolled back. – **Suspending**: The release task being released is being suspended. – **Suspended**: The release task has been stopped.
Release Task ID	Click to copy the release task ID.
Created	Creation time of a release task.
Released By	Creator of a release task.
Released	Release time of a release task whose **Release Status** is **Released**.
Operation Type	Operation type of a release task. – Bulk Deploy – Bulk Upgrade – Bulk Clone
Best Effort Policy	Status of **Best Effort Policy** of a release task. – **On**: If a component fails to be released, continue to release other components. – **Off**: If a component fails to be released, the release task fails.
Description	Description of a release task.
Release Info	View the configurations of each component in a release task on the **Release Info** page.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Item	Description
Release History	View the release overview and release details of each component in a release task in the following states on the **Release History** page. – Released – Releasing – Release failed – Rolling back – Rolled back – Rollback failed – Suspending – Suspended

- Click the **Release Info** tab to view the detailed configurations of the components in the release task.
- If the release task status is not **To release**, click the **Release History** tab to view the release overview and component release details of the release task.

----End

7.2.9 Deleting a Release Task

This section describes how to delete a release task that is no longer used.

NOTICE

Deleted release tasks cannot be restored. Exercise caution when performing this operation.

Prerequisites

You can delete a release task in the following states:

- To release
- Released
- Rollback failed
- Release failed
- Rolled back
- Suspended

For details about how to check the release task status, see **Viewing a Release Task**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Deleting a Release Task

Step 1 Log in to ServiceStage.

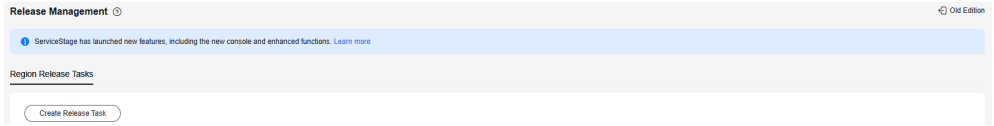
Step 2 Choose **Release Management**.

- If the **Release Management** page of the old version is displayed by default, go to **Step 4**.

  Overview describes release management of the old version.

- If the **Release Management** page of the new version is displayed by default, go to **Step 3**.

  Overview describes release management of the new version.

Step 3 Click **Old Edition** to switch to the **Release Management** page of the old version.

The data of the old and new versions are isolated. Switching them will not cause release task data loss.

Step 4 Select the target release task.

- Delete directly: Click **More** > **Delete** in the **Operation** column.
- Delete after confirming the release task details: Click the target release task to go to its **Release Management** page, confirm the details, and click **Delete**.

Step 5 Click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

8 System Settings

8.1 Tech Stack Management

8.1.1 Adding a Custom Technology Stack Version

A technology stack includes the operating system, framework, and runtime system for component running. It consists of attributes such as the stack name, type, status, and version. For details, see **Technology Stack**.

Based on the built-in technology stacks, ServiceStage provides technology stack management to support technology stack version customization and multiple runtime systems.

You can add the technology stack type and deployment type of a custom technology stack version to ServiceStage. For details, see **Table 8-1**. For details about deployment types, that is, the component deployment modes supported by the component technology stacks, see **Component Deployment Modes**.

Table 8-1 Custom technology stack versions

Type	Deployment Type
Java	• VM: A VM, or an ECS or custom server, is a basic computing unit that consists of vCPUs, memory, OS, and EVS disks. After creating a VM, you can use it like using your local computer or physical server to deploy components. • Kubernetes: container-based or CCE-based deployment. CCE is a scalable, enterprise-class hosted Kubernetes service. With CCE, you can easily deploy, manage, and scale containerized applications in the cloud.
Tomcat	

Prerequisites

The technology stack version software package (in .zip or .tar.gz format) to be added has been uploaded to OBS. For details, see **Streaming Upload (PUT)**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Adding a Custom Technology Stack Version

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Tech Stack Management**. The **Tech Stack Management** page is displayed.

Step 3 Select the target technology stack and click **Add Tech Stack Version**.

Step 4 Set the parameters. Parameters marked with an asterisk (*) are mandatory.

- Java

Parameter	Description
*Tech Stack	Technology stack version name. Enter 2 to 64 characters. Only use letters, digits, slashes (/), underscores (_), and hyphens (-). Start with a letter and end with a letter or digit.
*Tech Stack Version	Technology stack version. The format is A.B.C, where A, B, and C are natural numbers. Example: **1.0.0**.
*OS Image	This parameter is mandatory for technology stacks whose **Deployment Type** is **Kubernetes**. Kubernetes is a container orchestration platform that manages containers (usually Docker containers). Each container runs in an isolated environment provided by an OS image. An OS image contains the OS kernel, libraries, tools, and applications required for running a container, ensuring that the container can run consistently in any Docker-enabled environment. – **Built-in**: Select a built-in OS image from the drop-down list. – **Custom**: Click **Select Image Package** and select the uploaded custom OS image package. **My Images** (private images), **Open Source Images**, **Shared Images**, and **Third-Party Images** are supported. ▪ If you select **My Images**, upload the image to the image repository in advance. For details, see **Uploading an Image**. ▪ If you select **Third-Party Images**, ensure that you have obtained the address of the third-party image. The format of the image address is as follows: *{IP address of the third-party image repository}:{Port number for accessing the third-party image repository}/{Image storage path}/{Image name}:{Image tag}* or *{Image name}:{Image tag}* If the image tag is not specified, the latest version is used by default.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*JDK Version Path	Path for storing the JDK software package that has been uploaded to OBS. 1. Click **Select Software Package**. 2. Select the path. 3. Click **OK**.
Description	Technology stack description. Enter up to 512 characters.

- Tomcat

Parameter	Description
*Tech Stack	Technology stack version name. Enter 2 to 64 characters. Only use letters, digits, slashes (/), underscores (_), and hyphens (-). Start with a letter and end with a letter or digit.
*Tech Stack Version	Technology stack version. The format is A.B.C, where A, B, and C are natural numbers. Example: **1.0.0**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*OS Image	This parameter is mandatory for technology stacks whose **Deployment Type** is **Kubernetes**. Kubernetes is a container orchestration platform that manages containers (usually Docker containers). Each container runs in an isolated environment provided by an OS image. An OS image contains the OS kernel, libraries, tools, and applications required for running a container, ensuring that the container can run consistently in any Docker-enabled environment. – **Built-in**: Select a built-in OS image from the drop-down list. – **Custom**: Click **Select Image Package** and select the uploaded custom OS image package. **My Images** (private images), **Open Source Images**, **Shared Images**, and **Third-Party Images** are supported. ▪ If you select **My Images**, upload the image to the image repository in advance. For details, see **Uploading an Image**. ▪ If you select **Third-Party Images**, ensure that you have obtained the address of the third-party image. The format of the image address is as follows: *{IP address of the third-party image repository}:{Port number for accessing the third-party image repository}/{Image storage path}/{Image name}:{Image tag}* or *{Image name}:{Image tag}* If the image tag is not specified, the latest version is used by default.
*JDK Version Path	Path for storing the JDK software package that has been uploaded to OBS. 1. Click **Select Software Package**. 2. Select the path. 3. Click **OK**.
*Tomcat Version Path	Path for storing the Tomcat software package that has been uploaded to OBS. 1. Click **Select Software Package**. 2. Select the path. 3. Click **OK**.
Description	Technology stack description. Enter up to 512 characters.

Step 5 Click **OK**.

Click ⌄ before the technology stack for which a custom technology stack version has been added to expand the technology stack list. You can see that the custom

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

technology stack version exists with **Type** being **Custom** and **Status** being **Disabled**.

----End

Follow-up Operations

- **Enabling a Technology Stack Version**: A technology stack version can be used for **Creating and Deploying a Component** only after being enabled.
- **Editing a Custom Technology Stack Version**
- **Disabling a Technology Stack Version**
- **Deleting a Custom Technology Stack Version**

8.1.2 Enabling and Disabling a Technology Stack Version

Enabling a Technology Stack Version

The technology stack versions built in ServiceStage and the custom technology stack versions added in **Adding a Custom Technology Stack Version** can be used for **Creating and Deploying a Component** only after being enabled.

Technology stack versions in the **Disabled** state support this operation.

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Tech Stack Management**. The **Tech Stack Management** page is displayed.

Step 3 Click ⌄ before the target technology stack version to expand the list.

Step 4 Click **Enable** in the **Operation** column of the target technology stack version.

After a technology stack version is enabled, its status changes to **Enabling**. If the technology stack status is **Disabled** before the technology stack version is enabled, the technology stack status changes to **Enabling** after the technology stack version is enabled.

----End

Disabling a Technology Stack Version

If the technology stack versions built in ServiceStage and the custom technology stack versions added in **Adding a Custom Technology Stack Version** are no longer needed, you can disable them.

Disabled technology stack versions cannot be used for **Creating and Deploying a Component**.

Technology stack versions in the **Enabling** state support this operation.

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Tech Stack Management**. The **Tech Stack Management** page is displayed.

Step 3 Click ⌄ before the target technology stack version to expand the list.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 4 Click **Disable** in the **Operation** column of the target technology stack version.

After a technology stack version is disabled, its status changes to **Disabled**. If all versions of a technology stack are disabled, the technology stack status changes to **Disabled**.

----End

8.1.3 Enabling and Disabling a Technology Stack

This section describes how to disable a technology stack.

After a technology stack is disabled, all its technology stack versions are disabled and cannot be used for **Creating and Deploying a Component**.

Prerequisites

The technology stack is in the **Enabling** state.

Enabling a Technology Stack

After a technology stack is enabled, all its technology stack versions are enabled.

Technology stacks in the **Disabled** state support this operation.

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Tech Stack Management**. The **Tech Stack Management** page is displayed.

Step 3 Click **Enable** in the **Operation** column of the target technology stack.

After a technology stack is enabled, its status changes to **Enabling**. All its technology stack versions are enabled at the same time and the status changes **Enabling**.

----End

Disabling a Technology Stack

After a technology stack is disabled, all its technology stack versions are disabled and cannot be used for **Creating and Deploying a Component**.

Technology stacks in the **Enabling** state support this operation.

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Tech Stack Management**. The **Tech Stack Management** page is displayed.

Step 3 Click **Disable** in the **Operation** column of the target technology stack.

After a technology stack is disabled, its status changes to **Disabled**. All its technology stack versions are disabled at the same time and the status changes **Disabled**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

8.1.4 Editing a Custom Technology Stack Version

This section describes how to edit a custom technology stack version.

Prerequisites

- The custom technology stack version has been added and is not enabled.

  To add a custom technology stack version, see **Adding a Custom Technology Stack Version**. To edit an enabled technology stack version, see **Disabling a Technology Stack Version**.
- The technology stack version software package (in .zip or .tar.gz format) to be edited has been uploaded to OBS. For details, see **Streaming Upload (PUT)**.

Editing a Custom Technology Stack Version

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Tech Stack Management**. The **Tech Stack Management** page is displayed.

Step 3 Click ⌄ before the technology stack of the target custom technology stack version to expand the technology stack list.

Step 4 Click **Edit** in the **Operation** column of the target custom technology stack version and set the parameters.

- Java

Parameter	Description
*Tech Stack	Technology stack version name. Enter 2 to 64 characters. Only use letters, digits, slashes (/), underscores (_), and hyphens (-). Start with a letter and end with a letter or digit.
*Tech Stack Version	Technology stack version. The format is A.B.C, where A, B, and C are natural numbers. Example: **1.0.0**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*OS Image	This parameter is mandatory for technology stacks whose **Deployment Type** is **Kubernetes**. Kubernetes is a container orchestration platform that manages containers (usually Docker containers). Each container runs in an isolated environment provided by an OS image. An OS image contains the OS kernel, libraries, tools, and applications required for running a container, ensuring that the container can run consistently in any Docker-enabled environment. – **Built-in**: Select a built-in OS image from the drop-down list. – **Custom**: Click and select the uploaded custom OS image package. **My Images** (private images), **Open Source Images**, **Shared Images**, and **Third-Party Images** are supported. ▪ If you select **My Images**, upload the image to the image repository in advance. For details, see **Uploading an Image**. ▪ If you select **Third-Party Images**, ensure that you have obtained the address of the third-party image. The format of the image address is as follows: *{IP address of the third-party image repository}:{Port number for accessing the third-party image repository}/{Image storage path}/{Image name}:{Image tag}* or *{Image name}:{Image tag}* If the image tag is not specified, the latest version is used by default.
*JDK Version Path	Path for storing the JDK software package that has been uploaded to OBS. 1. Move the cursor to the card of the configured JDK software package. 2. Click . 3. Select the path. 4. Click **OK**.
Description	Technology stack description. Enter up to 512 characters.

- Tomcat

Parameter	Description
*Tech Stack	Technology stack version name. Enter 2 to 64 characters. Only use letters, digits, slashes (/), underscores (_), and hyphens (-). Start with a letter and end with a letter or digit.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Tech Stack Version	Technology stack version. The format is A.B.C, where A, B, and C are natural numbers. Example: **1.0.0**.
*OS Image	This parameter is mandatory for technology stacks whose **Deployment Type** is **Kubernetes**. Kubernetes is a container orchestration platform that manages containers (usually Docker containers). Each container runs in an isolated environment provided by an OS image. An OS image contains the OS kernel, libraries, tools, and applications required for running a container, ensuring that the container can run consistently in any Docker-enabled environment. – **Built-in**: Select a built-in OS image from the drop-down list. – **Custom**: Click and select the uploaded custom OS image package. **My Images** (private images), **Open Source Images**, **Shared Images**, and **Third-Party Images** are supported. ▪ If you select **My Images**, upload the image to the image repository in advance. For details, see **Uploading an Image**. ▪ If you select **Third-Party Images**, ensure that you have obtained the address of the third-party image. The format of the image address is as follows: *{IP address of the third-party image repository}:{Port number for accessing the third-party image repository}/{Image storage path}/{Image name}:{Image tag}* or *{Image name}:{Image tag}* If the image tag is not specified, the latest version is used by default.
*JDK Version Path	Path for storing the JDK software package that has been uploaded to OBS. 1. Move the cursor to the card of the configured JDK software package. 2. Click . 3. Select the path. 4. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Tomcat Version Path	Path for storing the Tomcat software package that has been uploaded to OBS. 1. Move the cursor to the card of the configured Tomcat software package. 2. Click . 3. Select the path. 4. Click **OK**.
Description	Technology stack description. Enter up to 512 characters.

Step 5 Click **OK**.

----End

8.1.5 Deleting a Custom Technology Stack Version

This section describes how to delete a custom technology stack version that is no longer used.

Prerequisites

- The custom technology stack version has been added and is not used for **Creating and Deploying a Component**. For details about how to add a custom technology stack version, see **Adding a Custom Technology Stack Version**.
- The custom technology stack version has been disabled. For details, see **Disabling a Technology Stack Version**.

Deleting a Custom Technology Stack Version

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Tech Stack Management**. The **Tech Stack Management** page is displayed.

Step 3 Click before the technology stack of the target custom technology stack version to expand the technology stack list.

Step 4 Click **Delete** in the **Operation** column of the target custom technology stack version.

Step 5 Click **OK**.

----End

8.2 Directory Settings

Manage the startup users, running directories, and log directories of VM-deployed components. You can view the default directory rules of the system or customize directory rules.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

The default system directory is the default path provided by the system for component deployment and log directories. The default startup user and user who can access the component deployment and log directories are **www**, and the default user group is **www**. For details, see **Viewing the Default System Directory**.

If the default system directory does not meet your service requirements, you can customize the system directory. For details, see **Customizing a System Directory**.

Viewing the Default System Directory

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Directory Settings**. The **Directory Settings** page is displayed.

Step 3 Click the **Default System Directory** tab and view the default system directory settings. For details, see **Table 8-2**.

Table 8-2 Default system directory configuration

Parameter	Description
Deployment Directory	• **Path**: parent directory of the component deployment path. The default value is **/opt/application**, and the absolute deployment path of the component is **/opt/application/*${Component instance name excluding the five random characters generated by the system}}/${Component version}/${Component instance ID}***. For example, if the component instance name is **comp-ecs-env-cce**_8DGO1, the component version number is 2025.0520.19070, and the component instance ID is 41a71313-46a4-411e-9eb5-388b16c9de3a, the absolute deployment path is **/opt/application/comp-ecs-env-cce/2025.0520.19070/41a71313-46a4-411e-9eb5-388b16c9de3a**. For details about the component instance name, see **Context**. • **User:User Group**: owner and group of the files in the component deployment directory. The default values are **www** and **www**, indicating that the file owner is the **www** user and the group to which the file belongs is the **www** group. • **Permissions**: access permissions on the files in the component deployment directory. The default value is **751**, indicating that the file owner (**www**) has the read, write, and execute permissions (**rwx**), the group to which the file belongs (**www**) has the read and execute permissions (**rx**), and other users have the execute permission (**x**).

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Log Directory	• **Path**: parent directory for storing component running logs. The default value is **/var/log/application**, and the absolute path is **/var/log/application/***${Component instance name excluding the five random characters generated by the system}/${Component instance ID}*. For example, if the component instance name is **comp-ecs-env-cce**_8DGO1, and the component instance ID is 41a71313-46a4-411e-9eb5-388b16c9de3a, the absolute storage path is **/opt/application/comp-ecs-env-cce/41a71313-46a4-411e-9eb5-388b16c9de3a**. For details about the component instance name, see **Context**. • **User:User Group**: owner and group of the files in the component log directory. The default values are **www** and **www**, indicating that the file owner is the **www** user and the group to which the file belongs is the **www** group. • **Permissions**: access permissions on the files in the component log directory. The default value is **751**, indicating that the file owner (**www**) has the read, write, and execute permissions (**rwx**), the group to which the file belongs (**www**) has the read and execute permissions (**rx**), and other users have the execute permission (**x**).
Running User	Component startup user. The default value is **www**.

----End

Customizing a System Directory

Step 1 Log in to ServiceStage.

Step 2 Choose **System Settings** > **Directory Settings**. The **Directory Settings** page is displayed.

Step 3 Click the **Custom System Directory** tab and configure the custom system directory by referring to **Table 8-3**.

- Set unique deployment directory for each component.
- Running users cann be changed in directory settings due to component upgrades.
- For servers with a custom system directory, the VM agent version must be 1.3.121 or later.
- If the custom directory does not exist, ensure that the running user has the permission to create the custom directory.

  When the **sudo** command that does not match that in the **/etc/sudoers** file cannot be used on the server where the component is deployed, and you use a custom user (for example, **appuser**) to create a component in the deployment directory (for example, **/opt/test**), ensure that this user has the permission to create directories in the **/opt** directory. You can run the following commands to grant the permission:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

```
sudo setfacl -m u:appuser:rwx /opt
sudo setfacl -R -m u:appuser:rwx /opt
```

Table 8-3 Custom system directory settings

Parameter	Description
Running User	Component startup user. If you use a custom user, ensure that it has been created on the server where the component is running.
Deployment Directory	• **Path**: component deployment path. For example, change it to **/opt/app**. Then, select **Use default system directory** for **Directory Configuration Mode** in **Configuring Directory Settings**. The absolute deployment path of the component is **/opt/app/***${Component instance name excluding the five random characters generated by the system}/${Component version number}/${Component instance ID}/*. For example, if the component instance name is **comp-ecs-env-cce**_8DGO1, the component version number is 2025.0520.19070, and the component instance ID is 41a71313-46a4-411e-9eb5-388b16c9de3a, the absolute deployment path is **/opt/app/comp-ecs-env-cce/2025.0520.19070/41a71313-46a4-411e-9eb5-388b16c9de3a/**. For details about the component instance name, see **Context**. • **User:User Group**: owner and group of the files in the component deployment directory. **User** is the **Running User** you entered. If you customize a **User Group**, ensure that it has been created on the server where the component is running. • **Permissions**: access permissions on the files in the component deployment directory. The value consists of three digits and ranges from 001 to 777. It defines the access permissions of the file owner, file group, and other users.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Log Directory	• **Path**: path for storing component running logs. For example, change it to **/var/log/app**. Then, select **Use default system directory** for **Directory Configuration Mode** in **Configuring Directory Settings**. The absolute storage path is **/var/log/app/***${Component instance name excluding the five random characters generated by the system}/${Component instance ID}*. For example, if the component instance name is **comp-ecs-env-cce**_8DGO1, and the component instance ID is 41a71313-46a4-411e-9eb5-388b16c9de3a, the absolute storage path is **/opt/app/comp-ecs-env-cce/41a71313-46a4-411e-9eb5-388b16c9de3a/**. For details about the component instance name, see **Context**. • **User:User Group**: owner and group of the files in the component log directory. **User** is the **Running User** you entered. If you customize a **User Group**, ensure that it has been created on the server where the component is running. • **Permissions**: access permissions on the files in the component log directory. The value consists of three digits and ranges from 001 to 777. It defines the access permissions of the file owner, file group, and other users.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

9 Deployment Source Management

9.1 Managing the Software Center

9.1.1 Managing Software Packages

To upload a software package to a new SWR software repository, you can create an SWR software repository after selecting an organization during software package creation.

- The SWR software repository does not scan or verify the security of the uploaded software packages. To avoid privacy leakage, do not include privacy information such as unencrypted passwords in uploaded software packages. When downloading public software packages, ensure that they are from trusted repositories and prevent malicious software from being downloaded.
- If a disk is full, software packages cannot be uploaded to the repository and error information is displayed, but services are not affected. To prevent services such as logs from occupying the entire disk, you are advised to attach an independent disk to the repository.

Prerequisites

An organization has been created. For details, see **Creating an Organization**.

Creating a Software Package

Step 1 Log in to ServiceStage. Choose **Deployment Source Management** > **Software Center**, and click **Create Package**.

Step 2 Configure the software package by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 9-1 Software package parameters

Parameter	Description
*Software Repository	Select an organization and a software repository. To create a software repository: 1. Click **Create Repository** and enter a new software repository name. 2. Click ✓.
*Sharing Type	Type of the software repository. The default value is **Private**. • **Private**: only for the current tenant and users under the current tenant. • **Public**: for all tenants and users.
*Package Name	The name must be unique in the repository organization. Enter 1 to 255 characters. Use letters, digits, underscores (_), or hyphens (-). Start with a letter, digit, or underscore (_).
*Version Number	Version number of the software package. Enter 1 to 128 characters. Use letters, digits, underscores (_), or hyphens (-). Start with a letter, digit, or underscore (_).
Package Description	Description of the software package. Enter 0 to 200 characters.
Version Description	Description of the software package version. Enter 0 to 200 characters.
Upload Software	• **Upload now**: Upload the software package by referring to **Step 3.a** in **Uploading a Software Package**. • **Upload later**: After the software package is created, upload it by referring to **Uploading a Software Package**.

Step 3 Click **OK**.

----End

Uploading a Software Package

A maximum of 10 files can be uploaded at a time. The size of a single file (including the decompressed files) cannot exceed 2 GB.

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Software Center**.

Step 2 Select an organization from the drop-down list on the right of **Organization Management**.

Step 3 Use either of the following methods to upload the software package:

- Directly upload

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

a. Click **Upload** next to the target software package.
b. Select a **Package Version**.
c. Click **Select File**, select the target software package, and click **Open**.

   You can also drag the software package to the page.
d. Set the parameters in the following table. All the parameters are optional.

   Table 9-2 Parameter description

Parameter	Description
Overwrite	If you select this option, the software package with the same name in the same path will be overwritten.
File Path	Enter a path to store the software package. The path is the virtual path of the software repository. By default, the root directory is used. By setting the path, you can easily view and manage the software package.

   Repeat the preceding steps to upload other software packages.
e. After the software package is selected:

   - Select a software file from the list of software files to be uploaded and click **Upload** in the **Operation** column to upload the specified software file.
   - In the upper part of the list of software to be uploaded, click **Upload** to upload software files in batches.

- View the software package details and then upload it
  a. Click the target software package to enter the details page.
  b. In the **Version List** area, click **Upload** in the **Operation** column of the target software package version.
  c. Click **Select File**, select the target software package, and click **Open**.

     You can also drag the software package to the page.
  d. Set the parameters in the following table. All the parameters are optional.

     Table 9-3 Parameter description

Parameter	Description
Overwrite	If you select this option, the software package with the same name in the same path will be overwritten.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
File Path	Enter a path to store the software package. The path is the virtual path of the software repository. By default, the root directory is used. By setting the path, you can easily view and manage the software package.

Repeat the preceding steps to upload other software packages.

e. After the software package is selected:

- Select a software file from the list of software files to be uploaded and click **Upload** in the **Operation** column to upload the specified software file.
- In the upper part of the list of software to be uploaded, click **Upload** to upload software files in batches.

----End

Editing a Software Package

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Software Center**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

Step 3 Click the target software package to enter the details page.

Step 4 Click **Edit** in the upper-right corner and set the following parameters:

Parameter	Description
Sharing Type	Set the type of the software repository. • **Private**: only for the current tenant and users under the current tenant. • **Public**: for all tenants and users.
Package Description	Edit the description of the software package. Enter 0 to 200 characters.

Step 5 Click **OK**.

----End

Adding a Software Package Version

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Software Center**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Click the target software package to enter the details page.

Step 4 Click **Add Version** in the upper-right corner and set the following parameters:

Parameter	Description
*Version Number	Version number of the software package. Enter 1 to 128 characters. Use letters, digits, underscores (_), or hyphens (-). Start with a letter, digit, or underscore (_).
Version Description	Edit the description of the software package version. Enter 0 to 200 characters.
Upload Software	• **Upload now**: Upload the software package by referring to **Step 3.a** in **Uploading a Software Package**. • **Upload later**: After the software package is created, upload it by referring to **Uploading a Software Package**.

Step 5 Click **OK**.

----End

Editing a Software Package Version

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Software Center**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

Step 3 Click the target software package to enter the details page.

Step 4 In the **Version List** area, click **Edit** in the **Operation** column of the target software package version.

Step 5 Enter up to 200 characters as the description.

Step 6 Click **OK**.

----End

Querying the Address of a Software Package

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Software Center**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

Step 3 Click the target software package to enter the details page.

Step 4 In the version list, click ⌄ before the target version to view the software package address.

- Move the cursor to the software package address and click ⧉ to copy the **Intranet address** or **External address**.
- In the row where the version file is located, click **Download** to download the version file.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- In the row where the version file is located, click **Delete** to delete the version file.

----End

Deleting a Software Package

Before deleting a software package, ensure that all versions in the software package are deleted. For details, see **Deleting a Software Package Version**.

After a software package is deleted, the software repository and its software package versions cannot be selected when **Creating and Deploying a Component**, **Upgrading a Single Component**, **Upgrading Components in Batches**, **Cloning Components in Batches**, or **Creating a Package Job**.

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Software Center**.

Step 2 Select an organization from the drop-down list on the right of **Organization Management**.

Step 3 Use either of the following methods to delete the software package:

- Delete directly: Click **Delete** on the right of the target software package.
- Delete after viewing software package details: Click the software package to go to the details page and click **Delete** in the upper right corner of the page.

Step 4 Click **OK**.

----End

Deleting a Software Package Version

After a software package version is deleted, it cannot be selected when **Creating and Deploying a Component**, **Upgrading a Single Component**, **Upgrading Components in Batches**, **Cloning Components in Batches**, or **Creating a Package Job**.

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Software Center**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

Step 3 Click the target software package to enter the details page.

Step 4 In the **Version List** area, click **More** > **Delete** in the **Operation** column of the target software package version.

Step 5 Click **OK**.

----End

9.1.2 Packaging Specifications of Software Packages

JAR and WAR packages can be directly uploaded. For other types of software packages, such as compressed packages:

The package name format must be software name+extension. The extension must be tar.gz or zip.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- The extension of a compressed package must be consistent with the compression mode. Otherwise, the software package cannot be decompressed.
- The prefix of a compressed package name must be the same as the name of the root directory after decompression. For example, if the package name is *xxx.tar.gz*, the root directory after decompression must be *xxx*.

Directory Structure

For decompressed software packages, ensure that lifecycle command scripts can be executed.

The following software package directory structure is recommended:

```
├── scripts/
│   │   ├── pre-stop.sh
│   │   ├── stop.sh
│   │   ├── uninstall.sh
│   │   ├── install.sh
│   │   ├── start.sh
│   │   ├── check.sh
│   │   ├── post-start.sh
├── packages/
│   │   ├── xxx.jar/xxx.war
├── config/
├── appspec.yml
```

Table 9-4 Software package directories and files

Directory/ File	Description
scripts	(Mandatory) Stores lifecycle scripts. When creating an application, you can specify execution commands based on the location of lifecycle scripts. For example, specify **bash scripts/install.sh** in the install phase to run the installation script. Lifecycles supported by software package applications are as follows: • **install.sh**: Command for installing software. • **post-start.sh**: Operation performed after software is started. • **start.sh**: Command for starting software. • **restart.sh**: Command for restarting software, which is used to recover the applications failing in health check. • **pres-top.sh**: Operation which is performed before software is stopped. • **stop.sh**: Command for stopping software. • **uninstall.sh**: Command for uninstalling software.
packages	(Mandatory) Stores application JAR or WAR packages.
config	(Mandatory) Stores application configuration information.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Directory/File	Description
appspec.yml	(Mandatory) Records the lifecycle definition and also specifies information such as health check.

9.2 Managing Image Repositories

9.2.1 Uploading an Image

After an organization is created, you can upload an image to it through the page or client.

- **Uploading an Image Through the Page**: Upload an image to SWR through the page.
- **Uploading an Image Through the Client**: Upload an image to an image repository of SWR by running commands on the client.

Image repositories are used to store and manage Docker images.

Prerequisites

- An organization has been created. For details, see **Creating an Organization**.
- The image has been saved as a .tar or .tar.gz file. For details, see **Creating an Image Package**.
- The image package is created using Docker 1.11.2 or later.
- If the image is uploaded through a client, the version of the container engine client to which the image is uploaded must be 1.11.2 or later.

Uploading an Image Through the Page

A maximum of 10 files can be uploaded at a time. The size of a single file (including the decompressed files) cannot exceed 2 GB. The file name should be a string starting with a letter or digit, containing 255 characters at most, and including letters, digits, underscores (_), and hyphens (-).

Step 1 Log in to ServiceStage.

Step 2 Choose **Deployment Source Management** > **Image Repository**.

Step 3 On the **Image Repository** page, click **Upload Through SWR**.

Step 4 In the displayed dialog box, select the organization to which the image is to be uploaded.

Step 5 Click **Select File** and select the target image file.

You can also drag the image file to the page. If you select multiple images to upload, the system uploads them one by one. Concurrent upload is not supported.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 6 In the displayed dialog box, click **Upload**.

If **Upload completed** is displayed, the image is successfully uploaded. If the image fails to be uploaded, the possible causes are as follows:

- The network is abnormal. In this case, check network connectivity.
- The HTTPS certificate has errors. Press **F12** to copy the URL that fails to be requested to the address bar of the browser, open the URL again, agree to continue the access, and return to the upload page to upload the certificate again.

----End

Uploading an Image Through the Client

Each layer of an image uploaded through the client cannot exceed 10 GB.

Step 1 Log in to ServiceStage.

Step 2 Choose **Deployment Source Management** > **Image Repository**.

Step 3 On the **Image Repository** page, click **Upload Through Client**.

Step 4 Upload the image as prompted.

----End

9.2.2 Managing Images

Obtaining an Image Pull Address

Step 1 Log in to ServiceStage.

Step 2 Choose **Deployment Source Management** > **Image Repository** > **My Images**.

Step 3 Select an organization from the drop-down list on the right of **Org Management**.

Step 4 In the image repository list, click an image repository name to go to the details page.

Step 5 Click the **Image Tags** tab and obtain the command for pulling an image.

- **Image Pull Command**: Click on the right of the command of the image tag to be downloaded to copy the command.
- **containerd Command**: Click **containerd Command** in the **Operation** column of the image tag to be downloaded, and click to copy the command.

----End

Setting Image Repository Attributes

Step 1 Log in to ServiceStage.

Step 2 Choose **Deployment Source Management** > **Image Repository** > **My Images**.

Step 3 Select an organization from the drop-down list on the right of **Org Management**.

Step 4 In the image repository list, click an image repository name to go to the details page.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 5 Click **Edit** in the upper-right corner. In the displayed dialog box, perform the following operations:

- Set **Sharing Type** to **Public** or **Private**.

  Public images can be downloaded and used by all users. If your node and the image repository are in the same region, you can access the image repository over private networks. If your node and the image repository are in different regions, the node must have access to public networks to pull images from the image repository.
- Set **Category** to set the repository category.
- Set **Description** to update the description of the image repository.

Step 6 Click **OK**.

----End

Sharing a Private Image

You can share your private images with other users and grant the users access permissions.

Only administrator and Identity and Access Management (IAM) users authorized to manage the private image can share the image. The users with whom you share the image only have the read permission. That is, they can only pull the image.

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Image Repository** > **My Images**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

Step 3 In the image repository list, click an image repository name to go to the details page.

Step 4 Click the **Sharing** tab, click **Share Image**, and set the following parameters:

1. **Share With**: Enter an account name.
2. **Valid Until**: Set the expiration date. If you want the image to be permanently accessible to the account, select **Permanently valid**.
3. **Description**: Enter the description.
4. **Permission**: Select the permission. Currently, only the **Download** permission is supported.

Step 5 Click **OK**.

- You can view all shared images in the shared image list.
- Select an account name and click **Edit** in the **Operation** column to edit the parameters of the shared image.
- Select an account name and click **Delete** in the **Operation** column to cancel sharing.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Setting Automatic Image Synchronization

If image synchronization is enabled, the latest images are automatically synchronized to image repositories in other regions. Only accounts and users with administrator permissions can configure automatic image synchronization.

After you configure automatic image synchronization, image updates will also be synchronized to target repositories. However, images that were pushed to repositories before automatic image synchronization was enabled will not be automatically synchronized. For details on how to synchronize images pushed before you set the automatic synchronization, see **Can Existing Images Be Automatically Synchronized?**

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Image Repository** > **My Images**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

Step 3 In the image repository list, click an image repository name to go to the details page.

Step 4 Click **Set Image Synchronization** in the upper-right corner.

Step 5 In the displayed dialog box, click **Add**, set the following parameters, and click **OK** in the **Operation** column.

- **Target Region**: Select the target region for synchronization.
- **Target Organization**: Select the target organization for synchronization.
- **Overwrite Existing Image**: Select this option if you want to overwrite any nonidentical images that have the same name in the target organization. Deselect this option if you do not want any nonidentical images having the same name in the target organization to be overwritten and you want to receive a notification of the existence of such images.

Step 6 Click **OK**.

On the **Synchronization Records** tab of image details page, you can view the details of each synchronization task, including the start time, image tag, task status, type, duration, target region and organization, and task operator.

----End

Adding Image Permissions

To allow IAM users of your account to read, write, and manage a specific image, add the required permissions to the IAM users on the details page of this image.

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Image Repository** > **My Images**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

Step 3 In the image repository list, click an image repository name to go to the details page.

Step 4 Click the **Permission Management** tab, click **Add Permission**, select an IAM user, add the **Read**, **Write**, or **Manage** permission, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Then, this IAM user has the corresponding permission.

----End

Deleting an Image

After an image is deleted, the image repository and its image packages cannot be selected when **Creating and Deploying a Component**, **Upgrading a Single Component**, **Upgrading Components in Batches**, or **Cloning Components in Batches**.

Step 1 Log in to ServiceStage and choose **Deployment Source Management > Image Repository > My Images**.

Step 2 Select an organization from the drop-down list on the right of **Org Management**.

Step 3 In the image repository list, click an image repository name to go to the details page.

- Deleting an image repository

  Click **Delete** in the upper-right corner of the page and delete the image repository as prompted.

- Deleting an image tag

  In the **Operation** column of the target image tag, click **Delete** to delete the image tag as prompted.

- Deleting image tags in batches

  Select the target image tags, click **Delete** above the tag list, and delete the image tags as prompted.

----End

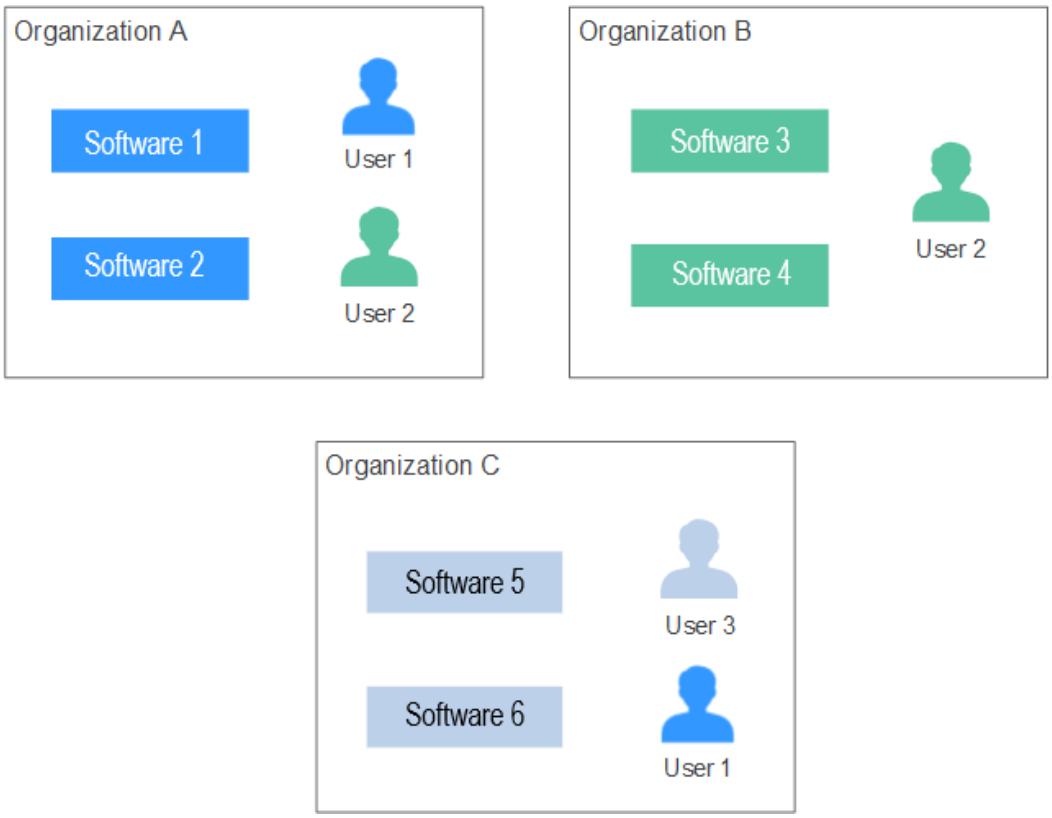
9.3 Managing Organizations

Overview

Organizations are used to isolate software and image repositories. With each organization being limited to one company or department, software can be managed in a centralized manner. A software name only needs to be unique within an organization. The same IAM user can join different organizations. Different permissions, namely read, write, and manage, can be assigned to different IAM users in the same account.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 9-1 Organization



Creating an Organization

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Organization**.

Step 2 Click **Create Organization**, enter **Organization Name**, and click **OK**.

----End

Adding Permissions

Grant permissions to users in an organization so that they can read, edit, and manage all images in the organization.

Only users with the **Management** permission can grant permissions.

User permissions include:

- **Read-only**: Users can only download software but cannot upload software.
- **Read/write**: Users can download software, upload software, and edit software attributes.
- **Management**: Users can download and upload software, delete software or versions, edit software attributes, grant permission, and share images.

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Organization**.

Step 2 Click **Add Permission** on the right of an organization.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 In the displayed dialog box, specify **Permission** and click **OK**.

----End

Deleting an Organization

After an organization is deleted, the images in the organization are also deleted. As a result, **Upgrading a Single Component** or **Upgrading Components in Batches** may fail.

Step 1 Log in to ServiceStage and choose **Deployment Source Management** > **Organization**.

Step 2 Click **Delete** on the right of an organization.

Before deleting an organization, delete the image and software repositories of the organization.

For details about how to delete an image repository, see **Deleting an Image**.

For details about how to delete a software repository, see **Deleting a Software Package**.

Step 3 Click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

10 Continuous Delivery

10.1 Continuous Delivery Overview

Continuous delivery provides functions such as project build and release.

Build Job Overview

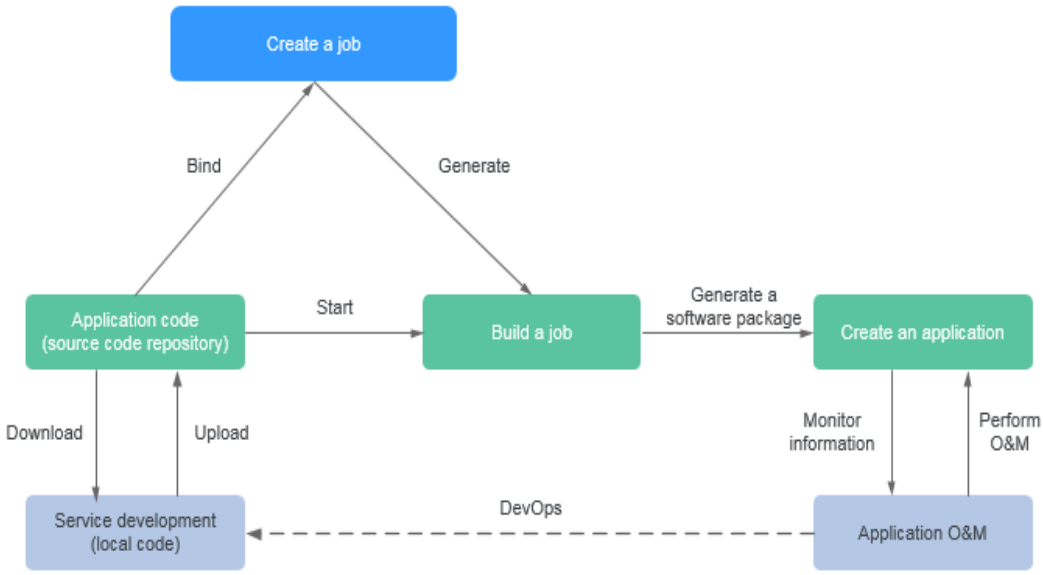
Based on the existing service code, you can create a build job, start the build job, package the service code, and archive the package to the deployment source. Then, you can use the package when deploying an application component.

There are two types of build jobs:

- System build jobs are created by the system when **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**.
- User build jobs are created when **Creating a Source Code Job** and **Creating a Package Job**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

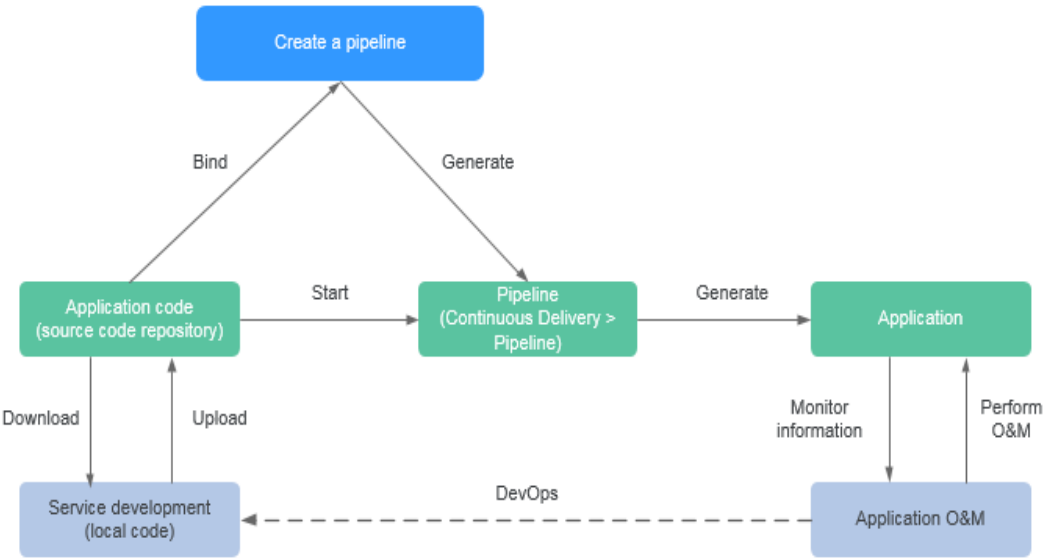
Figure 10-1 Creating a build job



Pipeline Overview

Based on the existing service code, you can create a pipeline and then start the pipeline to complete service code building and deployment. Application O&M can also be completed on ServiceStage.

Figure 10-2 Creating a pipeline



10.2 Viewing Build Jobs

You can view the build records and logs of a specified build job in the build job list to locate and rectify faults that occur during component deployment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

You have created a build job of the Java, Tomcat, Node.js, Python, or PHP technology stack.

- For system-created build jobs, see **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**.
- For user-created build jobs, see **Creating a Source Code Job** and **Creating a Package Job**.

Viewing Build Jobs

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Build**.

Step 3 On the **Build** page, use either of the following methods to search for a build job:

- Select the CCE cluster where the component is deployed and the build job status, and select the specified build job in the build list.
- Search for the build job in the search box.

Step 4 Click the build task name. The build task details page is displayed.

- View **Basic Information** and **Build Record** of the build job.
- Click **View Log** next to a build record to view **Build Details**, **CodeCheck**, and **Log**.

  Only the Maven build project supports code check. Currently, the following code check plug-ins are supported: Checkstyle, FindBugs, and PMD.

----End

10.3 Creating a Source Code Job

The software package or image package can be generated with a few clicks in a build job. In this way, the entire process of source code pull, compilation, packaging, and archiving is automatically implemented.

- Images built in the x86-system jobs are ones of the x86 system.
- Images built in the Arm-system jobs are ones of the Arm system.

Prerequisites

- A cluster has been created. For details, see **Buying a CCE Standard/Turbo Cluster**. Create a CCE Turbo or CCE Standard cluster based on your service requirements.
  - The build job starts a build container on the cluster node to perform build-related operations. To ensure build security, you are advised to perform security hardening on CCE cluster nodes. For details, see **Forbidding Containers to Obtain Host Machine Metadata**.
  - If you create a CCE Turbo cluster, set SNAT rules for the cluster so that the cluster can access the public network through the NAT gateway to pull source code. Otherwise, when a source code-build task is executed,

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

the source code fails to be obtained because the public network cannot be accessed. As a result, the component deployment or build fails. For details about how to set SNAT rules for the cluster, see **Accessing the Internet from a Container**.

- The build job depends on the JDK, Golang, Maven, Gradle, Ant, or Node.js compilation tool preconfigured in the build container.
- Different IAM users under the same account can perform operations on the same cluster. To cancel the build permission from a specific IAM user, set the **servicestage:assembling:create**, **servicestage:assembling:modify**, and **servicestage:assembling:delete** permissions to **Deny** by referring to **Creating a Custom Policy**.

- An EIP has been bound to the build node. For details, see **Binding an EIP to an Instance**.
- The CCE cluster must be of the current commercial version. CCE clusters that have reached EOS are not supported. The OS of the corresponding node must be of the commercial version. OS versions that have reached EOS are not supported.

Creating a Source Code Job

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Build**.

Step 3 Click **Create Source Code Job** and set basic information about the build job by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Name	Unique build job name. Enter 1 to 64 characters. Start with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
*Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after you **enable the enterprise project function**.
Description	Build job description. Enter up to 64 characters.
*Code Source	The technology stack used by the source code must be Java, Tomcat, Node.js, Python, or PHP. • If you select a source code repository, create authorization by referring to **Creating Repository Authorization** and set the code source. • Click **Samples** and select a required sample.
*Cluster	Select a cluster for build. The cluster must belong to the configured enterprise project.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Key	Set **Key** and **Value** of the node label and deliver the build job to the fixed node bound with an EIP. For details about how to add a label, see **Adding a Node Label**.
Value	

Step 4 Click **Next**.

Step 5 Select a build template.

- If you select **Maven**, **Ant**, **Gradle**, **Go**, or **Docker**, you can compile and archive binary packages or Docker images at the same time. Go to **Step 6**.
- If you select **Custom**, you can customize the build mode. Go to **Step 8**.

Step 6 Select an archive mode.

- **Not archived**: No Docker build job is added or archived.
- **Archive binary package**: No Docker build job is added and binary packages are archived.
- **Archive image compilation**: Docker build job is added and Docker images are archived.

Step 7 Set mandatory parameters based on the archive mode selected in **Step 6**.

Step 8 Click **Advanced Settings** to configure the build environment.

1. Click **Add Plug-in** in the left pane. The **Select Job Type** page is displayed.
2. Click **Select** next to the target job type, add a job type, and set job parameters based on the task type.

   When the Build Common Cmd plug-in is added to the compilation process, pay attention to the following:

   - Exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage.
   - Enter the compilation command. A maximum of 512 characters are allowed. Otherwise, an error message is displayed, indicating that the task input parameter is incorrect. In this case, you can add multiple Build Common Cmd plug-ins to split the command.
   - When **Language** is set to **Python** and **Python Framework Type** is set to a Python project that complies with the **WSGI** standard, you need to set **Main Python Module** and **Function of the Main Python Module**. The following is an example of the main Python module and main function:

     Main Python Module: If the entry point file of the Python project is **server.py**, the main module name is **server**.

     Function of the Main Python Module: If the application function name of the Python project entry point file **server.py** is **app=get_wsgi_application()**, the function name of the main module is **app**.

Step 9 Determine whether to execute the build:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If yes, click **Build** to save the settings and execute the build job.
- If no, click **Save** to save the settings only.

----End

Follow-up Operations

- After the build, you can manage the application component on ServiceStage. For details, see **Creating a Component Based on a Container Using Manual Configurations** or **Creating a Component Based on a Container Using YAML Configurations**.
- If the build fails, rectify the fault by referring to **What Do I Do If ServiceStage Failed to Build a Job?**

10.4 Creating a Package Job

The image package can be generated with a few clicks in a build job. In this way, the entire process of package obtainment, and image compilation and archiving is automatically implemented.

Prerequisites

- A cluster has been created. For details, see **Buying a CCE Standard/Turbo Cluster**. Create a CCE Turbo or CCE Standard cluster based on your service requirements.
  - The build job starts a build container on the cluster node to perform build-related operations. To ensure build security, you are advised to perform security hardening on CCE cluster nodes. For details, see **Forbidding Containers to Obtain Host Machine Metadata**.
  - The build job depends on the JDK, Golang, Maven, Gradle, Ant, or Node.js compilation tool preconfigured in the build container.
  - Different IAM users under the same account can perform operations on the same cluster. To cancel the build permission from a specific IAM user, set the **servicestage:assembling:create**, **servicestage:assembling:modify**, and **servicestage:assembling:delete** permissions to **Deny** by referring to **Creating a Custom Policy**.
- An EIP has been bound to the build node. For details, see **Binding an EIP to an Instance**.

Creating a Package Job

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Build**.

Step 3 Click **Create Package Job** and set basic information about the build job by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Job Name	Unique build job name. Enter 1 to 64 characters. Start with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).
*Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after you **enable the enterprise project function**.
Description	Build job description. Enter up to 64 characters.
*Package Source	The technology stack used by the software package must be Java, Tomcat, Node.js, Python, or PHP. • SWR Click **Select Software Package** and select the corresponding software package. Upload the software package to the software repository in advance. For details, see **Uploading a Software Package**. • OBS Click **Select Software Package** and select the corresponding software package. Upload the software package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**.
*Build Type	• System default 1. Select a basic image, which must be the same as the software package compilation language selected for **Package Source**. 2. Set **Basic Image Tag**. • Custom Dockerfile Enter custom commands in the compilation box. Exercise caution when inputting sensitive information in the **echo**, **cat**, or **debug** command, or encrypt sensitive information to avoid information leakage. • Image Select a basic image, which must be the same as the software package compilation language selected for **Package Source**.
*Image Class	• **Public**: This is a widely used standard image that contains an OS and pre-installed public applications and is visible to all users. You can configure the applications or software in the public image as needed. • **Private**: A private image contains an OS or service data, pre-installed public applications, and private applications. It is available only to the user who created it.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Archived Image Address	Archive path of the image.
*Cluster	Select a cluster for build. The cluster must belong to the configured enterprise project. If you use the selected cluster to perform a build job, you can deliver the build job to a fixed node through node labels. For details about how to add a label, see **Adding a Node Label**.

Step 4 Determine whether to execute the build:

- If yes, click **Build** to save the settings and execute the build job.
- If no, click **Save** to save the settings only.

----End

Follow-up Operations

- After the build, you can manage the application component on ServiceStage. For details, see **Creating a Component Based on a Container Using Manual Configurations** or **Creating a Component Based on a Container Using YAML Configurations**.
- If the build fails, rectify the fault by referring to **What Do I Do If ServiceStage Failed to Build a Job?**

10.5 Maintaining Build Jobs

10.5.1 Editing a Source Code Job

This section describes how to edit a source code job.

Prerequisites

The source code job to be edited is available.

- For system-created build jobs, see **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**.
- For user-created build jobs, see **Creating a Source Code Job**.

Editing a Source Code Job

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Build**.

Step 3 On the **Build** page, use either of the following methods to search for a build job:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Select **User created** and select the CCE cluster and build job status. Then, select the target build job in the build list.
- Search for the build job created by the specified user in the search box.

Step 4 Click **More** > **Edit**. The build job configuration page is displayed.

Step 5 Enter a name.

Enter 1 to 64 characters. Start with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).

Step 6 (Optional) Enter the description, which contains up to 64 characters.

Step 7 Click **Modify** and set **Code Source**.

You need to create repository authorization first. For details, see **Creating Repository Authorization**.

Step 8 Select **Cluster**.

1. (Optional) Specify **Node Label** to deliver the build job to a fixed node based on the node label.

   For details about how to add a label, see **Adding a Node Label**.
2. Click **Next**.

Step 9 Set the environment.

1. Edit a build template.

   Select **Maven**, **Ant**, **Gradle**, **Go**, **Docker**, or **Build Common Cmd**. You can compile and archive binary packages or Docker images at the same time.

   When using the Build Common Cmd template for build, enter a compilation command that contains up to 512 characters. If there are more than 512 characters, an error message is displayed, indicating that the task input parameter is incorrect. In this case, you can add multiple Build Common Cmd plug-ins to split the command.
2. Select an archive mode.
   - **Publish Build Artifact**: Binary package archive plug-in, archived to the SWR software repository.
   - **Publish Build Image**: Image archive plug-in, archived to the SWR image repository.

Step 10 Determine whether to execute the build:

- If yes, click **Build** to save the settings and execute the build job.
- If no, click **Save** to save the settings only.

----End

10.5.2 Editing a Package Job

This section describes how to edit a package job.

Prerequisites

The package job to be edited is available.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For system-created build jobs, see **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**.
- For user-created build jobs, see **Creating a Package Job**.

Editing a Package Job

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Build**.

Step 3 On the **Build** page, use either of the following methods to search for a build job:

- Select **User created** and select the CCE cluster and build job status. Then, select the target build job in the build list.
- Search for the build job created by the specified user in the search box.

Step 4 Click **More** > **Edit**. The build job configuration page is displayed.

Step 5 Enter a job name.

Enter 1 to 64 characters. Start with a letter or digit. Only use letters, digits, underscores (_), and hyphens (-).

Step 6 (Optional) Enter the description, which contains up to 64 characters.

Step 7 Set **Package Source**.

The following upload methods are supported:

- Select the corresponding software package from the SWR software repository. Upload the software package to the software repository. For details, see **Uploading a Software Package**.

  If PKIToken is disabled, the SWR repository is not supported.
- Select the corresponding software package from OBS. Upload the software package to the OBS bucket in advance. For details, see **Streaming Upload (PUT)**.

Step 8 Set **Build Type**.

- System default
  a. Select the language of the basic image, which must be the same as that of the software package.
  b. Set **Basic Image Tag**.

     The build node can download basic images only when it can access the public network.
- Custom Dockerfile

  Enter custom commands in the compilation box.
- Image

  Set **Basic Image**.

Step 9 Set **Image Class**.

- **Public**: This is a widely used standard image that contains an OS and pre-installed public applications and is visible to all users. You can configure the applications or software in the public image as needed.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Private**: A private image contains an OS or service data, pre-installed public applications, and private applications. It is available only to the user who created it.

Step 10 Specify **Archived Image Address**.

Step 11 Select **Cluster**.

Step 12 (Optional) Specify **Node Label** to deliver the build job to a fixed node based on the node label.

For details about how to add a label, see **Adding a Node Label**.

Step 13 Determine whether to execute the build:

- If yes, click **Build** to save the settings and execute the build job.
- If no, click **Save** to save the settings only.

----End

10.5.3 Starting a Build Job

This section describes how to start a build job.

Prerequisites

The build job to be started is available.

- For system-created build jobs, see **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**.
- For user-created build jobs, see **Creating a Source Code Job** and **Creating a Package Job**.

Starting a Build Job

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Build**.

Step 3 On the **Build** page, use either of the following methods to search for a build job:

- Select the CCE cluster where the component is deployed and the build job status, and select the specified build job in the build list.
- Search for the build job in the search box.

Step 4 Click **Build Now** to start the build.

----End

10.5.4 Configuring Branch or Tag Build

In software development and version control systems, branches and tags are different:

- A branch is an independent development line in the version control system. It allows developers to create a backup from the main or master branch to develop or repair new functions without affecting the stability of the main or

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

master branch. Branch build is usually used for continuous integration (CI). Each time code is pushed to a branch, the CI system automatically pulls the latest code and performs build and test.

- A tag is for a specific commit, and is usually used to mark a released version. Tag build is usually used for continuous deployment (CD). When the code is marked as a new version, the CD system can automatically build and deploy the code from that tag.

This section describes how to configure branch or tag build.

Prerequisites

The source code job is available.

- For system-created build jobs, see **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**.
- For user-created build jobs, see **Creating a Source Code Job**.

Configuring Branch or Tag Build

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Build**.

Step 3 On the **Build** page, use either of the following methods to search for a build job:

- Select the CCE cluster where the component is deployed and the build job status, and select the specified build job in the build list.
- Search for the build job in the search box.

Step 4 Click **Branch/Tag** and set build parameters.

1. Select **Branch/Tag**.
2. Select the corresponding branch or tag from the drop-down list.
3. Specify **Commit ID** for the branch or tag.

   The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Step 5 Click **OK**.

----End

10.5.5 Deleting a Build Job

Delete a user-created build job that is no longer used. After a build job is deleted, **Creating and Deploying a Component**, **Upgrading a Single Component**, or **Upgrading Components in Batches** may fail.

To delete a system-created build job, see **Deleting a Component**.

Prerequisites

The user-created build job to be deleted is available. For details, see **Creating a Source Code Job** and **Creating a Package Job**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Deleting a Build Job

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Build**.

Step 3 On the **Build** page, use either of the following methods to search for a build job:

- Select **User created** and select the CCE cluster and build job status. Then, select the target build job in the build list.
- Search for the build job created by the specified user in the search box.

Step 4 Click **More** > **Delete**.

Step 5 Click **OK**.

----End

10.6 Managing Pipelines

10.6.1 Pipeline Overview

You can set up a pipeline project to continuously and quickly deploy applications. One-click deployment can be achieved through pipeline. In this way, the entire process of source code pull, compilation, packaging, archiving, and deployment is automatically implemented. This unifies the integration environment and standardizes the delivery process.

Each pipeline includes a group of build jobs and one or more groups of environment (such as development, production-like, and production environments) tasks. Each group of environment tasks contains one or more subtasks (such as deployment and test tasks) and provides templates.

ServiceStage allows a single user to create a maximum of 100+N pipelines in a project. N indicates the total number of components created by the user.

10.6.2 Creating a Pipeline

This section describes how to create a pipeline.

Prerequisites

A build job has been created. For details, see **Creating a Source Code Job** or **Creating a Package Job**.

Creating a Pipeline

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Click **Create Pipeline** and set basic pipeline information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Pipeline	Pipeline name. Enter 1 to 99 characters. Start with a letter. Only use letters, digits, periods (.), at signs (@), and hyphens (-).
*Enterprise Project	Enterprise projects let you manage cloud resources and users by project. It is available after you **enable the enterprise project function**.
Description	Pipeline description. Enter up to 255 characters.

Step 4 Click **Back to Template**, and select and configure a pipeline template by referring to the following table.

ServiceStage provides built-in pipeline templates for typical scenarios. After a pipeline template is selected, a pipeline model is automatically generated.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Name	Description
Empty Template	You need to add a pipeline model. 1. Click the **Empty Template** card. 2. Add a build job to generate an image package. a. Click **Select Build Job**. b. Select a build job. If there is no build job, create one by referring to **Creating a Source Code Job** or **Creating a Package Job**. c. Click **OK**. Repeat this step to add more build jobs. The build job must belong to the enterprise project set in **Step 3**. 3. Add a deployment environment to deploy the image package generated by build job to the environment. a. Click **Add Environment**. b. Click and enter an environment name. c. Select the deployed application component in the build job and click **OK**. If there is no application component in the environment, create and deploy one by referring to **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**. Click **Change Component** to select another component. d. **Select Build Job**: Select the added build job for build. e. **Build Output**: Set the archive path of the image generated by the build job. f. Click **OK**. Repeat this step to add more deployment environments. 4. Configure pipeline approval to determine whether to deploy the image package generated during build to the environment. a. Click on an environment card. b. Set **Approval Mode**. **By all**: The image package generated during build can be deployed in the environment only after all selected approvers approve the deployment. **By one person**: The image package generated during build can be deployed in the environment after any approver approves the deployment. c. Set **Approved By**. You can select multiple approvers. The system automatically loads all subaccounts of the account. d. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Name	Description
Simple Template	The "build" model is automatically added to compile and build the source code of the code library. 1. Select **Simple Template**. 2. Click **Create**. 3. Add a build job to generate an image package. a. Click on a build job card. b. Select a build job. If there is no build job, create one by referring to **Creating a Source Code Job** or **Creating a Package Job**. c. Click **OK**. Click **Select Build Job** to add multiple build jobs. The build job must belong to the enterprise project set in **Step 3**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Name	Description
Common Template	The "Build/Production environment" model is automatically added to compile and build the source code of the code library, and the generated software package or image is continuously released to the production environment. 1. Select **Common Template**. 2. Click **Create**. 3. Add a build job to generate an image package. a. Click on a build job card. b. Select a build job. If there is no build job, create one by referring to **Creating a Source Code Job** or **Creating a Package Job**. c. Click **OK**. Click **Select Build Job** to add multiple build jobs. The build job must belong to the enterprise project set in **Step 3**. 4. Add a deployment environment to deploy the image package generated by build job to the environment. a. Click **Add Environment**. b. Click on the empty environment card. The **Add Environment** dialog box is displayed. c. Click and enter an environment name. d. Click to save the environment name. e. Select the deployed application component in the build job and click **OK**. If there is no application component in the environment, create and deploy one by referring to **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**. Click **Change Component** to select another component. f. **Select Build Job**: Select the added build job for build. g. **Build Output**: Set the archive path of the image generated by the build job. h. Click **OK**. Repeat this step to add more deployment environments. 5. Configure pipeline approval to determine whether to deploy the image package generated during build to the environment. a. Click on an environment card. b. Set **Approval Mode**. **By all**: The image package generated during build can be deployed in the environment only after all selected approvers approve the deployment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Name	Description
	By one person: The image package generated during build can be deployed in the environment after any approver approves the deployment. c. Set **Approved By**. You can select multiple approvers. The system automatically loads all subaccounts of the account. d. Click **OK**.

Step 5 Determine whether to start the pipeline:

- If yes, click **Create and Execute** to save the settings and start the pipeline. After the pipeline is started, its status is **Successful** .

  If the pipeline need to be approved, it is in the **To be confirmed** state during component deployment. The pipeline can continue to run only after you notify the approvers for approval. For details, see **Approving a Pipeline**.
- If no, click **Create** to save the settings only. After the pipeline is created, its status is **Not executed**.

To view the pipeline, see **Viewing Pipeline Details**.

----End

10.6.3 Configuring a Pipeline Triggering Policy

This section describes how to configure a pipeline triggering policy so that a pipeline can be started when the configured triggering policy is met.

Prerequisites

The pipeline is not in the **Running** state. To view the pipeline status, see **Viewing Pipeline Details**.

Configuring a Pipeline Triggering Policy

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Select a pipeline, choose **More** > **Triggering Policy**, and set the parameters by referring to the following table.

Triggering Policy	Mode	Description
Manual	-	Manually start the pipeline. 1. Set **Type** to **Manual**. 2. Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Triggering Policy	Mode	Description
Scheduled	Single-time	Set the triggering time to trigger a single-time pipeline. 1. Set **Type** to **Scheduled**. 2. Disable **Periodic Triggering**. 3. Set **Triggered**. 4. Click **OK**.
	Periodic	Set the triggering time segment, interval, and period to implement periodic pipeline triggering. 1. Set **Type** to **Scheduled**. 2. Enable **Periodic Triggering**. 3. Set **Period**, **Triggered**, **Effective Time**, and **Period**. 4. Click **OK**.
Automatic	-	Only pipelines whose build job source is source code support automatic triggering. Set the code source, corresponding namespace, repository name, and branch. When code is submitted to the corresponding branch of the source code repository, the pipeline is automatically triggered. You can set a maximum of eight trigger sources. 1. Set **Type** to **Automatic**. 2. Select **Source Code Repository** to push the code to the selected source code repository. 3. Click **OK**.

----End

10.6.4 Editing a Pipeline

This section describes how to edit a pipeline.

Prerequisites

The pipeline has been created and is not in the **Running** state. To view the pipeline status, see **Viewing Pipeline Details**.

Editing a Pipeline

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Use either of the following methods to go to the **Edit Pipeline** page:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- Directly edit: Select the pipeline and click **Edit**.
- View details and then edit: Click the pipeline. On the **Pipeline Details** page displayed, view the details by referring to **Viewing Pipeline Details** and then click **Edit**.

Step 4 Edit the basic pipeline information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Pipeline	Pipeline name. Enter 1 to 99 characters. Start with a letter. Only use letters, digits, periods (.), at signs (@), and hyphens (-).
Description	Pipeline description. Enter up to 255 characters.

Step 5 Modify the pipeline configuration based on the pipeline template selected in **Creating a Pipeline**.

Operation	Procedure
Edit a build job	1. Move the cursor to the build job. – Click to select another build job. – Click to delete the build job. 2. Click **OK**.
Add a build job	1. Click **Select Build Job**. 2. Select a build job. If there is no build job, create one by referring to **Creating a Source Code Job** or **Creating a Package Job**. 3. Click **OK**. Repeat this procedure to add more build jobs. The build job must belong to the enterprise project set in **Creating a Pipeline**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Procedure
Edit a deployment environment	1. Move the cursor to the deployment environment. 2. Click . The **Add Environment** dialog box is displayed. 3. Click and enter an environment name. 4. Select the deployed application component in the build job and click **OK**. – If there is no application component in the environment, create and deploy one by referring to **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**. – Click **Change Component** to select another component. 5. **Select Build Job**: Select the added build job for build. 6. **Build Output**: Set the archive path of the image generated by the build job. 7. Click **OK**.
Add a deployment environment	1. Click **Add Environment**. 2. Click and enter an environment name. 3. Click to save the environment name. 4. Select the deployed application component in the build job and click **OK**. – If there is no application component in the environment, create and deploy one by referring to **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**. – Click **Change Component** to select another component. 5. **Select Build Job**: Select the added build job for build. 6. **Build Output**: Set the archive path of the image generated by the build job. 7. Click **OK**. Repeat this procedure to add more deployment environments.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Procedure
Edit pipeline approval information	1. Click on an environment card. 2. Set **Approval Mode**. **By all**: The pipeline can be executed only after all selected approvers approve the deployment. **By one person**: The pipeline can be executed after any approver approves the deployment. 3. Set **Approved By**. You can select multiple approvers. The system automatically loads all subaccounts of the account. 4. Click **OK**.

Step 6 Determine whether to start the pipeline:

- If yes, click **Save and Execute** to save the settings and start the pipeline. After the pipeline is started, its status is **Successful** .

  If the pipeline need to be approved, it is in the **To be confirmed** state during component deployment. The pipeline can continue to run only after you notify the approvers for approval. For details, see **Approving a Pipeline**.
- If no, click **Save** to save the settings only. After the pipeline is created, its status is **Not executed**.

To view the pipeline, see **Viewing Pipeline Details**.

----End

10.6.5 Cloning a Pipeline

You can clone a pipeline to generate a new pipeline based on the existing pipeline configuration.

Prerequisites

The pipeline has been created and is not in the **Running** state. For details, see **Creating a Pipeline**.

Cloning a Pipeline

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Use either of the following methods to go to the **Clone Pipeline** page:

- Directly clone: Select the pipeline and choose **More** > **Clone**.
- View details and then clone: Click the pipeline. On the **Pipeline Details** page displayed, click **Clone**.

Step 4 ServiceStage automatically loads configurations of the clone pipeline. You can then modify the configurations as required by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Name	Description
Empty Template	You need to add a pipeline model. 1. Click the **Empty Template** card. 2. Add a build job. a. Click **Select Build Job**. b. Select a build job. If there is no build job, create one by referring to **Creating a Source Code Job** or **Creating a Package Job**. c. Click **OK**. Repeat this step to add more build jobs. The build job must belong to the enterprise project set in **Step 3**. 3. Add a deployment environment a. Click **Add Environment**. b. Click and enter an environment name. c. Click to save the environment name. d. Select the deployed application component in the build job and click **OK**. If there is no application component in the environment, create and deploy one by referring to **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**. Click **Change Component** to select another component. e. **Select Build Job**: Select the added build job for build. f. **Build Output**: Set the archive path of the image generated by the build job. g. Click **OK**. Repeat this step to add more deployment environments. 4. Set pipeline approval. a. Click on an environment card. b. Set **Approval Mode**. **By all**: The pipeline can be executed only after all selected approvers approve the deployment. **By one person**: The pipeline can be executed after any approver approves the deployment. c. Set **Approved By**. You can select multiple approvers. The system automatically loads all subaccounts of the account.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Name	Description
Simple Template	The "build" model is automatically added to compile and build the source code of the code library. 1. Select **Simple Template**. 2. Click **Create**. 3. Add a build job. a. Click on a build job card. b. Select a build job. If there is no build job, create one by referring to **Creating a Source Code Job** or **Creating a Package Job**. c. Click **OK**. Click **Select Build Job** to add multiple build jobs. The build job must belong to the enterprise project set in **Creating a Pipeline**. 4. (Optional) Add a deployment environment. a. Click **Add Environment**. b. Click and enter an environment name. c. Click to save the environment name. d. Select the deployed application component in the build job and click **OK**. If there is no application component in the environment, create and deploy one by referring to **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**. Click **Change Component** to select another component. e. **Select Build Job**: Select the added build job for build. f. **Build Output**: Set the archive path of the image generated by the build job. g. Click **OK**. Repeat this step to add more deployment environments. 5. Set pipeline approval. If a deployment environment is added, you can set pipeline approval. a. Click on an environment card. b. Set **Approval Mode**. **By all**: The pipeline can be executed only after all selected approvers approve the deployment. **By one person**: The pipeline can be executed after any approver approves the deployment. c. Set **Approved By**. You can select multiple approvers. The system automatically loads all subaccounts of the account.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Name	Description
Common Template	The "Build/Production environment" model is automatically added to compile and build the source code of the code library, and the generated software package or image is continuously released to the production environment. 1. Select **Common Template**. 2. Click **Create**. 3. Add a build job. a. Click on a build job card. b. Select a build job. If there is no build job, create one by referring to **Creating a Source Code Job** or **Creating a Package Job**. c. Click **OK**. Click **Select Build Job** to add multiple build jobs. The build job must belong to the enterprise project set in **Creating a Pipeline**. 4. Add a deployment environment a. Click **Add Environment**. b. Click on the empty environment card. The **Add Environment** dialog box is displayed. c. Click and enter an environment name. d. Click to save the environment name. e. Select the deployed application component in the build job and click **OK**. If there is no application component in the environment, create and deploy one by referring to **Creating a Component Based on a Container Using Manual Configurations** and **Creating a Component Based on a Container Using YAML Configurations**. Click **Change Component** to select another component. f. **Select Build Job**: Select the added build job for build. g. **Build Output**: Set the archive path of the image generated by the build job. h. Click **OK**. Repeat this step to add more deployment environments. 5. Set pipeline approval. a. Click on an environment card. b. Set **Approval Mode**. **By all**: The pipeline can be executed only after all selected approvers approve the deployment. **By one person**: The pipeline can be executed after any approver approves the deployment.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Template Name	Description
	c. Set **Approved By**. You can select multiple approvers. The system automatically loads all subaccounts of the account.

Step 5 Determine whether to start the pipeline:

- If yes, click **Create and Execute** to save the settings and start the pipeline. After the pipeline is started, its status is **Successful** .

  If the pipeline need to be approved, it is in the **To be confirmed** state during component deployment. The pipeline can continue to run only after you notify the approvers for approval. For details, see **Approving a Pipeline**.
- If no, click **Create** to save the settings only. After the pipeline is created, its status is **Not executed**.

To view the pipeline, see **Viewing Pipeline Details**.

----End

10.6.6 Starting and Stopping a Pipeline

Starting a Pipeline

The pipeline is not in the **Running** state. To view the pipeline status, see **Viewing Pipeline Details**.

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Start the pipeline in either of the following methods:

- Directly start: Select the pipeline and click **Start**.
- View details and then start: Click the pipeline. On the **Pipeline Details** page displayed, view the details by referring to **Viewing Pipeline Details** and then click **Start**.

If the pipeline need to be approved, it is in the **To be confirmed** state during component deployment. The pipeline can continue to run only after you notify the approvers for approval. For details, see **Approving a Pipeline**.

After the pipeline is started, you can build and deploy applications in one-click mode. For details about maintenance operations after application components are deployed, see **Component O&M**.

----End

Stopping a Pipeline

Step 1 Log in to ServiceStage.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Stop the pipeline in either of the following methods:

- Directly stop: Select the pipeline and click **Stop**.
- View details and then stop: Click the pipeline. On the **Pipeline Details** page displayed, view the details by referring to **Viewing Pipeline Details** and then click **Stop**.

----End

10.6.7 Approving a Pipeline

This section describes how to approve a started pipeline with approval requirements. Pipeline approval is used to control the release and deployment of application components through approval processes and standards to improve the work efficiency and quality of the team.

Prerequisites

The pipeline has been started and is in the **To be confirmed** state.

- To create a pipeline with approval requirements, see **Creating a Pipeline**.
- To start a pipeline, see **Starting and Stopping a Pipeline**.
- To view the pipeline status, see **Viewing Pipeline Details**.

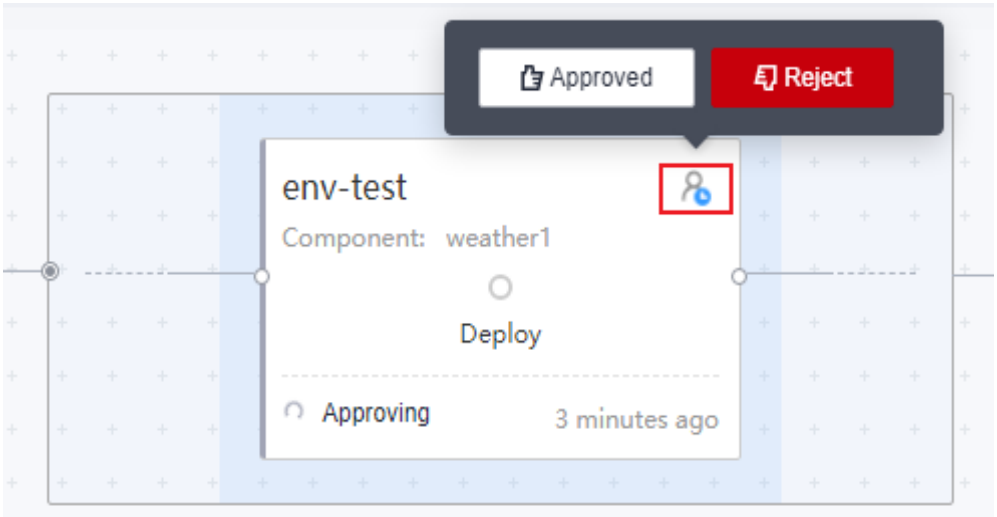
Approving a Pipeline

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Click a pipeline to go to its details page.

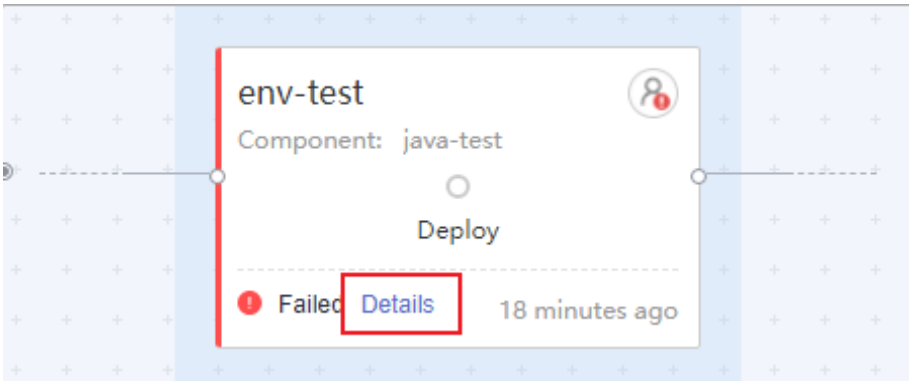
Step 4 Click on the target deployment environment card and determine whether to approve the deployment.



- **Approved**: The pipeline deployment continues until the deployment is complete.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- **Reject**: The pipeline deployment fails. Click **Details** on the deployment environment card to view the failure cause.



----End

10.6.8 Viewing Pipeline Details

This section describes how to view pipeline details.

Prerequisites

A pipeline has been created. For details, see **Creating a Pipeline**.

Viewing Pipeline Details

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Click a pipeline to go to its details page and view the details.

Parameter	Description
Pipeline	Pipeline name.
Created	Pipeline creation time.
Status	Pipeline status. • **Not executed**: The pipeline is not started after being created. • **Successful**: The pipeline is successfully started. • **Running**: The pipeline is being started. • **Failed**: The pipeline fails to be started. • **To be confirmed**: If a pipeline needs to be approved after startup, it waits for the approvers to determine whether to deploy the image package generated during build to the environment during image package environment deployment.
Executed	Time when the pipeline was last started.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Triggering Policy	Policy for triggering pipeline execution. • **Manual**: Manually start the pipeline. • **Automatic**: Set the code source, corresponding namespace, repository name, and branch. When code is submitted to the corresponding branch of the source code repository, the pipeline is automatically triggered. • **Scheduled**: The pipeline is automatically executed based on the preset schedule.
Description	Pipeline description.

----End

10.6.9 Viewing Pipeline Running Records

This section describes how to view pipeline running records.

Prerequisites

A pipeline has been created. For details, see **Creating a Pipeline**.

Viewing Pipeline Running Records

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

Step 3 Click a pipeline to go to its details page.

Step 4 Click **View Running Records** to view the pipeline running status.

----End

10.6.10 Deleting a Pipeline

This section describes how to delete a pipeline that is no longer used.

After a pipeline is deleted, the existing scheduled build jobs or component deployment may be unavailable.

Prerequisites

The pipeline has been created and is not in the **Running** state. For details, see **Creating a Pipeline**.

Deleting a Pipeline

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Pipeline**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Delete the pipeline in either of the following methods:

- Directly delete: Select the pipeline and choose **More** > **Delete**.
- View details and then delete: Click the pipeline. On the **Pipeline Details** page displayed, view the details and then click **Delete**.

Step 4 Click **OK**.

----End

10.7 Managing Repository Authorization

- **Creating Repository Authorization**: Create repository authorization so that ServiceStage build projects and application components can use the authorization information to access the software repository.
- **Re-authorizing a Repository**: Change the authorization mode of the repository authorization and re-authorize the repository.
- **Deleting Repository Authorization**: Delete the repository authorization that is no longer used. After the repository authorization is deleted, **Upgrading a Single Component** or **Upgrading Components in Batches** may fail for components created and deployed based on a container using source code. **Starting a Build Job** may also fail for source code jobs.

Creating Repository Authorization

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Repository Authorization**.

Step 3 Click **Create Authorization** and configure authorization information by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Table 10-1 Authorization information

Parameter	Description
*Name	Authorization name, which cannot be changed after being created. Enter 4 to 63 characters. Only use letters, digits, periods (.), underscores (_), and hyphens (-).

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Repository Authorization	The following supported official repository types vary depending on regions: • GitHub (**https://github.com**) Authorization mode: OAuth or private token. • Bitbucket (**https://bitbucket.org**) Authorization mode: OAuth or private Bitbucket. • GitLab (**https://gitlab.com**) Authorization mode: OAuth or private token. ServiceStage allows you to access official and private GitLab source code repositories using private tokens. – Visit the official GitLab source code repository, obtain and enter a private token as prompted, and select **Verify token (access the repository address from the public network)**. – Visit the private GitLab source code repository, enter the correct private GitLab source code repository address and private token as prompted. You do not need to select **Verify token (access the repository address from the public network)**.

Step 4 Click **Ok**.

----End

Re-authorizing a Repository

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Repository Authorization**.

Step 3 Click **Re-authorize** in the **Operation** column of the target repository authorization and reconfigure the authorization information based on the repository type by referring to the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 10-2 Authorization information

Parameter	Description
Method	Reset the authorization mode based on the authorization repository type. The following supported official repository types vary depending on regions: • GitHub (**https://github.com**) Authorization mode: OAuth or private token. • Bitbucket (**https://bitbucket.org**) Authorization mode: OAuth or private Bitbucket. • GitLab (**https://gitlab.com**) Authorization mode: OAuth or private token. ServiceStage allows you to access official and private GitLab source code repositories using private tokens. – Visit the official GitLab source code repository, obtain and enter a private token as prompted, and select **Verify token (access the repository address from the public network)**. – Visit the private GitLab source code repository, enter the correct private GitLab source code repository address and private token as prompted. You do not need to select **Verify token (access the repository address from the public network)**.

Step 4 Complete the authorization as prompted.

Step 5 Click **OK**.

----End

Deleting Repository Authorization

Step 1 Log in to ServiceStage.

Step 2 Choose **Continuous Delivery** > **Repository Authorization**.

- Batch deletion: Select the target repository authorizations and click **Delete** in the upper left corner.
- Single deletion: Click **Delete** in the **Operation** column of the target repository authorization.

Step 3 Enter **DELETE**.

Step 4 Click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

11 Microservice Engine

11.1 Cloud Service Engine Overview

Cloud Service Engine (CSE) provides service registry, service governance, and configuration management. It allows you to quickly develop microservice applications and implement high-availability O&M, and supports multiple languages, multiple runtime systems, and Spring Cloud and Apache ServiceComb Java Chassis (Java chassis) frameworks.

You can use the professional microservice engine named "Cloud Service Engine" or create an exclusive microservice engine.

- An exclusive microservice engine is physically isolated. A tenant exclusively uses an exclusive microservice engine.
- The professional microservice engine does not support multiple AZs.
- You can configure multiple AZs when creating an exclusive engine.
- After a microservice engine is created, the AZ cannot be modified. Select a suitable AZ when creating a microservice engine.
- Exclusive microservice engines cannot run across CPU architectures.

11.2 Creating a Microservice Engine

This section describes how to create a microservice engine.

Restrictions

- When a microservice engine is in use, do not disable the enterprise project. Otherwise, the engine will not be displayed in the engine list, affecting normal use.
- The specification cannot be changed once the microservice engine is created.
- The AZ of a created microservice engine cannot be changed.
- The VPC cannot be changed once the engine is created.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Prerequisites

A microservice engine runs on a VPC. Before creating a microservice engine, ensure that a VPC and subnet are available.

You have created a VPC. For details, see **Creating a VPC**.

If the engine is created using an account with the minimum permission for creating engines, for example, **cse:engine:create** in the **fine-grained permission dependencies of microservice engines**, the default VPC security group cse-engine-default-sg needs to be preset by the primary account and the rules listed in **Table 11-1** need to be added.

For details, see **Adding a Security Group Rule**.

Table 11-1 cse-engine-default-sg rules

Direction	Priority	Policy	Protocol and Port	Type	Source Address
Inbound	1	Allow	ICMPv6: All	IPv6	::/0
	1	Allow	TCP: 30100–30130	IPv6	::/0
	1	Allow	All	IPv6	cse-engine-default-sg
	1	Allow	All	IPv4	cse-engine-default-sg
	1	Allow	TCP: 30100–30130	IPv4	0.0.0.0/0
	1	Allow	ICMP: all	IPv4	0.0.0.0/0
Outbound	100	Allow	All	IPv4	0.0.0.0/0
	100	Allow	All	IPv6	::/0

Creating a Microservice Engine

Step 1 Go to the **Buy Exclusive Microservice Engine page**.

NOTE

- By default, a maximum of five exclusive microservice engines can be created for each project. To create more exclusive microservice engines, submit a service ticket to increase the quota. For details, see **Creating a Service Ticket**.
- For details about projects, see **Projects**.

Step 2 Set parameters according to the following table. Parameters marked with an asterisk (*) are mandatory.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
*Billing Mode	Billing mode. Currently, **Pay-per-use** is supported.
*Enterprise Project	Select the project where the microservice engine is located. You can search for and select the required enterprise project from the drop-down list. Enterprise projects let you manage cloud resources and users by project. An enterprise project can be used after it is created and enabled. For details, see **Enabling the Enterprise Project Function**. By default, **default** is selected. After a microservice engine is created, you can remove microservice engine resources out of the current enterprise project and into a new enterprise project. For details, see **Removing Resources from an Enterprise Project** and **Adding Resources to an Enterprise Project**.
*Specification	Select the microservice instance quota.
*Engine Type	Microservice engine type. If the engine type is cluster, the engine is deployed in cluster mode and supports host-level DR.
*Name	Name of a microservice engine. The name contains 3 to 24 characters, including letters, digits, and hyphens (-), and starts with a letter but cannot end with a hyphen. The name cannot be **default** and cannot be changed once the engine is created.
*AZ	Availability zone. Select one or three AZs for the engine based on the number of AZs in the environment. • Select one AZ to provide host-level DR. • Select three AZs to provide AZ-level DR. **NOTE** • The AZs in one region can communicate with each other over an intranet. • Multiple AZs enhance DR capabilities.
*Network	Select a VPC and subnet to provision logically isolated, configurable, and manageable virtual networks for your engine. • To use a created VPC, search for and select a VPC created under the current account from the drop-down list. • To use a new VPC, click **Create VPC** in the drop-down list. For details, see **Creating a VPC**.
Description	Click and enter the engine description.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Authentication Mode	The exclusive microservice engine with security authentication enabled provides the system management function using the role-based access control (RBAC) through the microservice engine console. • Select **Enable security authentication**: 1. Determine whether to enable **Authenticate Programming Interface**. After it is enabled, you need to add the corresponding account and password to the microservice configuration file. Otherwise, the service cannot be registered with the engine. After it is disabled, you can register the service with the engine without configuring the account and password in the microservice configuration file, which improves the efficiency. You are advised to disable this function when accessing the service in a VPC. 2. Enter and confirm the password of user **root**. Keep the password secure. • Select **Disable security authentication**: Disable security authentication. You can enable it after the instance is created.

Step 3 Click **Buy**. The page for confirming the engine information is displayed.

Step 4 Click **Submit** and wait until the engine is created.

NOTE

- It takes about 10–30 minutes to create a ServiceComb engine.
- After the microservice engine is created, its status is **Available**. For details about how to view the microservice engine status, see **Viewing Microservice Engine Information**.
- If the microservice engine fails to be created, view the failure cause on the **Operation** page and rectify the fault. Then, you can perform the following operations:
  - In the **Microservice Engine Information** area, click **Retry** to create an engine again.
  - If the retry fails, delete the microservice engine that fails to be created. For details, see **Deleting an Exclusive Microservice Engine**.

----End

11.3 Managing Microservice Engines

11.3.1 Viewing Microservice Engine Information

In the **Microservice Engine Information** area, you can view the engine information as shown in **Table 11-2**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Viewing Microservice Engine Information

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Microservice Engine Information** area, you can view the engine information as shown in **Table 11-2**.

Table 11-2 Engine details

Item	Description
Name	Engine name entered when **Creating a Microservice Engine**. Click to copy it.
Engine ID	Engine ID. Click to copy it.
Status	Engine status. • Creating • Available • Unavailable • Configuring • Deleting • Upgrading • Resizing • Creation failed • Deletion failed • Upgrade failed • Resizing failed • Frozen • Unknown
Version	Engine version.
Engine Type	Engine type selected when **Creating a Microservice Engine**.
AZ	Availability zone selected when **Creating a Microservice Engine**.

----End

11.3.2 Obtaining the Service Center Address of a Microservice Engine

This section describes how to obtain the service center address of a microservice engine. The service center address cannot be changed after the engine is created.

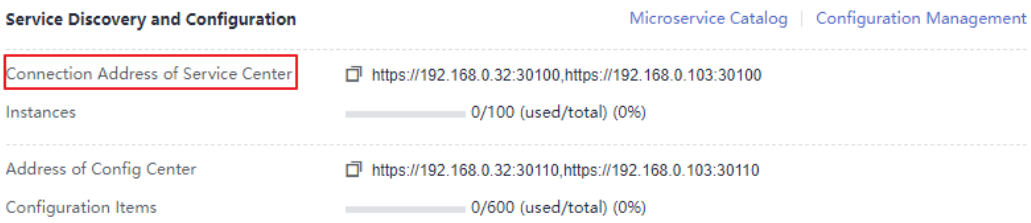
 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Obtaining the Service Center Address of a Microservice Engine

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Service Discovery and Configuration** area, view the service center address of the microservice engine.



----End

11.3.3 Obtaining the Configuration Center Address of a Microservice Engine

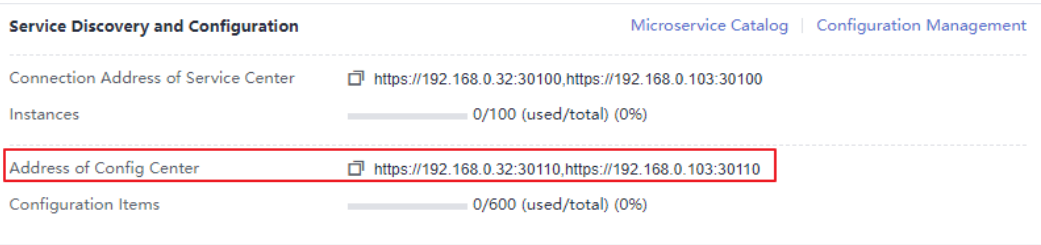
This section describes how to obtain the configuration center address of a microservice engine.

Obtaining the Configuration Center Address of a Microservice Engine

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Service Discovery and Configuration** area, view the configuration center address of the microservice engine.



NOTE

- For microservice engine 1.x, the port number of the configuration center address is 30103.
- For microservice engine 2.x, the port number of the configuration center address is 30110.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

11.3.4 Viewing the Instance Quota of a Microservice Engine

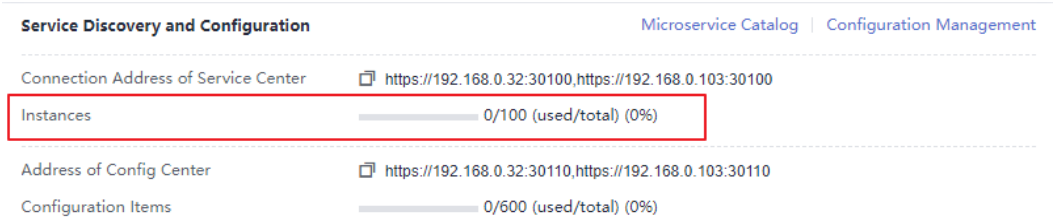
This section describes how to view the instance quota and quota usage of a microservice engine.

Viewing the Instance Quota of a Microservice Engine

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Service Discovery and Configuration** area, view the instance quota and quota usage of the microservice engine.



----End

11.3.5 Viewing the Configuration Item Quota of a Microservice Engine

This section describes how to view the configuration item quota and quota usage of a microservice engine.

NOTE

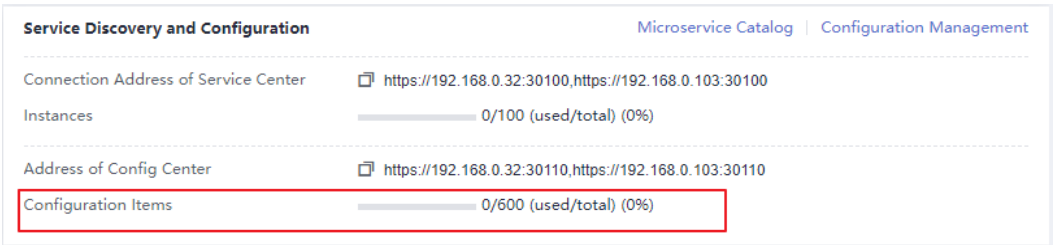
This section applies only to microservice engine 2.x.

Viewing the Configuration Item Quota of a Microservice Engine

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Service Discovery and Configuration** area, view the configuration item quota and quota usage of the microservice engine.



----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

11.3.6 Configuring Backup and Restoration of a Microservice Engine

The ServiceStage console provides the backup and restoration functions. You can back up and restore microservice engine data, including microservices, contracts, configurations, and account roles.

You can customize backup policies to periodically or manually back up microservice engines.

Restrictions

- Each exclusive microservice engine supports a maximum of 15 successful backups, including a maximum of 10 manual backups and a maximum of 5 automatic backups.
- The backup data will be stored for 10 days. Expired backup data will be deleted.
- The backup data will overwrite the current data of the microservice engine. As a result, the microservice and service instances may be messed, and dynamic configurations may be lost. Exercise caution when performing this operation.
- If security authentication is enabled, the backup data contains the account information. You are advised to disable security authentication before restoring the backup data. Otherwise, the authentication for accessing the microservice engine may fail after the restoration.

Automatic Backup

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Backup and Restoration** area, click **Automatic backup settings** and set backup parameters.

Table 11-3 Automatic backup parameters

Parameter	Description
Automatic Backup	After automatic backup is disabled, the previously set backup policy will be deleted. In this case, you need to set the backup policy again.
Backup Interval	Backup period. This parameter takes effect after **Automatic Backup** is enabled.
Trigger Time	Time at which a backup task starts. Only the hour is supported. This parameter takes effect after **Automatic Backup** is enabled.

Step 4 Click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Once the backup policy is set, the backup task is triggered within one hour after the preset time.

----End

Manual Backup

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Backup and Restoration** area, click **Create Manual Backup** and set backup parameters.

Table 11-4 Manual backup parameters

Parameter	Description
Name	Name of a backup task. The name cannot be changed after the backup task is created.
Remarks	(Optional) Description about the backup task.

Step 4 Click **OK**.

----End

Restoring Backup Data

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Backup and Restoration** area, click **Restore** in the **Operation** column of the row that contains the specified backup data.

1. Select **I have read and fully understand the risks**.
2. Click **OK**. To view the restoration status, click **Restoration History** in the **Backup and Restoration** area.

----End

Deleting Backup Data

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Backup and Restoration** area, click **Delete** in the **Operation** column of the target backup data. In the displayed dialog box, enter **DELETE** and click **OK**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

11.3.7 Managing Public Network Access for a Microservice Engine

11.3.7.1 Binding an EIP

Exclusive microservice engines that are bound with EIPs can be accessed from the public network.

Restrictions

Microservice engines with security authentication disabled do not have the authentication and authorization capabilities. Opening those engines to the public network may cause security risks and increases the system vulnerability. For example, data assets such as configurations and service information may be stolen.

Do not use this function in a production environment or a network environment with high security requirements.

Prerequisites

An EIP has been created.

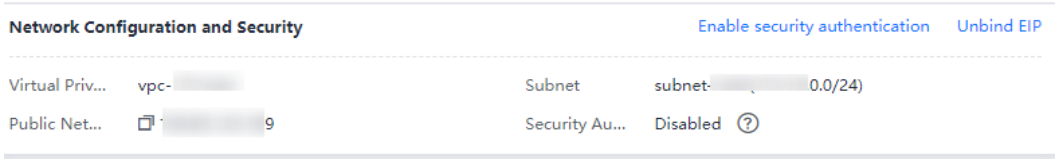
For details about how to create an EIP, see **Assigning an EIP**.

Binding an EIP

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Network Configuration and Security** area, click **Bind EIP**.



Step 4 Read the security risk prompt in the displayed dialog box and select **I understand the security risks**.

Step 5 In the **EIP** drop-down list, select the EIP to be bound. You can only select an EIP in the same enterprise project as the microservice engine.

Step 6 Click **OK**.

----End

11.3.7.2 Unbinding an EIP

If an EIP has been bound to an exclusive microservice engine, you can unbind the EIP from the engine to disable the public network access to the engine.

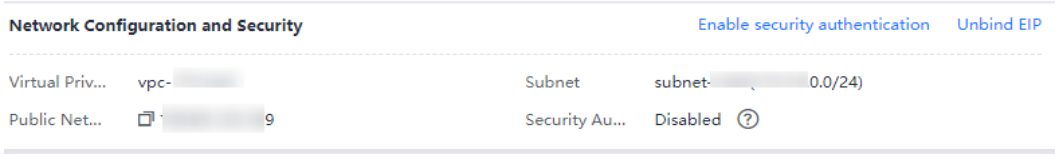
 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Unbinding an EIP

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Network Configuration and Security** area, click **Unbind EIP**.



Step 4 In the displayed dialog box, click **OK**.

----End

11.3.8 Viewing Microservice Engine Operation Logs

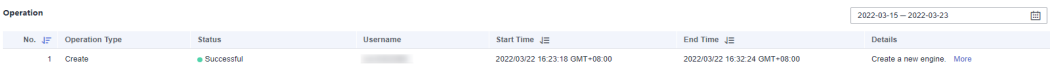
In the **Operation** area, you can view the operation logs of a microservice engine.

Viewing Microservice Engine Operation Logs

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Operation** area, view the operation logs of a microservice engine.



- Click in the upper right corner to view operation logs in a specified period.
- Click **More** in the **Details** column of a specified operation log to view details about the operation log.

----End

11.3.9 Upgrading a Microservice Engine Version

Microservice engines are created using the latest engine version. When a later version is released, you can upgrade your engine.

During upgrade, two instances are upgraded in rolling mode without service interruptions. However, one of the two access addresses may be unavailable. In this case, you need to quickly switch to the other instance. Currently, ServiceComb SDK and Mesher support instance switching. If you call the APIs of the service center and configuration center for service registry and discovery, instance switching is required.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Restrictions

- During the microservice engine upgrade, the microservice and engine are intermittently disconnected, but services of running microservices are not affected. You are advised not to upgrade, restart, or change microservices when upgrading a microservice engine.
- Only exclusive microservice engines can be upgraded. Version rollback is not supported after the upgrade.
- For details about the precautions for upgrading an exclusive microservice engine from 1.x to 2.x, see **What Do I Need to Know Before Upgrading an Exclusive Microservice Engine?**

Upgrading a Microservice Engine Version

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Microservice Engine Information** area, click **Upgrade**.

Step 4 Select **Target Version** and view the version description. Determine whether to upgrade the software to this version.

Step 5 Click **OK**.

If the upgrade fails, click **Retry** to perform the upgrade again.

----End

11.3.10 Deleting a Microservice Engine

You can delete an exclusive microservice engine if it is no longer used.

- Deleted engines cannot be restored. Exercise caution when performing this operation.
- For engine 1.x, if the cse_admin_trust agency is missing, deleting the engine will cause residual DNS, VPC, and security group resources on the tenant side. You need to delete them by yourself.

Background

You can delete exclusive microservice engines in the following states:

- Available
- Unavailable
- Creation failed
- Resizing failed
- Upgrade failed
- Unknown

Deleting a Microservice Engine

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Microservice Engine Information** area, click **Delete**. In the displayed dialog box, enter **DELETE** and click **OK**.

NOTE

If the deletion fails, click **Force Delete**.

----End

11.3.11 Managing Security Authentication for a Microservice Engine

A microservice engine may be used by multiple users. Different users must have different microservice engine access and operation permissions based on their responsibilities and permissions. If security authentication is enabled for an exclusive microservice engine, grant different access and operation permissions to users based on the roles associated with the accounts used by the users to access the microservice engine.

For details about security authentication, see **System Management**.

Currently, Java chassis and Spring Cloud support security authentication for microservices. The Java chassis version must be 2.3.5 or later, and Spring Cloud must integrate Spring Cloud Huawei 1.6.1 or later.

You can enable or disable security authentication for the exclusive microservice engine based on service requirements.

- **Enabling Security Authentication**

  If a microservice engine is available with security authentication disabled, you can enable security authentication based on service requirements.

  After security authentication is enabled and programming interface authentication is also enabled, if security authentication parameters are not configured for the microservice components connected to the engine, or the security authentication account and password configured for the microservice components are incorrect, the heartbeat of the microservice components fails and the service is forced to go offline. Perform the following steps:

  - Spring Cloud: For details, see **Connecting Spring Cloud Applications to CSE Engines**.
  - Java Chassis: For details, see **Connecting Java Chassis Applications to CSE Engines**.

- **Disabling Security Authentication**

  If a microservice engine is available with security authentication enabled, you can disable security authentication based on service requirements.

  After security authentication is disabled for a microservice component, service functions of the microservice component are not affected no matter whether security authentication parameters are configured for the microservice component.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Enabling Security Authentication

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Network Configuration and Security** area, click **Set Authentication**.

- If the engine version is earlier than 1.2.0, go to **Step 4**.
- If the engine version is 1.2.0 or later, go to **Step 5**.

Step 4 Upgrade the engine to 1.2.0 or later.

1. Click **Upgrade**.
2. Select **Target Version** and view the version description. Determine whether to upgrade the software to this version. Then, click **OK**.
3. Select the upgraded microservice engine. In the **Network Configuration and Security** area, click **Set Authentication**.

Step 5 On the **System Management** page, enable security authentication.

- To enable security authentication for the first time, click **Enable security authentication**.

  You need to create user **root** first. Enter and confirm the password of user **root**. Then, click **Create Now**.
- Enable security authentication again and enter the name and password of the account associated with the **admin** role in the engine.

Step 6 (Optional) Create a role based on service requirements. For details, see **Roles**.

Step 7 (Optional) Create an account based on service requirements. For details, see **Accounts**.

Step 8 On the **System Management** page, click **Enable security authentication** and configure the security settings.

- If you enable **Authenticate Console**, go to **Step 10**.

  After **Authenticate Console** is enabled, you need to use an account and password to log in to the CSE console. The login user can only view and configure services on which the user has permission.
- If you enable **Authenticate Programming Interface**, go to **Step 9**.

  After **Authenticate Programming Interface** is enabled, **Authenticate Console** is automatically enabled.

  After it is enabled, you need to add the corresponding account and password to the microservice configuration file. Otherwise, the service cannot be registered with the engine.

  After it is disabled, you can register the service with the engine without configuring the account and password in the microservice configuration file, which improves the efficiency. You are advised to disable this function when accessing the service in a VPC.

Step 9 Configure the SDK. For microservice components that have been deployed but not configured with security authentication parameters, configure the account name and password for security authentication and then upgrade the component. For

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

details, see **Configuring the Security Authentication Account and Password for a Microservice**.

Step 10 Click **OK**.

After the microservice engine is updated and the engine status changes from **Configuring** to **Available**, security authentication is enabled successfully.

----End

Disabling Security Authentication

After security authentication is disabled, accounts created on the engine will not be deleted.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 In the **Network Configuration and Security** area, click **Set Authentication**.

Step 4 On the **System Management** page, click **Set Authentication**.

Step 5 On the **Security Settings** page, disable **Authenticate Console**.

Step 6 Click **OK**. After the microservice engine is updated and the engine status changes from **Configuring** to **Available**, security authentication is disabled successfully.

----End

11.4 Using Microservice Engines

11.4.1 Using the Microservice Dashboard

You can view metrics related to microservices through the dashboard in real time. Based on abundant and real-time dashboard data, you can take corresponding governance actions for microservices.

Restrictions

- If a microservice application is deployed on ServiceStage, you need to configure the microservice engine during application deployment. The application automatically obtains the service registry and discovery address, configuration center address, and dashboard address. You do not need to configure the monitor address.
- If the microservice application is locally started and registered with the microservice engine, manually configure the monitor address before using the dashboard.

  For details, see **Using Dashboard**.
- When the Spring Cloud Huawei framework is used for access, the median latency, 90th latency, and 99th latency cannot be viewed on the dashboard.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Viewing Microservice Running Metrics

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Dashboard**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Dashboard** page, select an application from the drop-down list box and enter a microservice name in the search box. The operating metrics of the microservice are displayed.

Click **View Diagram** to view the description of operating metrics.

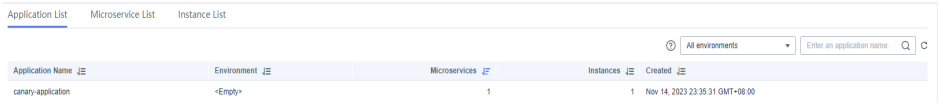
Step 6 Select a sorting order to sort the filtered microservices.

----End

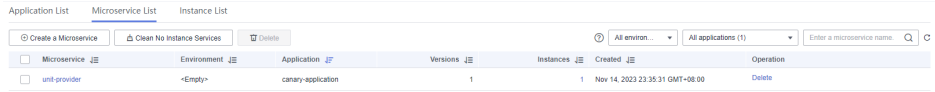
11.4.2 Managing Microservices

You can use the microservice catalog to view microservice details and search for target microservices to maintain microservices. The **Microservice Catalog** page contains the following tabs:

- **Application List**: displays all applications of the current microservice engine. You can search for the target application by application name, or filter applications by environment. For details, see **Viewing the Application List**.



- **Microservice List**: For details about the operations supported by in **Microservice List**, see the following table.



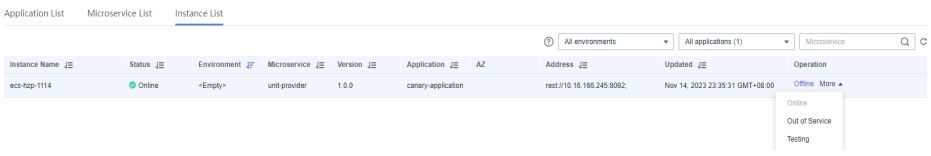
 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Viewing the Microservice List	Displays all microservices of the current microservice engine. You can search for the target microservice by microservice name, or filter microservices by environment and application.
Viewing Microservice Details	On the microservice details page, you can view the instance list, called services, calling services, dynamic configuration, and service contract. You can also perform the operations described in Table 11-5 on microservice-level configurations.
Creating a Microservice	Creates a microservice.
Cleaning Versions Without Instances	Cleans microservice versions that have no instances.
Deleting a Microservice	Deletes a microservice that is no longer used.

Table 11-5 Microservice-level configuration operations

Operation	Description
Dynamic Configuration	Create, edit, disable, and delete microservice-level configurations, and view historical versions of microservice-level configurations.
Dark Launch	In dark launch, new features are tested in a selected group of users. When the features become mature and stable, they are released to all users.

- **Instance List**: For details about the operations supported by in **Instance List**, see the following table.



 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Viewing the Instance List	Displays all instances of the current microservice engine. You can search for the target instance by microservice name, or filter instances by environment and application.
Changing the Instance Status	**Status** indicates the status of a microservice instance.

Viewing the Application List

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Application List** to view details about all applications of the current account under the engine.

You can search for the target application by application name, or filter applications by environment.

----End

Viewing the Microservice List

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Microservice List** to view all microservices of the current account under the engine.

You can search for the target microservice by microservice name, or filter microservices by environment and application.

----End

Viewing Microservice Details

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the microservice to be viewed in **Microservice List**. On the displayed page, view the instance list, called services, calling services, dynamic configuration, dark launch, and service contract.

----End

Creating a Microservice

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Choose **Microservice List** > **Create a Microservice** and set microservice parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Microservice	Microservice name, for example, **myServiceName**.
*Application	Name of the application to which the microservice belongs. Microservices are isolated by applications.
*Version	Microservice version. The default value is **1.0.0**. The microservice version is in the format of X.Y.Z or X.Y.Z.B, where X, Y, Z, and B are digits and range from 0 to 32767. The value contains 3 to 46 characters.
*Environment	Environment where the microservice is located to isolate microservice data, including the version and instance.
Detail	Microservice description.

Step 6 Click **OK**.

Once the microservice is created, it will be displayed in **Microservice List**.

----End

Cleaning Versions Without Instances

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Choose **Microservice List** > **Clean No Instance Services**. Select the microservice version without instances to be cleaned.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

You can search for the target microservice by microservice name, or filter microservices by environment and application.

Step 6 Click **OK**.

----End

Deleting a Microservice

- After a microservice is deleted, you can restore it by referring to **Restoring Backup Data**.
- If the service to be deleted has instances, delete the instances first. Otherwise, the service will be registered again.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Microservice List**.

- To delete microservices in batches, select the microservices to be deleted and click **Delete** above the microservices.
- To delete one microservice, locate the row that contains the microservice to be deleted and click **Delete** in the **Operation** column.

Step 6 In the displayed dialog box, enter **DELETE** to confirm the deletion and click **OK**.

----End

Dynamic Configuration

Configuration items are stored in plaintext. Do not include sensitive data.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Microservice List**.

Step 6 Click the target microservice.

Step 7 Choose **Dynamic Configuration**. On the **Dynamic Configuration** tab, perform the following operations.

Operation	Procedure
Creating a configuration item	See **Creating a Microservice-Level Configuration**. **Microservice-level** is selected for **Configuration Range** and **Microservices** is set to the current microservice.
Viewing historical versions	Click **View Historical Version** in the **Operation** column of the target configuration item.
Disabling a Configuration Item	1. In the **Operation** column of the target configuration item, choose **More** > **Disable**. 2. Click **OK**.
Modifying a configuration item	1. Click **Edit** in the **Operation** column corresponding to the target configuration item. 2. On the configuration details page, click **Edit**. 3. On the **Configuration Details** tab, enter the new configuration. 4. Click **Save**.
Deleting a configuration item	1. In the **Operation** column of the target configuration item, choose **More** > **Delete**. 2. Click **OK**.

----End

Dark Launch

In dark launch, new features are tested in a selected group of users. When the features become mature and stable, they are released to all users. This ensures the smooth feature rollout.

- For microservices developed based on the ServiceComb Java Chassis framework, add dependency **darklaunch** or **handler-router** to POM and add **servicecomb.router.type=router** to the configuration file.
- For microservices developed based on the Spring Cloud Huawei framework, add dependency **spring-cloud-starter-huawei-router** to POM.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 In the microservice list, click a microservice. On the displayed page, choose **Dark Launch**.

Step 6 Click **Add Launch Rule**.

- To add a launch rule by **Weight**:
  a. Click **Weight**.
  b. Set the following parameters.

Item	Description
Rule Name	Name of the rule.
Scope	▪ Microservice version to which the rule applies. ▪ Select **Do you want to add a customized version?** and add a new version as prompted.
Rule Configuration	Traffic allocation rate for the selected version. Traffic is evenly allocated to the selected service versions based on the configured value.

  c. Click **OK** to complete the weight rule configuration and dark launch.
- To add a launch rule by **Customization**:
  a. Click **Custom**.
  b. Set the following parameters.

Item	Description
Rule Name	Name of the rule.
Scope	▪ Microservice version to which the rule applies. ▪ Select **Do you want to add a customized version?** and add a new version as prompted.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Item	Description
Rule Configuration	Configure the matching rule. When **darklaunch** is used to implement dark launch, this parameter configures **policyCondition**. When **handler-router** is used to implement dark launch, it configures **Headers**. ▪ Parameter Name Set this parameter based on the parameter name of contract or the customized key of the header. ▪ Rules By selecting the matching character and the value corresponding to the key of contract or the key of the header, requests that meet the rules are allocated to the microservice version. **NOTE** ○ If **~** is selected from the drop-down list next to **Rules**, the asterisk (*) and question mark (?) in the **Rules** value can be used for fuzzy matching. The asterisk (*) indicates a character of any length, and the question mark (?) indicates one character. For example, if the rule value of **Name** is set to ***1000**, all **Name** fields ending with 1000 can be matched. ○ If **~** is not selected from the drop-down list next to **Rules**, the asterisk (*) and question mark (?) in the **Rules** value cannot be used for fuzzy matching.

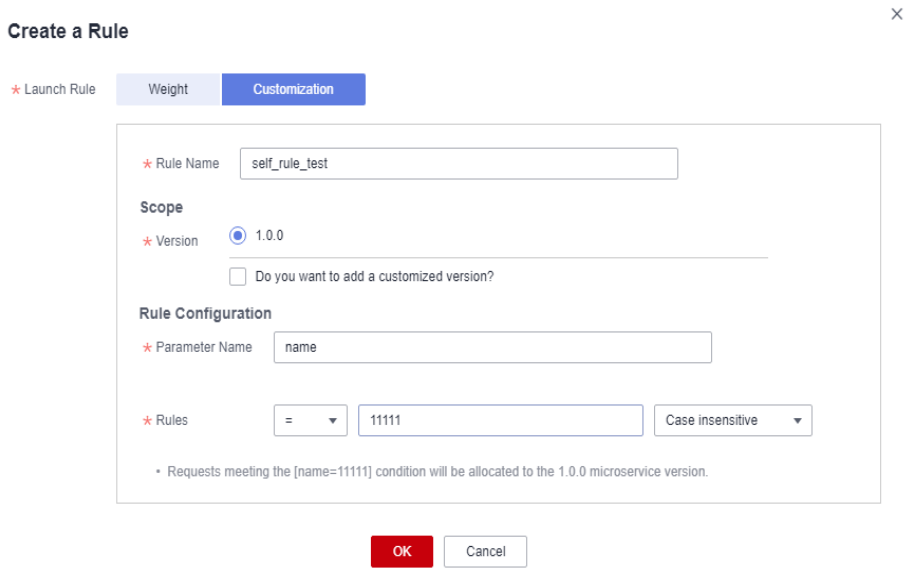
c. Click **OK** to complete the customization rule configuration and dark launch.

----End

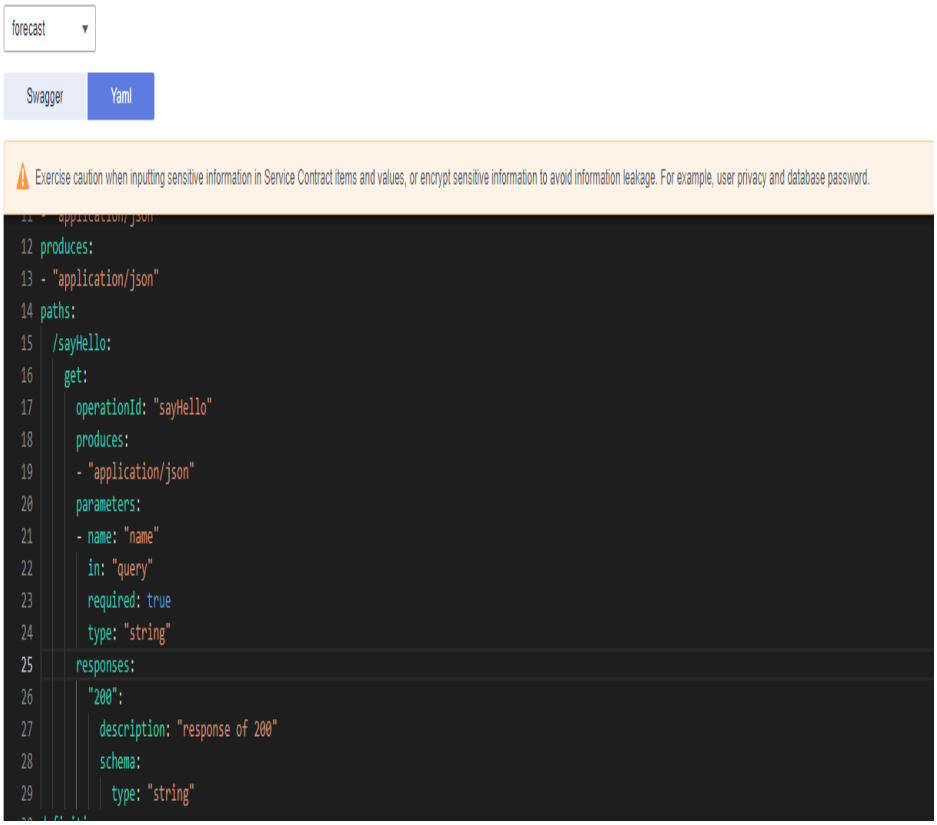
Examples of delivering dark launch rules:

- For microservices developed based on the ServiceComb Java Chassis framework, rules are delivered based on dependency **darklaunch** on the microservice engine page. You can add dark launch rules in customized mode.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

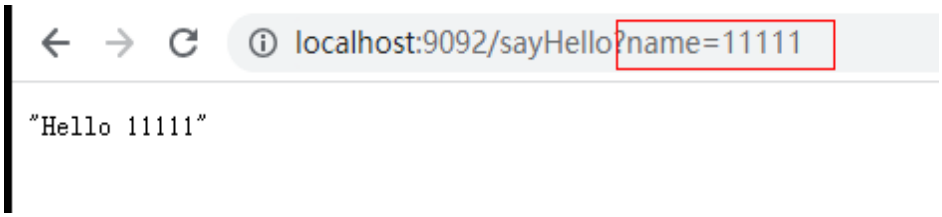


This key must exist in the contract. It is possible that the server API is **String paramA**, but **paramB** is actually generated after the annotation is added. Therefore, **paramB** should be set here.



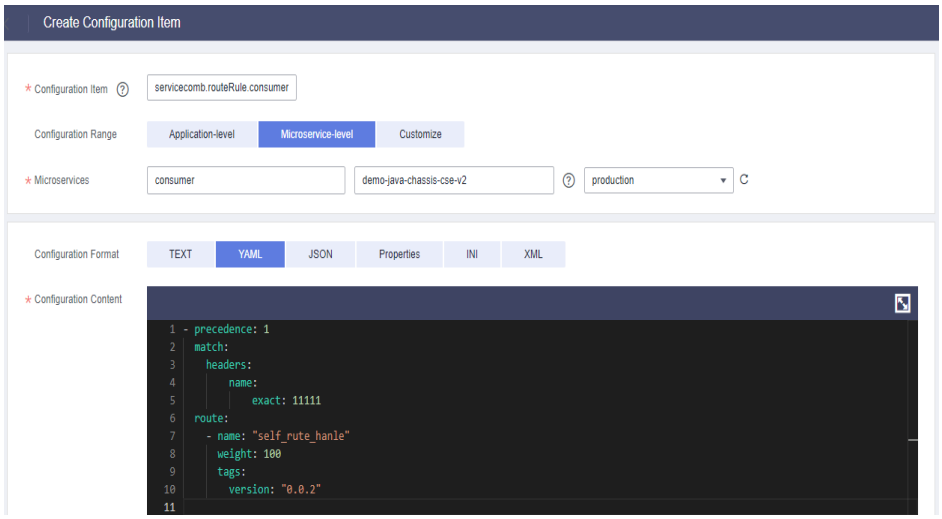
By selecting the matching character and the value corresponding to the key of contract, requests that meet the rules are allocated to the microservice version.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.



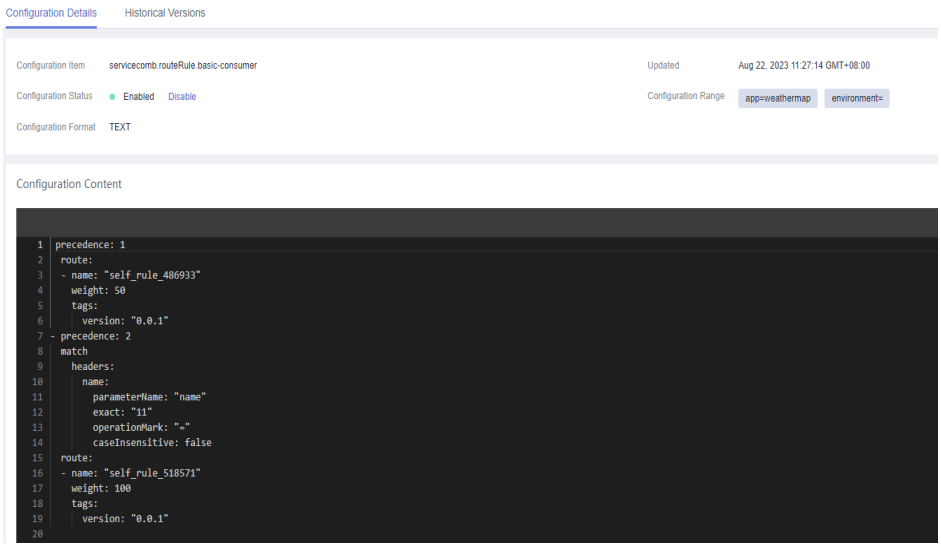
A delivered rule is as follows. The configuration item is **cse.darklaunch.policy.*${serviceName}***.

- For microservices developed based on the ServiceComb Java Chassis framework, dark launch rules that depend on **handler-router** need to be manually delivered in the configuration center. The configuration item is **servicecomb.routeRule.*${serviceName}***. The content is as follows:



- For microservices developed based on the Spring Cloud Huawei framework, dark launch rules delivered on the microservice engine page are as follows:

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.



Viewing the Instance List

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Instance List** to view all instances of the engine.

You can search for the target instance by microservice name, or filter instances by environment and application.

----End

Changing the Instance Status

Status indicates the status of a microservice instance.

The status of microservice instances synchronized by binding microservice engines cannot be changed during component creation and deployment by referring to **Creating and Deploying a Component**.

The following table describes the microservice instance statuses.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Status	Description
Online	The instance is running and can provide services.
Offline	Before the instance process ends, the instance is marked as not providing services externally.
Out of Service	The instance has been registered with the microservice engine and does not provide services.
Testing	The instance is in the internal joint commissioning state and does not provide services.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Catalog**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Instance List**, select the target instance, and change the instance status.

- Offline

  In the **Operation** column, click **Offline**.
- Online

  In the **Operation** column, choose **More** > **Online**.
- Out of Service

  In the **Operation** column, choose **More** > **Out of Service**.
- Testing

  In the **Operation** column, choose **More** > **Testing**.

----End

11.4.3 Microservice Governance

If an application is developed using the microservice framework, the microservice is automatically registered with the corresponding microservice engine after the

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

application is managed and started. After the microservice is deployed, you can go to the microservice engine console and perform service governance by referring to this section.

NOTE

This function is supported by microservice engine 1.x and 2.4.0 and later versions.

Prerequisites

- You can create a microservice in **Microservice List** from **Service Catalog** and start the microservice. After the microservice starts, the service instance is registered under the corresponding service based on configurations in the **.yaml** file.
- If the microservice is not created in advance or has been deleted, the microservice is automatically created when the service instance is registered.
- After a microservice is created, register the service instance before performing the corresponding operation.

Governance Policies

You can configure the following policies: Load Balancing, Rate Limiting, Fault Tolerance, Service Degradation, Circuit Breaker, Fault Injection, and Blacklist and Whitelist. For details, see the following table.

Name	Description
Load Balancing	• Application scenario Generally, multiple instances are deployed for a microservice. Load balancing controls the policy for a microservice consumer to access multiple instances of a microservice provider to balance traffic. It includes polling, random, response time weigh, and session stickiness. • For details about the configuration example of the governance policy and how to add dependencies to POM, see **Load Balancing**.
Rate Limiting	• Application scenario This policy controls the number of requests for accessing microservices to prevent the system from being damaged due to traffic impact. • For details about the configuration example of the governance policy and how to add dependencies to the POM, see **Rate Limiting**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Name	Description
Service Degradation	• Application scenario When a microservice invokes other microservices, the default value is forcibly returned or an exception is thrown instead of sending the request to the target microservice. In this way, the access to the target microservice is shielded and the pressure on the target microservice is reduced. • For details about the configuration example of the governance policy and how to add dependencies to the POM, see **Service Degradation**.
Fault Tolerance	• Application scenario If an exception occurs when a microservice consumer accesses a provider, for example, the instance network is disconnected, the request needs to be forwarded to another available instance. Fault tolerance is often referred to as retry. • For details about the configuration example of the governance policy and how to add dependencies to POM, see **Fault Tolerance**.
Circuit Breaker	• Application scenario If an exception occurs when a microservice consumer accesses a provider, for example, the instance network is disconnected or the request times out, and the exception accumulates to a certain extent, the consumer needs to stop accessing the provider and return an exception or a default value to prevent the avalanche effect. Automatic circuit breaker is supported, which determines a circuit breaker according to the error rate. • For details about the configuration example and how to add dependencies to the POM, see **Circuit Breaker**.
Fault Injection	This policy applies only to microservices accessed through Java chassis. • Application scenario Fault injection can simulate an invoking failure, which is mainly used for function verification and fault scenario demonstration. • Governance of microservices connected to the Java Chassis development framework. For details about the configuration example of the governance policy and how to add dependencies to POM, see **Fault Injection**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Name	Description
Blacklist/ Whitelist	This policy applies only to microservices accessed through Java chassis. • Application scenario Based on the public key authentication mechanism, microservice engines provide the blacklist and whitelist functions. The blacklist and whitelist can be used to control which services can be accessed by microservices. • Governance of microservices accessed through Java chassis The blacklist and whitelist take effect only after public key authentication is enabled. For details, see **Configuring Public Key Authentication**. • This policy applies only to microservices accessed through Java chassis.

Configuring Load Balancing

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Governance**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the microservice to be governed.

Step 6 Choose **Load Balancing**.

Step 7 Click **New**. Select the microservices to be governed and select a proper load balancing policy. For details, see the following table.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 11-1 Configuring load balancing (for microservices accessed through Spring Cloud)

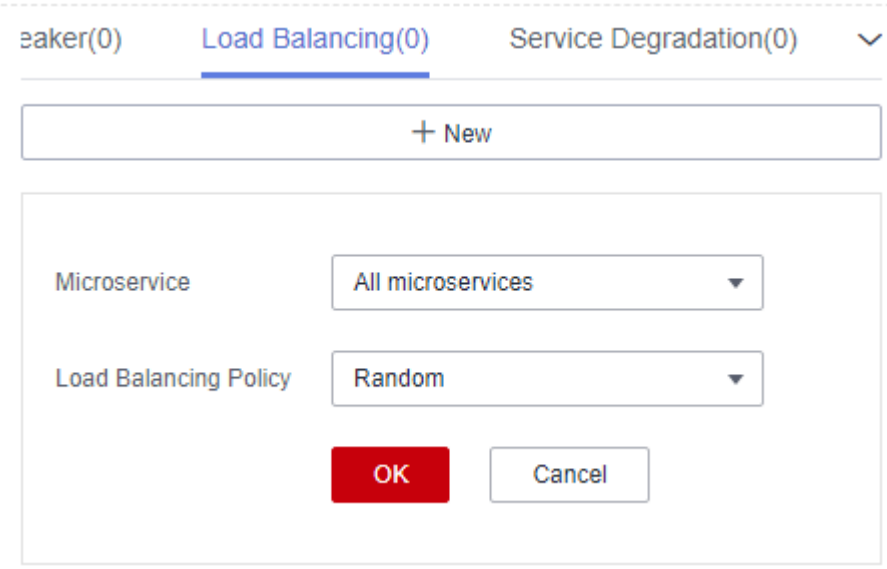
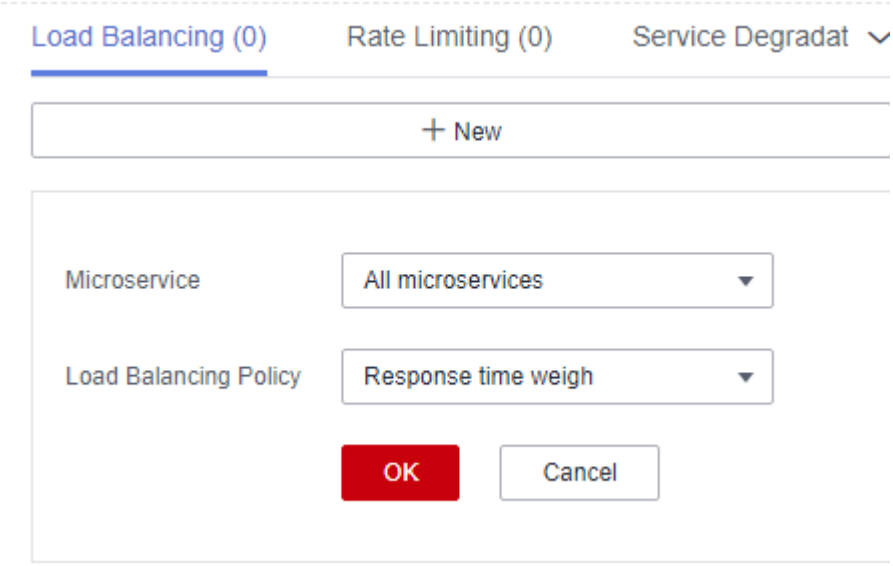


Figure 11-2 Configuring load balancing (for microservices accessed through Java chassis)



Policy	Description
Round robin	Supports routes according to the location information about service instances.
Random	Provides random routes for service instances.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Policy	Description
Response time weight	This configuration applies to microservices accessed through Java chassis. Provides weight routes with the minimum active number (latency) and supports service instances with slow service processing in receiving a small number of requests to prevent the system from stopping response. This load balancing policy is suitable for applications with low and stable service requests.
Session stickiness	This configuration applies to microservices accessed through Java chassis. Provides a mechanism on the load balancer. In the specified session stickiness duration, this mechanism allocates the access requests related to the same user to the same instance. • **Stickiness Duration**: time limit for keeping a session. The value ranges from 0 to 86400, in seconds. • **Failures**: number of access failures. The value ranges from 0 to 10. If the upper limit of failures or the session stickiness duration exceeds the specified values, the microservice stops accessing this instance.

Step 8 Click **OK**.

----End

Configuring Rate Limiting

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Governance**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the microservice to be governed.

Step 6 Click **Rate Limiting**.

Step 7 Click **New**. The following table describes configuration items of rate limiting.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 11-3 Configuring rate limiting (for microservices accessed through Spring Cloud)

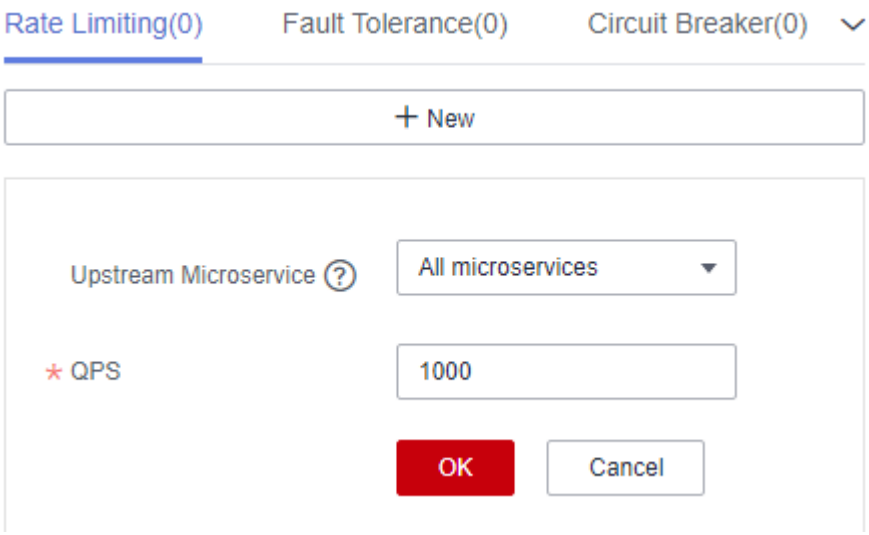
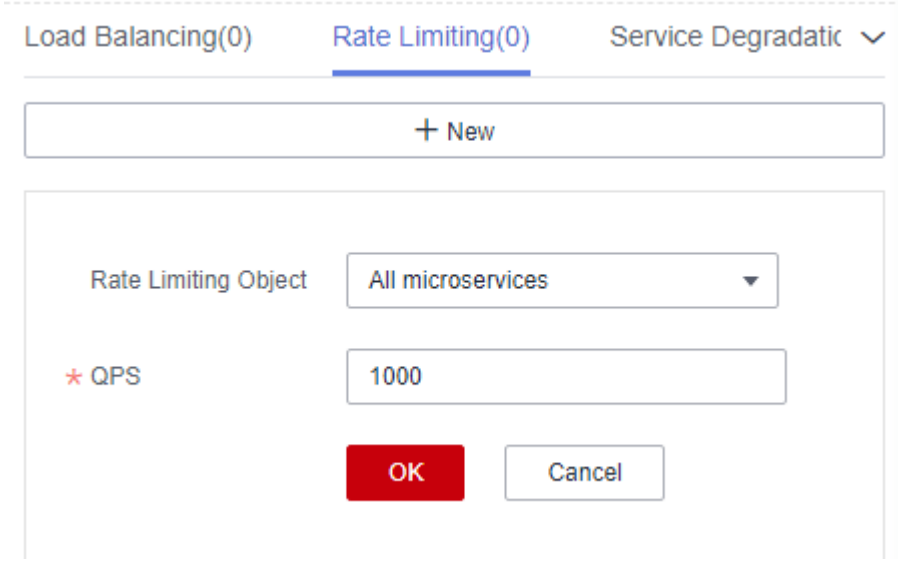


Figure 11-4 Configuring rate limiting (for microservices accessed through Java chassis)



Configuration Item	Description	Value Range
Rate Limiting Object	This configuration applies to microservices accessed through Java chassis. Other microservices that access the microservice.	Select an item from the drop-down list next to **Rate Limiting Object**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuration Item	Description	Value Range
Upstream Microservice	This configuration applies to microservices accessed through Spring Cloud. Configure rate limiting for the upstream microservice to invoke the service.	Select an item from the drop-down list next to **Upstream Microservice**.
QPS	Requests generated per second. When the number of requests sent by the rate limiting object to the current service instance exceeds the specified value, the current service instance no longer accepts requests from the rate limiting object.	Enter an integer ranging from 1 to 99999.

NOTE

If a microservice has three instances, the rate limiting of each instance is set to 2700 QPS, then the total QPS is 8100, and rate limiting is triggered only when the QPS exceeds 8100.

Step 8 Click **OK**.

----End

Configuring Service Degradation

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Governance**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the microservice to be governed.

Step 6 Click **Service Degradation**.

Step 7 Click **New** and select a proper policy. The following table describes the configuration items of service degradation.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 11-5 Configuring service degradation (for microservices accessed through Spring Cloud)

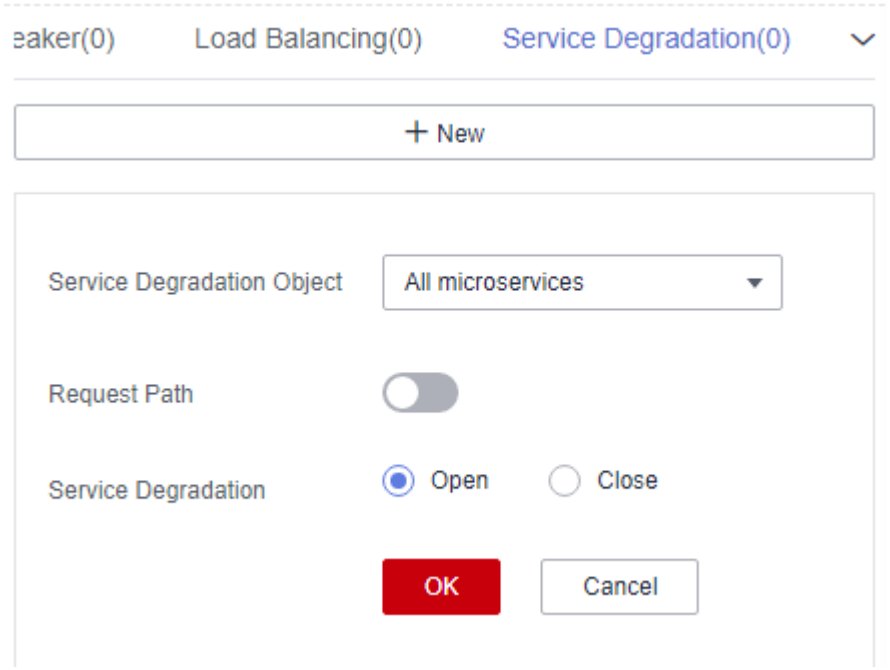
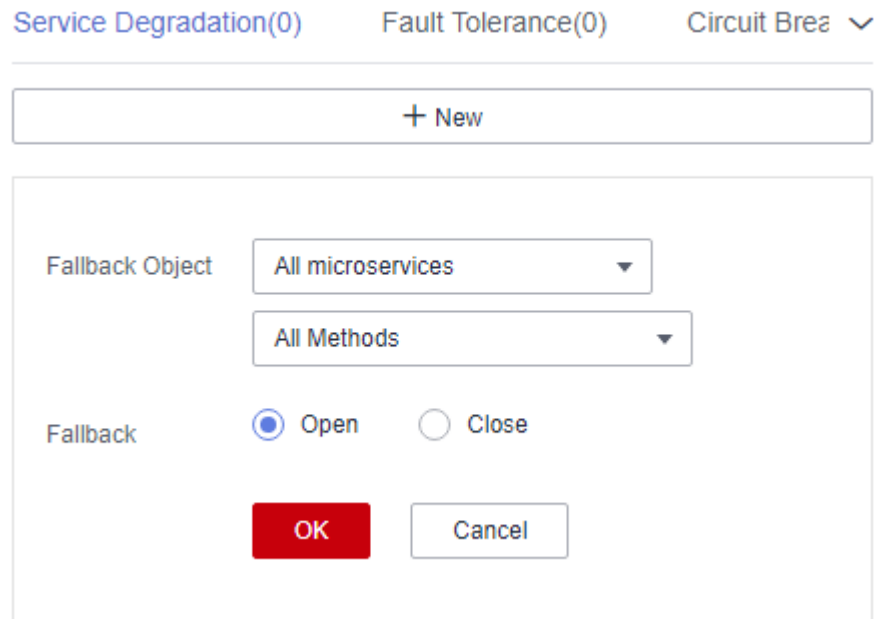


Figure 11-6 Configuring service degradation (for microservices accessed through Java chassis)



Configuration Item	Description
Fallback Object	Microservice to be degraded.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuration Item	Description
Request Path	This configuration applies to microservices accessed through Spring Cloud. Click and set **Method**, **Path**, and **Headers** to specify the request path.
Fallback	• Open • Close

Step 8 Click **OK**.

----End

Configuring Fault Tolerance

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Governance**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the microservice to be governed.

Step 6 Click **Fault Tolerance**.

Step 7 Click **New** and select a proper policy. The following table describes the configuration items of fault tolerance.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 11-7 Configuring fault tolerance (for microservices accessed through Spring Cloud)

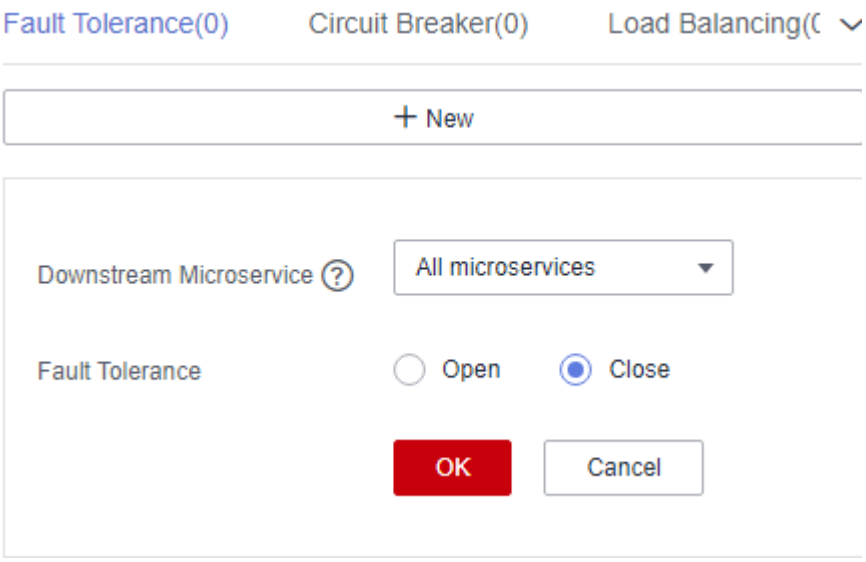
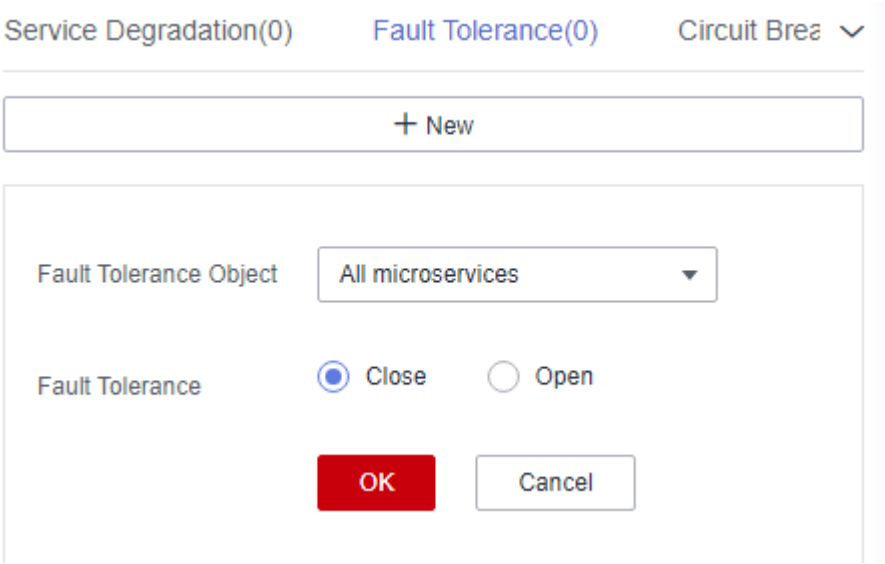


Figure 11-8 Configuring fault tolerance (for microservices accessed through Java chassis)



Configuration Item	Description
Downstream Microservice	This configuration applies to microservices accessed through Spring Cloud. Configure fault tolerance for the microservice to invoke the downstream microservice. You can select a value from the drop-down list.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuration Item	Description
Fault Tolerance Object	This configuration applies to microservices accessed through Java chassis. Microservice or method that the application relies on.
Fault Tolerance	**Open**: The system processes a request sent to the fault tolerance object based on the selected fault tolerance policy when the request encounters an error. **Close**: The system waits until the timeout interval expires and then returns the failure result even though the service request fails to be implemented.
FT Policy	This parameter is mandatory when **Fault Tolerance** is set to **Open**. For microservices accessed through Spring Cloud, set the following parameters: • Number of attempts to the same microservice instance • Number of attempts to the new microservice instance For microservices accessed through Java chassis, set the following parameters: • Failover The system attempts to reestablish connections on different servers. • Failfast The system does not attempt to reestablish a connection. After a request fails, a failure result is returned immediately. • Failback The system attempts to reestablish connections on the same server. • custom – Number of attempts to reestablish connections on the same server – Number of attempts to reestablish connections on new servers

Step 8 Click **OK**.

----End

Configuring Circuit Breaker

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Choose **Microservice Governance**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

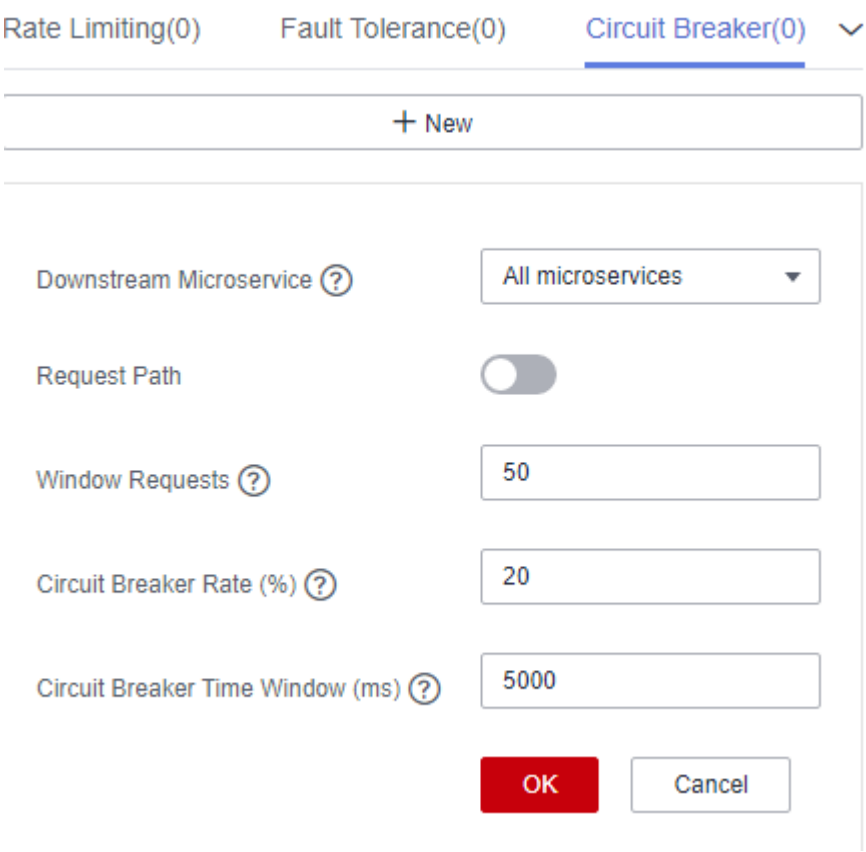
- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the microservice to be governed.

Step 6 Click **Circuit Breaker**.

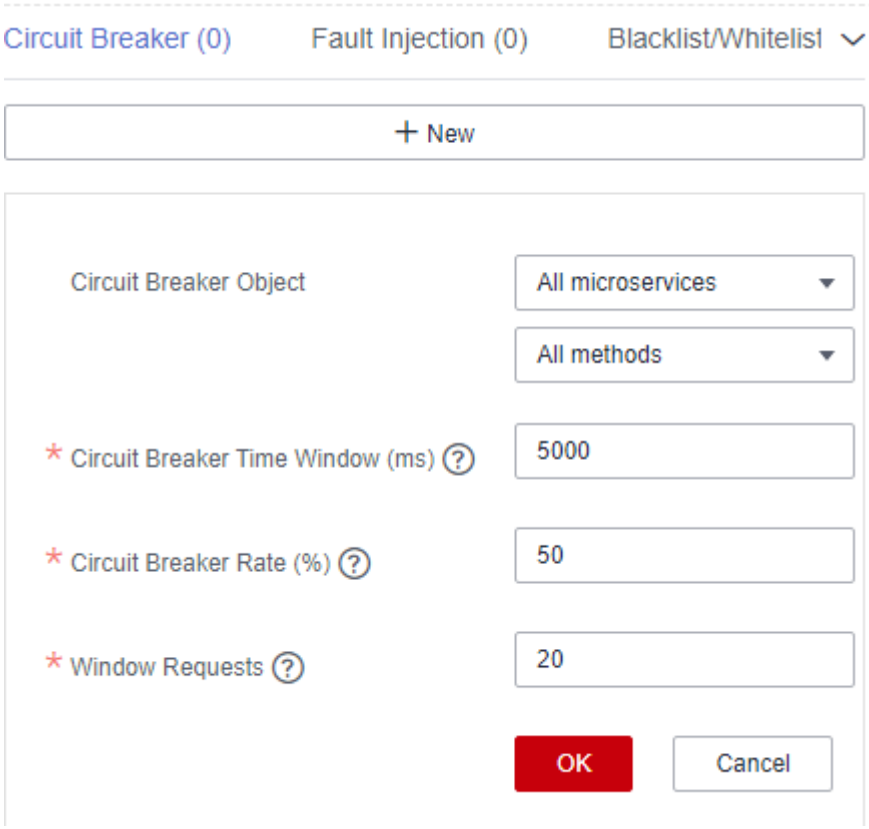
Step 7 Click **New** and select a proper policy. The following table describes the configuration items of circuit breaker.

Figure 11-9 Configuring circuit breaker (for microservices accessed through Spring Cloud)



 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 11-10 Configuring circuit breaker (for microservices accessed through Java chassis)



Configuration Item	Description
Downstream Microservice	This configuration applies to microservices accessed through Spring Cloud. Configure circuit breaker for the microservice to invoke the downstream microservice.
Circuit Breaker Object	This configuration applies to microservices accessed through Java chassis. Microservice or method invoked by the application.
Request Path	This configuration applies to microservices accessed through Spring Cloud. Click and set **Method**, **Path**, and **Headers** to specify the request path.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuration Item	Description
Triggering Condition	• **Circuit Breaker Time Window**: circuit breaker duration. The system does not respond to requests within this time window. • **Request Failure Rate**: failure rate of window requests. • **Window Requests**: number of requests received by the window. Circuit breaker is triggered only when **Request Failure Rate** and **Window Requests** both reach their thresholds.

Step 8 Click **OK**.

----End

Configuring Fault Injection

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Governance**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

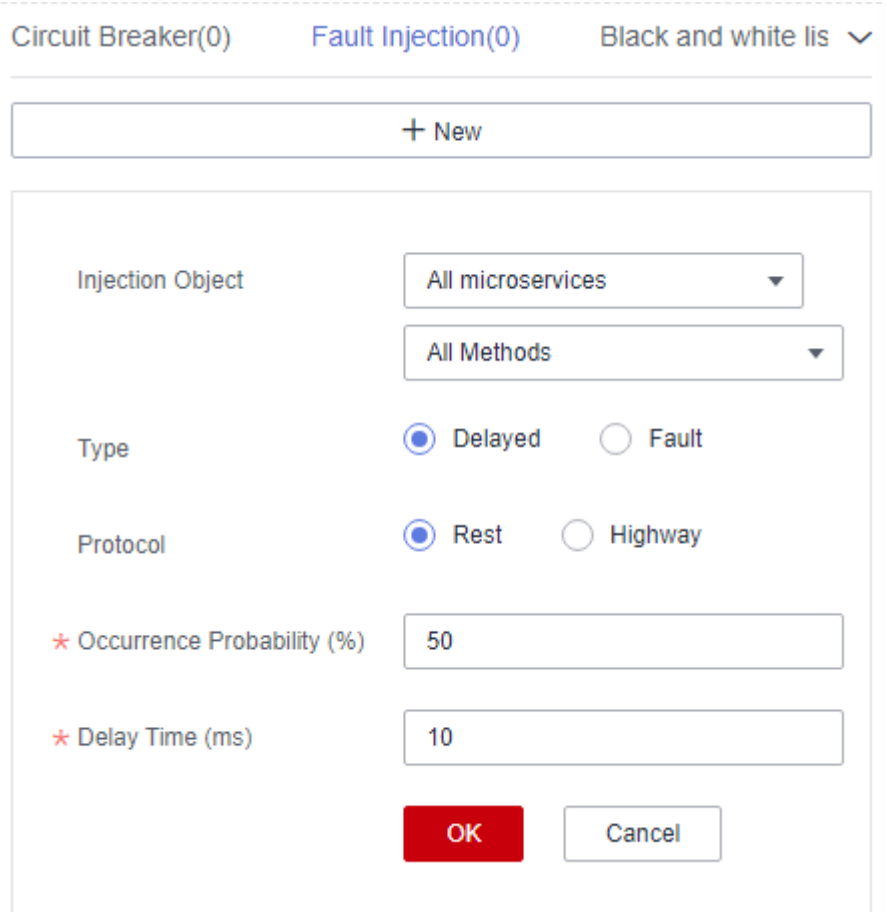
Step 5 Click the microservice to be governed.

Step 6 Click **Fault Injection**.

Step 7 Click **New** and select a proper policy. The following table describes the configuration items of fault injection.

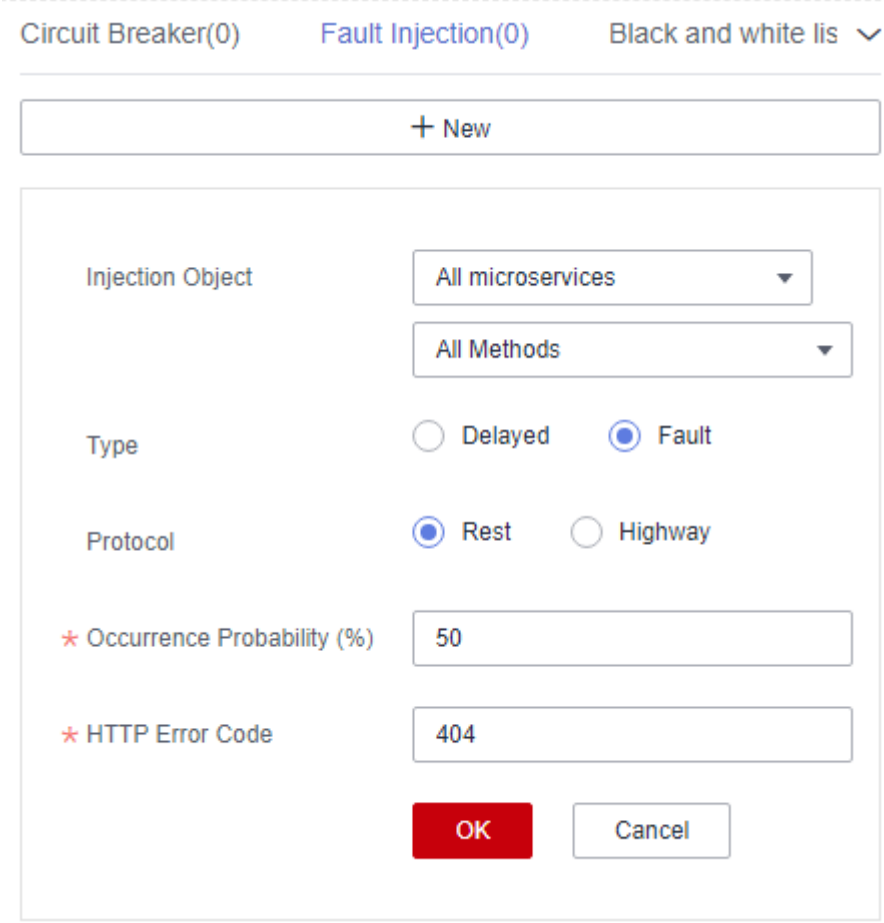
 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 11-11 Configuring fault injection (delayed)



 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 11-12 Configuring fault injection (fault)



Configuration Item	Description
Injection Object	Microservices for which fault injection is required. You can specify a method for this configuration item.
Type	Type of the fault injected to the microservice. • Delayed • Fault
Protocol	Protocol for accessing the microservice when latency or fault occurs. • Rest • Highway
Occurrence Probability	Probability of latency or fault occurrence.
Delay Time	Duration of the latency during microservice access. This parameter is required when **Type** is set to **Delayed**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Configuration Item	Description
HTTP Error Code	HTTP error code during microservice access. This parameter is required when **Type** is set to **Fault**. This error code is an HTTP error code.

Step 8 Click **OK**.

----End

Configuring Blacklist and Whitelist

Based on the public key authentication mechanism, microservice engines provide the blacklist and whitelist functions. The blacklist and whitelist can be used to control which services can be accessed by microservices.

The blacklist and whitelist take effect only after public key authentication is enabled. For details, see **Configuring Public Key Authentication**.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Microservice Governance**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

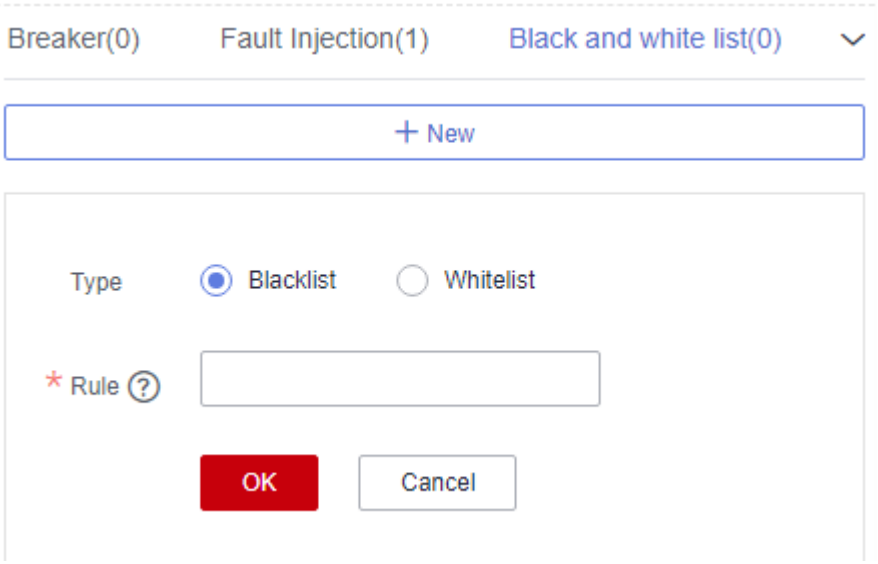
Step 5 Click the microservice to be governed.

Step 6 Click **Black and white list**.

Step 7 Click **New** to add a blacklist or whitelist for the application. The following table describes configuration items of blacklist and whitelist.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Figure 11-13 Configuring blacklist and whitelist



Configuration Item	Description
Type	• **Blacklist**: Microservices that match the matching rule are not allowed to access the current service. • **Whitelist**: Microservices that match the matching rule are allowed to access the current service.
Rule	Use a regular expression. For example, if **Rule** is set to **data***, services whose names start with **data** in the blacklist are not allowed to access the current service, or services whose names start with **data** in the whitelist are allowed to access the current service.

Step 8 Click **OK**.

----End

Configuring Public Key Authentication

Public key authentication is a simple and efficient authentication mechanism between microservices provided by CSE. Its security is based on the reliable interaction between microservices and the service center. That is, the authentication mechanism must be enabled between microservices and the service center. The procedure is as follows:

1. When the microservice starts, a key pair is generated and the public key is registered with the service center.
2. Before accessing the provider, the consumer uses its own private key to sign a message.
3. The provider obtains the public key of the consumer from the service center and verifies the signed message.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

To enable public key authentication, perform the following steps:

1. Enable public key authentication for both the consumer and provider.
```
servicecomb:
  handler:
    chain:
      Consumer:
        default: auth-consumer
      Provider:
        default: auth-provider
```
2. Add the following dependency to the **pom.xml** file:
```
<dependency>
    <groupId>org.apache.servicecomb</groupId>
    <artifactId>handler-publickey-auth</artifactId>
  </dependency>
```

11.4.4 Configuration Management (Applicable to Engine 2.x)

Microservice engines define a configuration mechanism that is irrelevant to development frameworks. A configuration item consists of a key, label, and value. The label is used to identify whether a configuration item belongs to global configuration or microservice configuration. The label can also indicate the value type.

You can refer to the following table to select the operations to be performed.

Operation	Description
Creating an Application-Level Configuration	Associates the new configuration with an application, and adds the application name and environment label.
Creating a Microservice-Level Configuration	Associates the new configuration with a microservice, and adds the microservice name, application name, and environment.
Creating a Customized Configuration Item	If application-level and microservice-level configurations cannot meet service requirements, you can customize configuration files.
Importing Configurations	Imports the local configuration file.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Exporting Configurations	Exports the selected configuration file to the local host.
Comparing Configuration Versions	Compares differences between historical versions.
Rolling Back a Version	Rolls back to the selected historical version.
Viewing Historical Versions	Displays configurations of different historical versions.
Editing a Configuration Item	Edits a configuration item.
Disabling a Configuration Item	Disables a configuration item.
Deleting a Configuration Item	Deletes a configuration item.

Restrictions

- Configuration items are stored in plaintext. Do not include sensitive data.
- When the configuration item quota specified by the engine specifications is about to be used up, the engine allows new configuration items that exceed the remaining quota to be created to ensure capacity availability. Expand the capacity of the engine as soon as possible to prevent configuration creation failures.
- When editing or deleting a configuration item used by a microservice, the microservice may fail to read the configuration or read an incorrect configuration, causing service exceptions. Therefore, back up the configuration before modifying or deleting configuration items.

Creating an Application-Level Configuration

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Create Configuration Item** and set parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Configuration Item	Enter a configuration item. The configuration item is the global ID of the configuration. In the coding phase, the configuration item is used to index and operate the configuration. You are advised to use the Java package naming rule (for example, **cse.service.registry.address**) to ensure the readability and uniqueness of the configuration. Configuration items starting with **servicecomb.matchGroup.** cannot be created during application-level configuration creation. Such configuration items conflict with the configuration generated during service scenario governance creation, so the service scenario cannot be displayed.
Configuration Range	Select **Application-level**.
*Application	1. Select or enter an application name. 2. Select an environment.
Configuration Format	Select a configuration format. Common configuration formats such as TEXT, YAML, JSON, Properties, INI and XML can be edited online. The default value is **TEXT**.
*Configuration Content	Enter the configuration content.
Enable Configuration	Determine whether to enable the configuration item. • **Enable now**: The configuration item takes effect immediately once being created. • **Not Enabled**: The configuration item does not take effect.

Step 6 Click **Create Now** to enable the configuration item.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Creating a Microservice-Level Configuration

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Create Configuration Item** and set parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Configuration Item	Enter a configuration item. The configuration item is the global ID of the configuration. In the coding phase, the configuration item is used to index and operate the configuration. You are advised to use the Java package naming rule (for example, **cse.service.registry.address**) to ensure the readability and uniqueness of the configuration.
Configuration Range	Select **Microservice-level**.
*Microservice	1. Select or enter a microservice name. 2. Select or enter an application name. 3. Select an environment.
Configuration Format	Select a configuration format. Common configuration formats such as TEXT, YAML, JSON, Properties, INI and XML can be edited online. The default value is **TEXT**.
*Configuration Content	Enter the configuration content.
Enable Configuration	Determine whether to enable the configuration item. • **Enable now**: The configuration item takes effect immediately once being created. • **Not Enabled**: The configuration item does not take effect.

Step 6 Click **Create Now** to enable the configuration item.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Creating a Customized Configuration Item

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Create Configuration Item** and set parameters by referring to the following table. Parameters marked with an asterisk (*) are mandatory.

Parameter	Description
*Configuration Item	Enter a configuration item. The configuration item is the global ID of the configuration. In the coding phase, the configuration item is used to index and operate the configuration. You are advised to use the Java package naming rule (for example, **cse.service.registry.address**) to ensure the readability and uniqueness of the configuration.
Configuration Range	Select **Customize**.
Labels	If application-level and microservice-level configurations cannot meet service requirements, you can use labels to customize configurations.
Configuration Format	Select a configuration format. Common configuration formats such as TEXT, YAML, JSON, Properties, INI and XML can be edited online. The default value is **TEXT**.
*Configuration Content	Enter the configuration content.
Enable Configuration	Determine whether to enable the configuration item. • **Enable now**: The configuration item takes effect immediately once being created. • **Not Enabled**: The configuration item does not take effect.

Step 6 Click **Create Now** to enable the configuration item.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Importing Configurations

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Import** in the upper right corner and set parameters by referring to the following table.

Parameter	Description
Import to a specific environment	• Disabled: The imported configuration does not change the environment label. • Enabled: Importing the configuration to a specific environment will change the environment label.
Same Configuration	• **Terminate**: If a configuration is the same as that in the system, the import terminates. • **Skip**: During import, if a configuration is the same as that in the system, the configuration is skipped and other configurations are imported. • **Overwrite**: During import, if a configuration is the same as that in the system, the value of the configuration will be replaced.
Configuration File	Click **Import** and select the target file. The file size cannot exceed 2 MB.

Step 6 Click **Close**.

----End

Exporting Configurations

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Select the configuration items to be exported and click **Export**. In the displayed dialog box, click **Export**. Alternatively, click **Export All** in the upper right corner to export all configurations.

----End

Comparing Configuration Versions

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the configuration item to be compared.

Step 6 Click **View Historical Version**.

Step 7 In **Historical Versions** on the left, select the historical version to be viewed.

In the **Configuration file** on the right, you can view the differences between the current and historical versions.

NOTE

A maximum of 100 historical versions can be displayed.

----End

Rolling Back a Version

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the target configuration item.

Step 6 Click **View Historical Version**.

Step 7 In **Historical Versions** on the left, select the target historical version.

Step 8 In **Configuration file** on the right, click **Roll Back to the Selected Version**.

----End

Viewing Historical Versions

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **View Historical Version** in the **Operation** column of a configuration item. On the **Historical Versions** page that is displayed, you can view the historical versions of the configuration item. On this page, you can compare the configuration version with the rollback version.

----End

Editing a Configuration Item

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Edit** in the **Operation** column of the target configuration item. You can also click the target configuration item and then click **Edit** on the displayed configuration details page.

Step 6 Enter the configuration information in the **Configuration Content** text box and click **Save**.

----End

Disabling a Configuration Item

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 In the **Operation** column of the target configuration item, click **More** > **Disable**.

Step 6 Click **OK**.

----End

Deleting a Configuration Item

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Delete** in the **Operation** column of the target configuration item. You can also click the target configuration item and then click **Delete** on the displayed configuration details page.

Step 6 Click **OK**.

----End

11.4.5 Configuration Management (Applicable to Engine 1.x)

The configuration added here is a global configuration. After being added, the configuration takes effect immediately if all microservices registered with the engine use it.

If dynamic configuration is set for a single microservice, the dynamic configuration overwrites the global configuration. For details about how to set dynamic configuration, see **Dynamic Configuration**.

Creating a Configuration

Configuration management provides common configurations for microservices, such as log levels and running parameters. After being added, the configuration item is used as the default one if no same configuration items are defined for microservices.

Configuration items are stored in plaintext. Do not include sensitive data.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Create Configuration Item**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 6 On the **Create Configuration Item** page, select a microservice environment and enter **Configuration Item** and **Value**.

Step 7 Click **OK**.

----End

Importing Configurations

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Import**.

Step 6 Select a microservice environment, click **Import**, and select the target file. A maximum of 150 configuration items can be imported at a time.

Step 7 Click **Close**.

----End

Exporting Configurations

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Export All**.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Deleting a Configuration

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Select the target configuration item and click **Delete**. You can also click **Delete** in the **Operation** column of the target configuration item.

Step 6 Click **OK**.

----End

Editing a Configuration

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **Configuration Management**.

- For microservice engines with security authentication disabled, go to **Step 5**.
- For microservice engines with security authentication enabled, go to **Step 4**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click **Edit** in the **Operation** column of the target configuration item and edit the values of the configuration item.

Step 6 Click **OK**.

----End

11.4.6 System Management

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

11.4.6.1 Overview

A microservice engine may be used by multiple users. Different users must have different microservice engine access and operation permissions based on their responsibilities and permissions.

The exclusive microservice engine with security authentication enabled provides the system management function using the role-based access control (RBAC) through the microservice console.

The exclusive microservice engine with security authentication enabled supports the access of Spring Cloud and Java chassis microservice frameworks.

NOTE

- The RBAC-based system management function is irrelevant to IAM permission management. It is only an internal permission management mechanism of CSE.
- To operate a microservice engine on CSE, you must have both the IAM and RBAC permissions, and the IAM permission takes precedence over the RBAC permission.
- If you perform operations on a microservice engine through APIs or the microservice framework, you only need to have the RBAC permissions.

1. You can use an account associated with the **admin** role to create an account and associate a proper role with the account based on service requirements. The user who uses this account has the access and operation permissions on the microservice engine.
   - When you create an exclusive microservice engine with security authentication enabled, the system automatically creates the **root** account associated with the **admin** role. The **root** account cannot be edited or deleted.
   - You can create an account using the **root** account of the microservice engine or an account associated with the **admin** role of the microservice engine. For details about how to create and manage an account, see **Accounts**.
2. You can create a custom role using an account associated with the **admin** role and grant proper microservice engine access and operation permissions to the role based on service requirements.
   - The system provides two default roles: administrator (**admin**) and developer (**developer**). Default roles cannot be edited or deleted.
   - You can create a custom role using the **root** account of the microservice engine or an account associated with the **admin** role of the microservice engine. For details about how to create and manage a role, see **Roles**.
   - For details about role permissions, see **Table 11-6**.

Table 11-6 Role permissions

Role	Permission Description
Admin	Full permissions for all microservices, accounts, and roles of the microservice engine.
Developer	Full permissions for all microservices of the microservice engine.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Role	Permission Description
Custom role	You can create roles based on service requirements and grant microservice operation and configuration permissions to the roles.

11.4.6.2 Accounts

You can use an account associated with the **admin** role to log in to the microservice engine console and create an account or manage a specified account created in the engine based on service requirements.

Table 11-7 Account management operations

Operation	Description
Adding an Account	Creates an account and associates a proper role with the account. Users who use the account have the access and operation permissions on the microservice engine. You can create up to 1000 accounts, including new accounts and the imported IAM account.
Importing an IAM Account	Imports an IAM account and associates roles with it. Users using this IAM account have the access and operation permissions on the microservice engine. If the imported IAM account needs to connect microservice applications to the engine through programming interface authentication, **reset** its password and then use the new password to configure security authentication parameters. When you use this IAM account to log in to the ServiceStage console with security authentication enabled, you do not need to enter the account and password. However, a password is still required after the VDC read-only user is imported. ServiceStage can manage up to 1000 accounts, including new accounts and the imported IAM account.
Viewing Role Permissions	Displays the permissions of the role associated with a specified account.
Editing an Account	Adds or deletes roles for an account. The **root** account cannot be edited.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Changing the Password	Changes the password of an account that has logged in to the microservice engine based on service requirements or security regulations. • If the account and password are used to register a microservice in the SDK, changing the account and password may affect the service running of the microservice (the microservice cannot be registered with the microservice engine). As a result, the service system will be damaged. Exercise caution when performing this operation. • After the password is changed, update the microservice authentication configuration in a timely manner. – Spring Cloud: see **Connecting Spring Cloud Applications to CSE Engines**. – Java Chassis: see **Connecting Java Chassis Applications to CSE Engines**. • After the password is changed, the account may be locked due to three consecutive incorrect password attempts. The account will be unlocked after 15 minutes.
Resetting a Password	Based on service requirements or security regulations, you can use the account that has logged in to the microservice engine to reset the passwords of other accounts under the microservice engine. • If the account and password are used to register a microservice in the SDK, resetting the account and password may affect the service running of the microservice (the microservice cannot be registered with the microservice engine). As a result, the service system will be damaged. Exercise caution when performing this operation. • After the password is reset, update the microservice authentication configuration in a timely manner. – Spring Cloud: see **Connecting Spring Cloud Applications to CSE Engines**. – Java Chassis: see **Connecting Java Chassis Applications to CSE Engines**. • After the password is reset, the account may be locked due to three consecutive incorrect password attempts. The account will be unlocked after 15 minutes.
Deleting an Account	Deletes an account that is no longer used. The **root** account cannot be deleted. If the account and password are used to register a service in the SDK, deleting the account will affect the service running (the service cannot be registered with the engine). As a result, the service system will be damaged. Exercise caution when performing this operation.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Adding an Account

Before adding an account, you can create a role based on service requirements. For details, see **Creating a Role**.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

NOTE

If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.

Step 5 Choose **Accounts** > **Create Account** and configure account parameters by referring to the following table:

Parameter	Description
Account	Enter an account name. The account name cannot be changed once the account is created.
Role	Select a role based on service requirements. An account can be associated with up to five roles.
Password	Enter the password.
Confirm Password	Enter the password again.

Step 6 Click **OK**.

----End

Importing an IAM Account

Before importing an IAM account, you can create a role based on service requirements. For details, see **Creating a Role**.

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**. In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 4 Choose **Accounts** > **Import IAM User Name**.

Step 5 Select the IAM account to be imported and select account roles. An account can be associated with up to five roles.

Step 6 Click **Confirm**.

The imported account cannot be used for login using a password. If you want to use the imported IAM account to connect microservice applications to the engine through programming interface authentication, **Resetting a Password** first.

----End

Viewing Role Permissions

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 Click the role in the **Role** column of the account to be viewed in the account list. On the displayed page, view the role and permission configuration associated with the account.

----End

Editing an Account

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Accounts** tab page, click **Edit Account** in the **Operation** column of the account to be edited.

Step 6 Select a role based on service requirements. An account can be associated with up to five roles.

Step 7 Click **Save**.

----End

Changing the Password

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the account name and password, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**
- The account for connecting to the microservice engine is not associated with the admin role. You can only change the password of the current login account.
- The account for connecting to the microservice engine is associated with the admin role. You can change the passwords of all accounts of the microservice engine.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Accounts** tab, select the account for logging in to the microservice engine and click **Reset Own Password** in the **Operation** column.

1. Enter the old password and a new password, and confirm the password.
2. After confirming that the password needs to be changed, select **I Understand**.

NOTE

You can also click **Reset Own Password** in the upper right corner of the **System Management** page to change the password of the current login account.

Step 6 Click **Save**.

----End

Resetting a Password

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Accounts** tab page, select the account whose password is to be reset, and click **Reset Password** in the **Operation** column.

1. Enter a new password and confirm the password.
2. After confirming that the password needs to be reset, select **I Understand**.

Step 6 Click **Save**.

----End

Deleting an Account

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Accounts** tab page, click **Delete** in the **Operation** column of the account to be deleted.

Step 6 In the displayed dialog box, enter **DELETE** and click **OK**.

----End

11.4.6.3 Roles

In addition to the default roles **admin** and **developer**, you can use a microservice engine account associated with the **admin** role to log in to the CSE console and perform operations listed in **Table 11-8** based on service requirements.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 11-8 Role management operations

Operation	Description
Creating a Role	Creates a role and configures permission actions for the role in different service and configuration groups. A maximum of 100 roles can be created.
Editing a Role	Modifies the permissions of the created role.
Deleting a Role	Deletes a role that is no longer used. • Deleted roles cannot be restored. Exercise caution when performing this operation. • Before deleting a role, ensure that the role is not associated with any account. For details about how to cancel the association between a role and an account, see **Editing an Account**.
Viewing a Role	Displays the created roles of the microservice engine based on the keyword of the role name.

Creating a Role

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Roles** tab page, click **Create Role**.

Step 6 Enter a role name. The role name cannot be changed once the role is created.

Step 7 Configure permissions.

1. Set **Permission Group**.
   a. Set the service permissions.
      - If you select **All Services**:
        You can perform corresponding permission actions on all microservices of the microservice engine.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If you select **Custom Service Groups**, set the parameters according to **Table 11-9**.

Table 11-9 Custom service group operations

Operation	Description
Adding a Matching Rule	Click **Add Service Group Matching Rule**. Select **Application**, **Environment**, and **Service** based on service requirements to filter the microservices on which the role can perform permission actions. ○ A maximum of 20 microservice matching rules can be set for a custom service group. ○ If multiple matching rules are set for a custom service group, the role has the operation permission on those microservices which meet any of the matching rules. **Application**, **Environment**, and **Service** are three parameters of a microservice: ○ If only one parameter is set for a single matching rule, the role has the operation permission on the microservice that matches the parameter value. For example, if you add **Environment: production**, the role has the operation permission only on the microservice whose environment name is **production**. ○ If more than one parameter is set for a single matching rule, the role has the operation permission on the microservices that match all parameter values. For example, if you add **Environment: production Application: abc**, the role has the operation permission on the microservice whose environment name is **production** and application name is **abc**. ○ When automatic discovery is enabled, microservices query the instance addresses of services such as the registry center, configuration center, and dashboard through the registry center. When you grant the query permission to a microservice, the permission of the default application must be included. In this case, add the matching rule **Application: default**. After the microservice matching rule is set, click **OK**.
Editing a Matching Rule	Click next to the matching rule to be edited. You can reconfigure **Service Group** and **Action** of the matching rule based on service requirements. After the service group matching rule is set, click **OK**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Description
Deleting a Matching Rule	Click next to the matching rule to be deleted. You can delete the matching rule based on service requirements.

b. Set the configuration permissions.

- If you select **All Configurations**:

  You can perform corresponding permission actions on all microservices of the microservice engine.

- If you select **Custom Configuration Groups**, set the parameters according to **Table 11-10**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 11-10 Custom configuration group operations

Operation	Description
Adding a Matching Rule	Click **Add Configuration Group Matching Rule**. Select **Application**, **Environment**, and **Service** based on service requirements to filter the configurations on which the role can perform permission actions. If application-level and microservice-level configurations cannot meet service requirements, you can customize a matching rule to match the configured custom labels and filter the permission actions that can be performed by the role. ○ A maximum of 20 matching rules can be set for a custom configuration group. ○ If multiple matching rules are set for a configuration service group, the role has the operation permission on those configurations which meet any of the matching rules. **Application**, **Environment**, and **Service** are three parameters of a configuration: ○ If only one parameter is set for a single matching rule, the role has the operation permission on the configuration that matches the parameter value. For example, if you add **Environment: production**, the role has the operation permission only on the configuration whose environment name is **production**. ○ If more than one parameter is set for a single matching rule, the role has the operation permission on the configurations that match all parameter values. For example, if you add **Environment: production Application: abc**, the role has the operation permission on the configuration whose environment name is **production** and application name is **abc**. After the configuration matching rule is set, click **OK**.
Editing a Matching Rule	Click next to the matching rule to be edited. You can reconfigure **Configuration Group** and **Action** of the matching rule based on service requirements. After the configuration group matching rule is set, click **OK**.
Deleting a Matching Rule	Click next to the matching rule to be deleted. You can delete the matching rule based on service requirements.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

2. Set **Action**.

   Configure the permission actions that can be performed by the role on the selected service group and configuration group based on service requirements. You can select multiple permission actions.

   - **All**: Add, delete, modify, and query resources in the service group and configuration group.
   - **Add**: Add resources to the service group and configuration group.
   - **Delete**: Delete resources from the service group and configuration group.

     If only **Delete** is selected, you cannot delete resources in the service group and configuration group. You must select **View** at the same time.
   - **Modify**: Modify resources in the service group.

     If only **Modify** is selected, you cannot modify resources in the service group and configuration group. You must select **View** at the same time.
   - **View**: View resources in the service group and configuration group.

Step 8 Click **Create**.

----End

Editing a Role

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Roles** tab page, click **Edit** in the **Operation** column of the role to be edited.

Step 6 Modify **Service Group**, **Configuration Group**, and **Action** based on service requirements.

Step 7 Click **Save**.

----End

Deleting a Role

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Roles** tab, click **Delete** in the **Operation** column of the role to be deleted. In the displayed dialog box, enter **DELETE** and click **OK**.

- Deleted roles cannot be restored. Exercise caution when performing this operation.
- Before deleting a role, ensure that the role is not associated with any account. For details about how to cancel the association between a role and an account, see **Editing an Account**.

----End

Viewing a Role

Step 1 Log in to ServiceStage and choose **Cloud Service Engine** > **Engines**.

Step 2 Select the target microservice engine with security authentication enabled from the **Microservice Engine** drop-down list in the upper part of the page.

Step 3 Choose **System Management**.

Step 4 In the displayed **Security Authentication** dialog box, enter the name and password of the account associated with the **admin** role under the microservice engine, and click **OK**.

NOTE

- If you connect to the microservice engine for the first time, enter the account name **root** and the password entered when **Creating a Microservice Engine**.
- For details about how to create an account, see **Adding an Account**.

Step 5 On the **Roles** tab page, click ﹀ next to the role to be viewed to expand the role details.

Service Group, **Configuration Group**, and **Action** of the role are displayed.

----End

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

12 Key Operations Recorded by CTS

12.1 ServiceStage Operations That Can Be Recorded by CTS

Cloud Trace Service (CTS) records ServiceStage application management operations, enabling you to query, audit, and review operations.

After CTS is **enabled**, the system starts recording operations on ServiceStage resources. You can view the operation records of the last seven days on the CTS console.

Table 12-1 ServiceStage operations that can be recorded by CTS

Operation	Resource Type	Event Name
Creating a component	component	createComponent
Deleting a component	component	deleteComponent
Upgrading a component	component	updateComponent
Starting a component	component	startComponent
Stopping a component	component	stopComponent
Restarting a component	component	restartComponent
Scaling a component	component	scaleComponent

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Operation	Resource Type	Event Name
Rolling back a component	component	rollbackComponent
Deploying a component	component	provisionComponent
Uninstalling a component	component	deprovisionComponent
Creating an application	application	createApplication
Deleting an application	application	deleteApplication
Updating an application	application	updateApplication
Registering a VM agent	vmagent	registerVmagent
Deregistering a VM agent	vmagent	unregisterVmagent
Creating an environment	environment	createEnvironment
Deleting an environment	environment	deleteEnvironment

12.2 Viewing CTS Traces in the Trace List

Scenarios

Cloud Trace Service (CTS) records operations performed on cloud service resources. A record contains information such as the user who performed the operation, IP address, operation content, and returned response message. These records facilitate security auditing, issue tracking, and resource locating. They also help you plan and use resources, and identify high-risk or non-compliant operations.

What Is a Trace?

A trace is an operation log for a cloud service resource, tracked and stored by CTS. Traces record operations such as adding, modifying, or deleting cloud service resources. You can view them to identify who performed operations and when for detailed tracking.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

What Is a Management Tracker and Data Tracker?

A management tracker identifies and associates with all your cloud services, recording all user operations. It records management traces, which are operations performed by users on cloud service resources, such as their creation, modification, and deletion.

A data tracker records details of user operations on data in OBS buckets. It records data traces reported by OBS, detailing user operations on data in OBS buckets, including uploads and downloads.

Constraints

- Before the organization function is enabled, you can query the traces of a single account on the CTS console. After the organization function is enabled, you can only view multi-account traces on the **Trace List** page of each account, or in the OBS bucket or the **CTS/system** log stream configured for the management tracker with the organization function enabled. For details about organization trackers, see **Organization Trackers**.
- You can only query operation records of the last seven days on the CTS console. They are automatically deleted upon expiration and cannot be manually deleted. To store them for longer than seven days, configure transfer to Object Storage Service (OBS) or Log Tank Service (LTS) so that you can view them in OBS buckets or LTS log groups.
- After creating, modifying, or deleting a cloud service resource, you can query management traces on the CTS console 1 minute later and query data traces 5 minutes later.
- Data traces are not displayed in the trace list of the new version. To view them, you need to go to the old version.

Prerequisites

1. **Register with Huawei Cloud and complete real-name authentication.**

   If you already have a Huawei Cloud account, skip this step. If you do not have one, do as follows:

   a. Log in to the **Huawei Cloud official website**, and click **Sign Up** in the upper right corner.

   b. Complete the registration as prompted. For details, see **Registering with Huawei Cloud**.

      Your personal information page is displayed after the registration completes.

   c. Complete individual or enterprise real-name authentication by referring to **Real-Name Authentication**.

2. **Grant permissions for users.**

   If you log in to the console using a Huawei Cloud account, skip this step.

   If you log in to the console as an IAM user, first contact your CTS administrator (account owner or a user in the **admin** user group) to obtain the **CTS FullAccess** permissions. For details, see **Assigning Permissions to an IAM User**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Viewing Traces

After you enable CTS and the management tracker is created, CTS starts recording operations on cloud resources. After a data tracker is created, CTS starts recording user operations on data in OBS buckets. CTS retains operation records of the latest seven days.

This section describes how to query and export operation records of the last seven days on the CTS console.

Viewing Real-Time Traces in the Trace List of the New Edition

Step 1 Log in to the **CTS console**.

Step 2 Log in to the management console, click ☰ in the upper left corner, and choose **Management & Deployment** > **Cloud Trace Service**.

Step 3 In the navigation pane, choose **Trace List**.

Step 4 In the time range drop-down list above the trace list, select a desired query time range: **Last 1 hour**, **Last 1 day**, or **Last 1 week**. You can also select **Custom** to specify a custom time range within the last seven days.

Step 5 The search box above the trace list supports advanced queries. Combine one or more filters to refine your search.

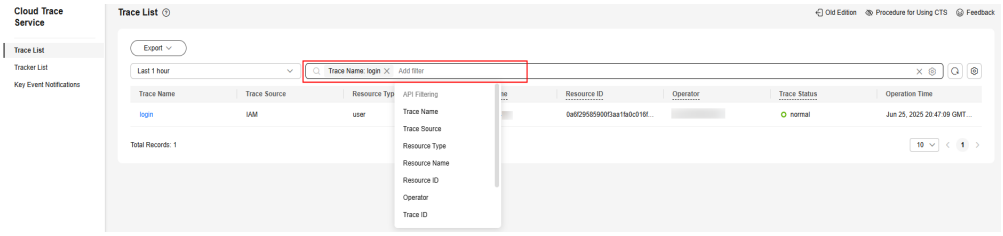
Table 12-2 Trace filtering parameters

Parameter	Description
Trace Name	Name of a trace. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. For details about the operations that can be audited for each cloud service, see **Supported Services and Operations**section "Supported Services and Operations" in the *Cloud Trace Service User Guide*. Example: **updateAlarm**
Trace Source	Cloud service name abbreviation. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. Example: **IAM**
Resource Name	Name of a cloud resource involved in a trace. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. If the cloud resource involved in the trace does not have a resource name or the corresponding API operation does not involve the resource name parameter, leave this field empty. Example: **ecs-name**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Parameter	Description
Resource ID	ID of a cloud resource involved in a trace. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. Leave this field empty if the resource has no resource ID or if resource creation failed. Example: *{VM ID}*
Trace ID	Value of the **trace_id** parameter for a trace reported to CTS. The entered value requires an exact match. Fuzzy matching is not supported. Example: **01d18a1b-56ee-11f0-ac81-******1e229**
Resource Type	Type of a resource involved in a trace. The entered value is case-sensitive and requires an exact match. Fuzzy matching is not supported. For details about the resource types of each cloud service, see **Supported Services and Operations**section "Supported Services and Operations" in the *Cloud Trace Service User Guide*. Example: **user**
Operator	User who triggers a trace. Select one or more operators from the drop-down list. If the value of **trace_type** in a trace is **SystemAction**, the operation is triggered by the service and the trace's operator may be empty. For details about the relationship between IAM identities and operators and the operator username format, see **Relationship Between IAM Identities and Operators**.
Trace Status	Select one of the following options from the drop-down list: • **normal**: The operation succeeded. • **warning**: The operation failed. • **incident**: The operation caused a fault that is more serious than a normal failure, for example, causing other faults.
Enterprise Project ID	ID of the enterprise project to which a resource belongs. To check enterprise project IDs, go to the Enterprise Project Management Service (EPS) console and choose **Project Management** in the navigation pane. Example: **b305ea24-c930-4922-b4b9-******1eb2**
Access Key	Temporary or permanent access key ID. To check access key IDs, hover over your username in the upper right corner of the console and select **My Credentials** from the pop-up list. On the displayed page, choose **Access Keys** in the navigation pane. Example: **HSTAB47V9V*******TLN9**

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.



Step 6 On the **Trace List** page, you can also export and refresh the trace list, and customize columns to display.

- Enter any keyword in the search box and press **Enter** to filter desired traces.
- Click **Export** to export all traces in the query result as an .xlsx file. The file can contain up to 5,000 records.
- Click to view the latest information about traces.
- Click to customize the information to be displayed in the trace list. If **Auto wrapping** is enabled ( ), excess text will move down to the next line; otherwise, the text will be truncated. By default, this function is disabled.

Step 7 (Optional) On the **Trace List** page of the new edition, click **Old Edition** in the upper right corner to switch to the **Trace List** page of the old edition.

----End

Viewing Traces in the Trace List of the Old Edition

Step 1 Log in to the **CTS console**.

Step 2 Log in to the management console, click in the upper left corner, and choose **Management & Deployment** > **Cloud Trace Service**.

Step 3 In the navigation pane, choose **Trace List**.

Step 4 Each time you log in to the CTS console, the new edition is displayed by default. Click **Old Edition** in the upper right corner to switch to the trace list of the old edition.

Step 5 In the upper right corner of the page, set a desired query time range: **Last 1 hour**, **Last 1 day**, or **Last 1 week**. You can also click **Customize** to specify a custom time range within the last seven days.

Step 6 Set filters to search for your desired traces.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Table 12-3 Trace filtering parameters

Parameter	Description
Trace Type	Select **Management** or **Data**. • Management traces record operations performed by users on cloud service resources, including creation, modification, and deletion. • Data traces are reported by OBS and record operations performed on data in OBS buckets, including uploads and downloads. Management traces record operations performed by users on cloud service resources, including creation, modification, and deletion.
Trace Source	Select the name of the cloud service that triggers a trace from the drop-down list.
Resource type	Select the type of the resource involved in a trace from the drop-down list. For details about the resource types of each cloud service, see **Supported Services and Operations**section "Supported Services and Operations" in the *Cloud Trace Service User Guide*.
Operator	User who triggers a trace. Select one or more operators from the drop-down list. If the value of **trace_type** in a trace is **SystemAction**, the operation is triggered by the service and the trace's operator may be empty. For details about the relationship between IAM identities and operators and the operator username format, see **Relationship Between IAM Identities and Operators**.
Trace Status	Select one of the following options: • **Normal**: The operation succeeded. • **Warning**: The operation failed. • **Incident**: The operation caused a fault that is more serious than a normal failure, for example, causing other faults.

Step 7 Click **Query**.

Step 8 On the **Trace List** page, you can also export and refresh the trace list.

- Click **Export** to export all traces in the query result as a CSV file. The file can contain up to 5,000 records.
- Click to view the latest information about traces.

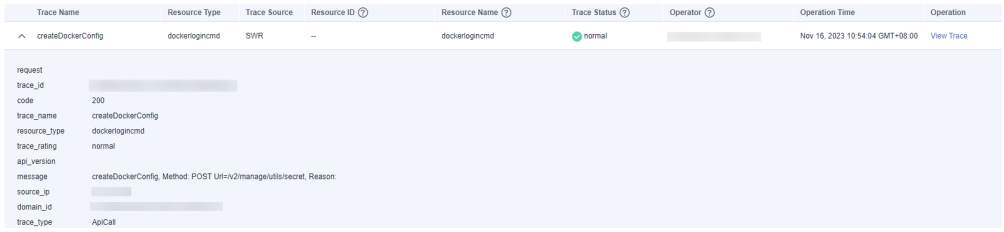
Step 9 In the **Tampered or Not** column of a trace, check whether the trace is tampered with.

- If no, **No** is displayed.

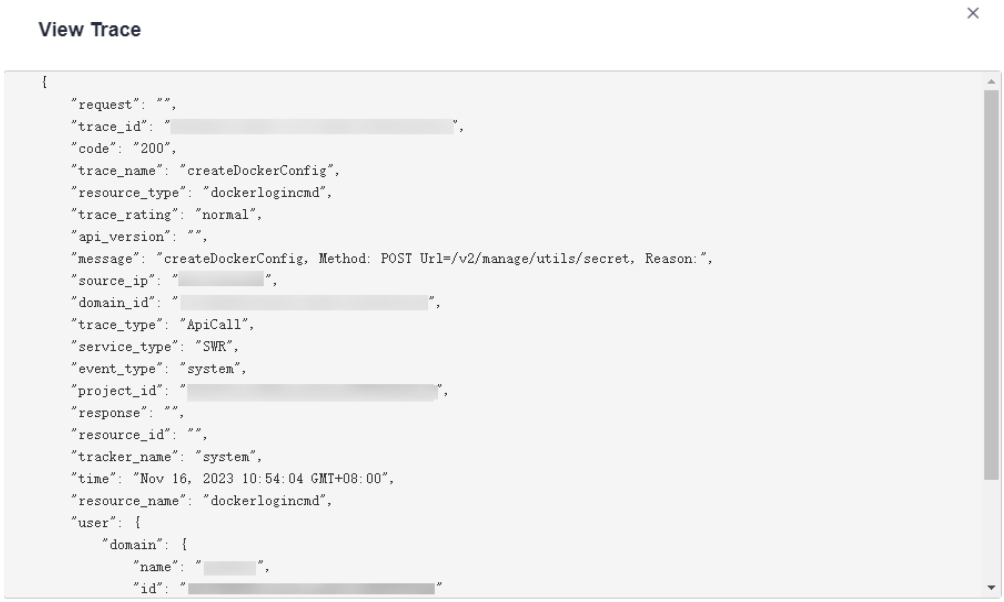
 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

- If yes, **Yes** is displayed.

Step 10 Click ⌄ on the left of a trace to expand its details.



Step 11 Click **View Trace** in the **Operation** column. The trace details are displayed.



Step 12 (Optional) On the **Trace List** page of the old edition, click **New Edition** in the upper right corner to switch to the **Trace List** page of the new edition.

----End

Helpful Links

- For details about the key fields in the trace structure, see **Trace Structure**section "Trace References" > "Trace Structure" in the *Cloud Trace Service User Guide* and **Example Traces**section "Trace References" > "Example Traces" in the *Cloud Trace Service User Guide*.
- You can use the following examples to learn how to query a specific trace:
  - Use CTS to audit Elastic Volume Service (EVS) creation and deletion operations from the last two weeks. For details, see **Security Auditing**.
  - Use CTS to locate a fault or creation failure for an Elastic Cloud Server (ECS). For details, see **Fault Locating**.
  - Use CTS to check all operation records for an ECS. For details, see **Resource Tracking**.

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

13 Viewing Monitoring Metrics and Alarms

Introduction

Application Operations Management (AOM) monitors and displays the running status of ServiceStage and the usage of each metric, and creates alarm rules for monitoring items.

After you use ServiceStage to deploy components, AOM can associate monitoring metrics of the components to help you master the performance metrics of the components in real time and accurately master the running status of the components.

Setting Monitoring and Alarms

ServiceStage supports container-based and VM-based deployment modes.

- Container-based deployment

  CCE works with AOM to comprehensively monitor clusters. When a node is created, the ICAgent (the DaemonSet named **icagent** in the kube-system namespace of the cluster) of AOM is installed by default. The ICAgent collects monitoring data of underlying resources and workloads running on the cluster, and uploads the data to AOM. In addition, after **Customizing Component Running Metrics**, the ICAgent can collect monitoring data of user-defined load metrics and upload the data to AOM.

  After **Configuring Alarm Thresholds for Resource Monitoring**, alarms generated during component running are reported to AOM.

- VM-based deployment

  Before deploying components on a VM, install the VM agent on the VM. During the installation, the AOM ICAgent is installed by default. The monitoring metrics of the VM-deployed components are uploaded to AOM.

Supported Metrics

Metrics reflect the resource performance or status.

- Metrics supported by the container-deployed components

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Basic resource monitoring includes CPU, memory, and disk monitoring. For details, see **Table 13-1**.

Table 13-1 Resource metrics

Metric	Description	Value Range	Unit
Total CPU cores (cpuCoreLimit)	Total number of CPU cores that have been applied for a measured object	≥1	Cores
Used CPU cores (cpuCoreUsed)	Number of CPU cores used by a measured object	≥0	Cores
CPU usage (cpuUsage)	CPU usage of a measured object, that is, the ratio of the used CPU cores to the total CPU cores.	0%–100%	%
Total physical memory (memCapacity)	Total physical memory that has been applied for a measured object	≥0	MB
Physical memory usage (memUsage)	Percentage of the used physical memory to the total physical memory	0%–100%	%
Used physical memory (memUsed)	Used physical memory of a measured object	≥0	MB
Disk read rate (diskReadRate)	Volume of data read from a disk per second	≥0	KB/s
Disk write rate (diskWriteRate)	Volume of data written into a disk per second	≥0	KB/s
Downlink rate (recvPackRate)	Number of data packets received by the NIC per second	≥0	Packets per second (PPS)

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Metric	Description	Value Range	Unit
Total file system (filesystemCapacity)	Total file system capacity of a measured object. This metric is available only for containers using the Device Mapper storage drive in the Kubernetes cluster of version 1.11 or later.	≥0	MB
Downlink rate (recvBytesRate)	Inbound traffic rate of a measured object	≥0	Byte per second (BPS)
Downlink error rate (recvErrPackRate)	Number of error packets received by an NIC per second	≥0	PPS
Uplink rate (sendPackRate)	Outbound traffic rate of a measured object	≥0	BPS
Uplink error rate (sendErrPackRate)	Number of error packets sent by the NIC per second	≥0	PPS
Uplink rate (sendBytesRate)	Outbound traffic rate of a measured object	≥0	BPS
Error packets (rxPackErrors)	Number of error packets received by a measured object	≥0	Packets
Threads (threadsCount)	Number of threads created on a host	≥0	N/A
Available file system (filesystemAvailable)	Available file system capacity of a measured object. This metric is available only for containers using the Device Mapper storage drive in the Kubernetes cluster of version 1.11 or later.	≥0	MB

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Metric	Description	Value Range	Unit
File system usage (filesystemUsage)	File system usage of a measured object, that is, the ratio of the used file system to the total file system. This metric is available only for containers using the Device Mapper storage drive in the Kubernetes cluster of version 1.11 or later.	≥0	%
Handles (handleCount)	Number of handles used by a measured object	≥0	N/A
Component status (status)	Status of an application group	• 0: normal • 1: abnormal	N/A
Total virtual memory (virMemCapacity)	Total virtual memory that has been applied for a measured object	≥0	MB

- Metrics supported by the VM-deployed components

  In AOM, VM-deployed components refer to processes, and VM-deployed component metrics refer to process metrics. For details, see **Table 13-2**.

  Table 13-2 Process metrics

Metric	Description	Value Range	Unit
Total CPU cores (cpuCoreLimit)	Total number of CPU cores that have been applied for a measured object	≥1	Cores
Used CPU cores (cpuCoreUsed)	Number of CPU cores used by a measured object	≥0	Cores
CPU usage (cpuUsage)	CPU usage of a measured object, that is, the ratio of the used CPU cores to the total CPU cores.	0%–100%	%
Handles (handleCount)	Number of handles used by a measured object	≥0	N/A

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Metric	Description	Value Range	Unit
Total physical memory (memCapacity)	Total physical memory that has been applied for a measured object	≥0	MB
Physical memory usage (memUsage)	Percentage of the used physical memory to the total physical memory	0%–100%	%
Used physical memory (memUsed)	Used physical memory of a measured object	≥0	MB
Status (status)	Process status	• 0: normal • 1: abnormal	N/A
Threads (threadsCount)	Number of threads used by a measured object	≥0	N/A
Total virtual memory (virMemCapacity)	Total virtual memory that has been applied for a measured object	≥0	MB

 Copyright © Huawei Cloud Computing Technologies Co., Ltd.