Deploy

Getting Started

Issue 01

Date 2023-11-30

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

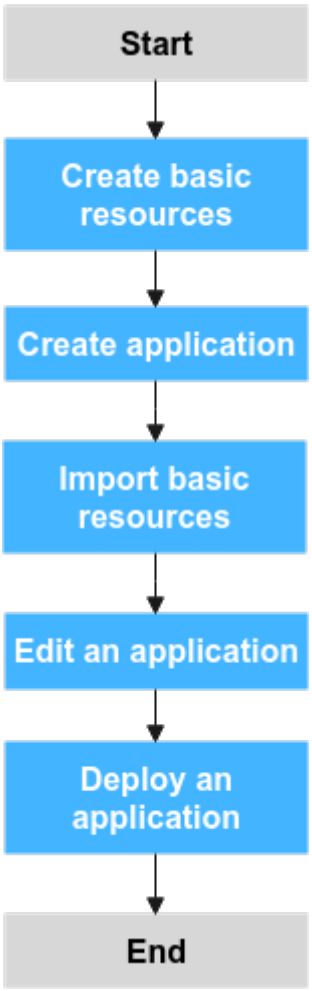
1 Overview

CodeArts Deploy provides visualized and automatic deployment services. It has various deployment actions to help you make a standard deployment process, reduce deployment costs, and improve release efficiency.

CodeArts Deploy has the following features:

- Supports host (physical machine and virtual machine) and container deployment.
- Provides system templates such as Tomcat, Spring Boot, and Django for you to create applications quickly. You can drag and drop atomic actions to orchestrate applications flexibly.
- Supports deployment with multiple hosts in an environment at the same time.
- Implements container deployment using Cloud Container Engine (CCE).
- Saves custom templates to create applications at one click.
- Supports parameter configuration, provides parameter types such as text, environment, and enumeration, and supports dynamic parameter replacement during application deployment.
- Seamlessly integrates with CodeArts Pipeline to support continuous service release.
- Generates run logs for atomic actions and provides keywords to accurately match FAQs. If the deployment fails, you can quickly locate the cause and find a solution.

If you are developing a project locally and want to use the deployment service to deploy the project, the process is as follows:



The operations are as follows:

- **Create basic resources**.
- **Create an application**.
- **Import basic resources**.
- **Edit an application**.
- **Deploy an application**.

2 Creating a Deployment Task

This section uses **Deploying a Tomcat Application** as an example to describe how to deploy an application on a host.

Preparations

- A project is available. If no project is available, **create one** first.
- You have permission to create applications. For details, see **Permission Management**.
- The related software package has been compiled.

NOTICE

You can set **Source** to **Artifact** or **Build task**. In this example, **Artifact** is selected. Therefore, you need to prepare the compiled software package in advance. If you select **Build task**, you do not need to prepare the software package.

Procedures

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

Step 2 Create basic resources.

1. Choose **Settings > General > Basic Resources**. The **Host Clusters** page is displayed by default.
2. Click **Create Host Cluster**, enter the following information, and click **Save**.

Parameter	Mandatory	Description
Cluster Name.	Yes	Enter a custom name.
OS	Yes	Select **Linux** or **Windows** based on the OS of the host to be added.

Parameter	Mandatory	Description
Use Proxy	Yes	If the target host cannot connect to the public network, you need to select a host bound with an EIP as the proxy host to connect CodeArts to the target host. Example: disabled.
Execution Host	Yes	A resource pool is a collection of physical environments where commands are executed during software package deployment. In this scenario, select **official**.
Description	No	Enter a description.

3. (The following uses a target host running Linux as an example.) Click **Add Target Host**, enter the following information, and click **OK**.

Table 2-1 Parameters of the target host (Linux)

Parameter	Mandatory	Description
Host Name	Yes	Enter a custom name.
Proxy Host	Yes	Select a network proxy that cannot connect to the public network.
IP	Yes	Enter the private or public IP address of the target host.
OS	Yes	Keep the default value because it is the OS of your host cluster.
Authorization	Yes	Select a **Password** or **Key** for authentication as required. – Password: The **Username** and **Password** are displayed on the page. Take ECS as an example. You need to enter the ECS username and password. – Key: The **Username** and **Key** are displayed on the page. For details about how to generate and obtain a key, see **Obtaining the Linux Key**.
SSH Port	Yes	Port **22** is recommended. Custom ports are supported.

4. Click ▷ in the **Operation** column of a host to start the host for connectivity verification.

Step 3 Choose **CICD** > **Deploy**.

Step 4 Create an application.

1. Click **Create Application**. On the **Set Basic Information** page that is displayed, modify the basic information such as **App Name**, **Description**, and **Execution Resource Pool** as required.
2. After editing the basic application information, click **Next**. The deployment template selection page is displayed.

   Select **Deploying a Tomcat Application** and click **OK**.

NOTE

The following describes the initialization procedure and parameters of the Tomcat application deployment template. For details, see **Deployment Actions**.

Step 5 Import basic resources.

1. Switch to the **Environment Management** tab page and click **Create Environment**.
2. Enter basic information, such as the **Environment**, **OS**, and **Description**, and click **Save**.
3. Switch to the **Resource** tab page and click **Import Host**.
4. The system automatically filters all clusters that meet the requirements of the current environment. In the dialog box that is displayed, select the target host cluster and click in the **Operation** column of a host to import the host to the environment.

Step 6 Edit an application.

1. Install JDK

   Table 2-2 Parameters for installing the JDK

Parameter	Description
Action Name	Name of an added deployment action.
Environment	Target environment.
JDK Version	JDK version.
Installation Path	Installation path of JDK.
Action Control	– Whether to enable this action. – Whether to continue the task even if this action fails. – Whether to execute this action with the sudo permission.

2. Install Tomcat

Table 2-3 Parameters for installing Tomcat

Parameter	Description
Action Name	Name of an added deployment action.
Environment	Target environment.
Tomcat Version	Tomcat version.
Installation Path	Installation path of Tomcat.
HTTP Port	Default port: **8080**
AJP Port	Default port: **8009**
Service Shutdown Port	Default port: **8005**
Action Control	– Whether to enable this action. – Whether to continue the task even if this action fails. – Whether to execute this action with the sudo permission.

3. Stop Tomcat

Table 2-4 Parameters for stopping Tomcat

Parameter	Description
Action Name	Name of an added deployment action.
Environment	Target environment.
Operation	**Start** and **Stop** are available.
Absolute Path	Absolute path of the Tomcat service.
Action Control	– Whether to enable this action. – Whether to continue the task even if this action fails. – Whether to execute this action with the sudo permission.

4. Select Deployment Source

Table 2-5 Parameters for selecting a deployment source

Parameter	Description
Action Name	Name of an added deployment action.
Source	Two types are available: **Artifact** and **Build task**.

Parameter	Description
Environment	Target environment.
Software package	You can select an existing software package in CodeArts Artifact or from a local host. **package_url** indicates the path of the software package in CodeArts Artifact. **NOTE** Local software packages or files uploaded to CodeArts Artifact can be reused.
Download Path	Where the downloaded software package is stored.
Action Control	– Whether to enable this action. – Whether to continue the task even if this action fails. – Whether to execute this action with the sudo permission.

NOTICE

In this example, **Artifact** is selected as the deployment source. If you select **Build task**, see in the *User Guide*Selecting a Deployment Source.

5. Start Tomcat

Table 2-6 Parameters for starting Tomcat

Parameter	Description
Action Name	Name of an added deployment action.
Environment	Target environment.
Operation	**Start** and **Stop** are available.
Absolute Path	Absolute path of the Tomcat service.
HTTP Port	HTTP port of the Tomcat service.
AJP Port	AJP port of the Tomcat service.
Service Shutdown Port	Shutdown port monitored by the Tomcat service.

Parameter	Description
Waiting Time	The time required for starting the service. If you set **Operation** to **Start**, the system checks the process during startup to determine whether the service is started successfully. You can adjust the time based on the actual time required for starting the service. If the time is improper, the check fails.
Action Control	– Whether to enable this action. – Whether to continue the task even if this action fails. – Whether to execute this action with the sudo permission.

6. Health test via URLs

Table 2-7 Parameters for health test via URLs

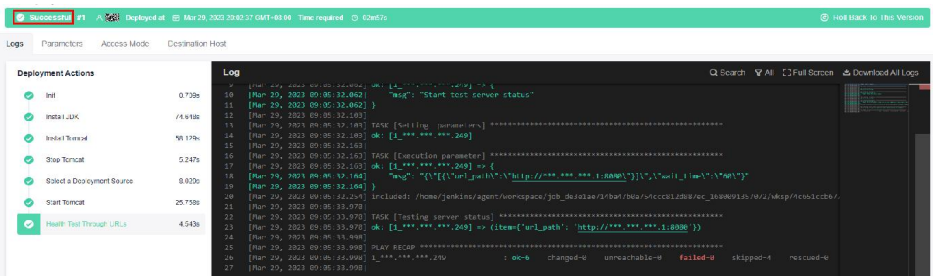
Parameter	Description
Action Name	Name of an added deployment action.
Environment	Target environment.
Retries	The number of times you want the system to retry the health test before the test is considered to have failed.
Interval (s)	Test interval.
Test Path	Path of the service to be tested. You can add multiple paths.
Action Control	– Whether to enable this action. – Whether to continue the task even if this action fails.

After the **Health Test via URLs** parameter is set, all parameters are completed. For more configuration details, see "Managing Applications" > "Editing an Application"**Editing an Application**.

Step 7 Deploy an application.

After setting the preceding parameters, click **Save and Deploy** to deploy an application.

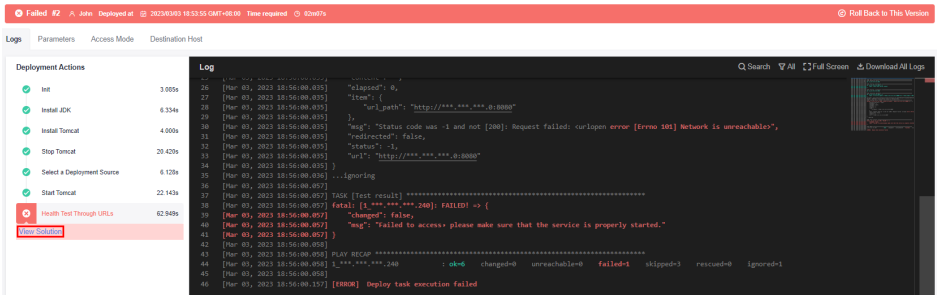
- Application deployed.



- If an application fails to be deployed, perform the following steps to locate the fault:

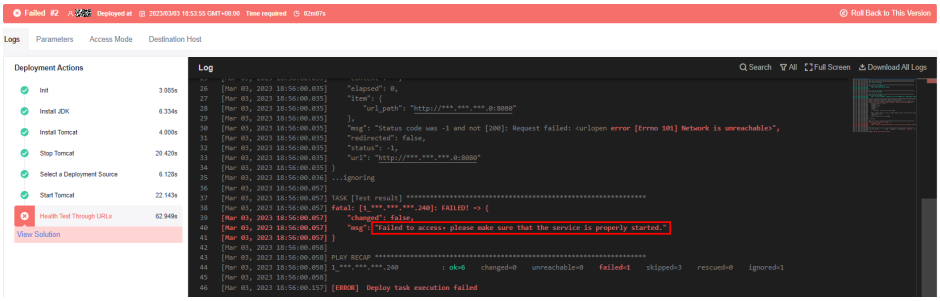
  a. **View the automatic matching solution.**

  If the deployment fails, click **View Solution** under deployment actions to go to the corresponding help center page.



  b. **Search for a solution manually.**

  If the issue persists, paste the error information (content in the red box) to the search box of the help center and manually search for the solution.



----End

(Optional) Verifying the Deployment Result

The **Deploying a Tomcat Application** template supports deployment result verification. This section describes how to verify the deployment result. After the application is successfully deployed, enter the following information in the address box of the browser:

<Public IP address of the target host>:**8080/tomcat-demo**

Press **Enter**. If the following information is displayed, the application is successfully deployed.