CodeArts Artifact

FAQs

 Issue
 01

 Date
 2024-10-30





Copyright © Huawei Technologies Co., Ltd. 2024. 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 Technologies Co., Ltd.

Trademarks and Permissions

NUAWEI and other Huawei trademarks are trademarks 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 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.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:

https://www.huawei.com/en/psirt/vul-response-process

For vulnerability information, enterprise customers can visit the following web page: <u>https://securitybulletin.huawei.com/enterprise/en/security-advisory</u>

Contents

1 Release Repo	1
1.1 Why Can't I Upload Files or Create Directories on the Release Repos Homepage?	1
1.2 Can I Change the Dependency ID in pom.xml to Invoke a JAR File in My Release Repos?	1
1.3 Can I Restore Files in the Recycle Bin of My Release Repos?	1
2 Self-Hosted Repo	3
2.1 How Do I Upload Snapshots to a Maven Repository?	3
2.2 How Do I Pull Components from a Maven Repository?	6
2.3 Can I Call Software Packages in Self-Hosted Repos During Local Builds?	6
2.4 What Should I Do With Error Code 500 When Uploading Maven Package for a Gradle Build?	6
2.5 Why Did the Dependency WAR or JAR Files Fail to Be Downloaded?	7
2.6 Why Is Error 401 Returned When Uploading Maven Components to Self-Hosted Repos?	7

Release Repo

1.1 Why Can't I Upload Files or Create Directories on the Release Repos Homepage?

The top-level directory names on this page map to the names of your projects that hold each software package.

You can only browse files and directories in this directory.

Click a project name, and upload files and create directories there.

1.2 Can I Change the Dependency ID in pom.xml to Invoke a JAR File in My Release Repos?

No.

Packages in Release Repos are used for deployment, not as dependencies during build.

You need to first upload your dependency to a self-hosted repo.

1.3 Can I Restore Files in the Recycle Bin of My Release Repos?

Symptom

A file cannot be restored from the recycle bin page. A message indicating that **duplicate file exists** is displayed.

Cause Analysis

A file with the same name exists at the restored location in the repository.

Solution

You can choose **Move and replace**, **Do not move**, or **Move and rename**.

- **Move and replace**: The file restored from the recycle bin will replace the file with the same name in the restored location.
- **Do not move**: Ignore the file restoration in the recycle bin.
- **Move and rename**: Both the original file and the file in the recycle bin are retained in the restored location. The file restored from the recycle bin will be renamed.

2 Self-Hosted Repo

2.1 How Do I Upload Snapshots to a Maven Repository?

Background

Snapshots can be uploaded in any of the following ways:

- Uploading Snapshots on the Release Repos Page
- Uploading Snapshots Using the Maven CLI
- Releasing Snapshots to a Maven Repository Through CodeArts Build

Uploading Snapshots on the Release Repos Page

- **Step 1** Log in to CodeArts.
- **Step 2** Choose **Services** > **Artifact**, click the **Self-hosted Repos** tab, and find the target repository.
- **Step 3** Click the Snapshot repository in the repository list. Click **Upload**. In the displayed dialog box, select **GAV**.

All Types • 🔭 Advanced Search +	Create ··· 🔄 🔰 mvn_1 (Snapshot) 💿	View All Repositories Settings Tutorial Proxy Settings Upload
Enter repo name.	Q	
Repository View 🛛 Vers	Upload 🔍 Help	×
+ 👉 docker≡aaaa + 🗸 mvn_1 (Release)	Target Repository	
wwn_1 (Snapshot) wwn_3 (Release)	mvn_1 (snapsnot)	<u> </u>
₩ Wn_3 (Snapshot)	V Maven	O
	Upload Mode 🞯	
	POM GAV	
	File @	
	* Extension @	
+ 😴 pipi-test-729 + 🔁 pypi-test826		
💽 🏕 docker	Upload Cancel	

There are two GAV definition modes.

GAV Definition Description Mode	
POM	GAV information is extracted from POM files.
GAV	GAV information is manually specified.

Step 4 Set related parameters as prompted and upload the package.

----End

Uploading Snapshots Using the Maven CLI

- **Step 1** Select Maven as the package type, and choose the **Snapshot** repository in the repository list.
- **Step 2** Click **Tutorial** in the upper right corner.
- **Step 3** Configure the local Maven tool by following the configuration guide.
- Step 4 Run mvn deploy to upload the Maven project.

In the Maven CLI, access the directory where the **pom.xml** file of the Maven project is stored, then run the following command to upload a local JAR package: mvn deploy:deploy-file -DgroupId=com.huawei -DartifactId=aopalliance -Dversion=1.0-SNAPSHOT -Dpackaging=jar -Dfile=D:\aopalliance-1.0-SNAPSHOT.jar -Durl={Maven Snapshot address} -DrepositoryId=snapshots

D NOTE

- Set DgroupId, DartifactId, Dversion, and Dpackaging as required.
- Set **Dfile** to the absolute path of the local JAR package.
- Set **Durl** to the Maven snapshot path, which can be obtained by clicking \Box in the following figure.

🐓 mvn_1 (Snapsl	hot) 💿	View All Repositories	Settings	Tutorial	Proxy Settings	Upload	
General Repository Pe	ermissions Resources Operatio	n Logs					
Details							
Repository Name	mvn_1(Snapshot)						
Package Type	🔰 Maven						
Repository Type 💼 Local Repository							
Repository Path				-	0		
Relative Path							
Creator							
Created	May 17, 2024 17:09:20 GMT+08:00)					
Modifier							
Modified	May 21, 2024 23:51:06 GMT+08:00)					
Artifact Count / Size	View 💿						
Description							

----End

Releasing Snapshots to a Maven Repository Through CodeArts Build

Step 1 Go to Repo, open the **pom.xml** file, and define the GAV information of the component to be uploaded.

Web-Dem	10 / pom.xml	
🖟 pom.x	xml 🕞 History 🐻 Blame 🖏 Comparison	0 ± 0
1 2 3 4 5 6 7 8	<pre>condict: walkine "http://maxee.apache.org/2004.4.0.0" walkin:xsl="http://maxee.apache.org/2004/00_5chema-instance" xsl:schematic-condition="http://maxee.apache.org/2004.4.0.0" http://maxee.apache.org/maxeen.v4_0_0.ssd"></pre>	EXCLUSION CONTRACTOR OF CONTRA
9 10 11	<pre><name>Demo Dropklzard Application</name></pre>	

- When a build task is run, CodeArts Build identifies the component properties uploaded to the Maven repository based on the definition.
- version: Releases are uploaded by default. To upload a Snapshot, add the suffix **SNAPSHOT** to the value of **version**, for example, **1.0-SNAPSHOT**.
- Step 2 Edit a build task in the build action Build with Maven:
 - In the command box, comment out the mvn package command (add # before the command) and uncomment the mvn deploy command (delete # before the command).
 - Click Release to Self-hosted Repos, and select Configure all POMs.

	build a Java project with Apache Maven. View guide
Action I	lame
Build v	/ith Maven
Tool Ve	sion
maver	3.5.3-jdk8-open
Comma	nds
6	 -B: Run in Datch mode to avoid ArrayIndexoutOrBoundsException during log printing.
	# Package a project without performing unit tests. mmun package a project without performing unit tests.
	# Pentorm unit tests and use test reports for analysis. # Enable test report printing and specify the storage location.
	# Package a project and release dependencies to Self-hosted Repos. # Palase huild results to Self-hosted Repos.
	mvn deploy -Dmaven.test.skip=true -U -e -X -B
setting	File Configuration 🍘
setting	File Configuration
setting Release	File Configuration to Self-hosted Repos
setting Release	File Configuration
setting Release	File Configuration to Self-hosted Repos To nfigure POM Configure all POMs
setting Release	File Configuration to Self-hosted Repos To not configure POM Configure all POMs tase
setting Release Rele	File Configuration to Self-hosted Repos To not configure POM Configure all POMs tase
setting Release Rele	File Configuration to Self-hosted Repos Do not configure POM Configure all POMs ase
setting Release Rele	File Configuration to Self-hosted Repos To not configure POM Configure all POMs ase
setting Release Rele Sna	File Configuration to Self-hosted Repos Configure all POMs ase pshot
setting Release Rele Sna	File Configuration to Self-hosted Repos To not configure POM Configure all POMs ase pshot
setting Release Rele Sna	File Configuration to Self-hosted Repos To configure all POMs ase pshot
setting Release Rele Sna	File Configuration to Self-hosted Repos To not configure POM Configure all POMs tase pshot
setting Release Rele Sna	File Configuration to Self-hosted Repos To not configure POM Configure all POMs tase pshot

Step 3 Run a build task.

After the build task is executed, you can find the generated Maven component in the Maven repository.

----End

2.2 How Do I Pull Components from a Maven Repository?

Step 1 Go to the self-hosted repo page, and click the name of the component to be pulled. The **General** page is displayed.

🌍 All Types 🔹 🌹 Advanced S	Gearch + Create … 🗉	🔰 m2122 (Relea	ase) 💿	View All Repositories	Settings Tutorial	Proxy Settings Upload
Enter repo name.	Q	m2122 (Release) / asm	a / asm / mayen-metadata xml			↓ Download 前 Delete
E Repository View	Version View	General Artifact Secu	urity			<u> </u>
🛨 👉 docker 🔲 aaaa						
\pm 🔰 mvn_1 (Release)		Details				
🛨 🐓 mvn_1 (Snapshot)		Repository Name	m2122(Release)			
🛨 🔰 mvn_3 (Release)		Package Type	🖌 Maven			
🛨 🐓 mvn_3 (Snapshot)		Repository Type	 Virtual Repository 			
🖃 🐓 m2122 (Release)		Repository Path	ingenie and second second	distant states		0
asm 📄 🧮 asm		Relative Path	/asm/asm/maven-metadata.xml	0		
🖃 🧮 asm		Creator	cloudrelease			
- 🕂 🚞 1.3.3		Created	May 29 2024 16:42:08 GMT+08:0	0		
🗈 maven-metadata.xml	⊻ @	Madifiar	slaudralaase			
\pm 🔰 testMaven222 (Release)		Modifier	cioudielease			
🖃 🐓 maven-virtual-test81 (Release)		Modified	May 29, 2024 16:42:08 GMT+08:0	0		
🛨 🔟 npm_1		Size	361.00 B			
🛨 🔲 npm_3		Last Downloaded				
🛨 🔲 dqweqw31		Download Count				
🛨 🗃 ssssssssss		Source	Client			

Step 2 Obtain the dependency download address, copy it, and paste it to the **pom.xml** file.

----End

2.3 Can I Call Software Packages in Self-Hosted Repos During Local Builds?

Yes.

Go to the repository where the packages are stored and click **Tutorial** in the upper right corner. Download the configuration file and modify it by following the guide.

2.4 What Should I Do With Error Code 500 When Uploading Maven Package for a Gradle Build?

Symptom

A build task fails, and the log information similar to the following is displayed.



Cause Analysis

The release address, rather than the snapshot address, is set.

Solution

Change the address to the snapshot address and upload the package again.

2.5 Why Did the Dependency WAR or JAR Files Fail to Be Downloaded?

Symptom

The local tools cannot download components in the self-hosted repo. A message is displayed indicating that the POM file cannot be found. The log information similar to the following is recorded.

3: Task failed with an exception.
* What went wrong:
Execution failed for task ':app:mergeXichangDebugResources'.
> Could not resolve all files for configuration ':app:xichangDebugRuntimeClasspath'.
> Could not find com.pandabus.android:PBPullToRefresh-library:1.8.3.
Searched in the following locations:
- https://sharasharasharasharasharasharasharasha
##P8PullToRefresh-library-1.8.3.pps

Cause Analysis

The POM file is missing in the dependency.

When downloading dependencies using Gradle or Maven, you need to download a POM file first, and then a JAR or WAR file. Otherwise, the download will fail.

Solution

Re-upload the components that cannot be downloaded according to the components uploading standard.

2.6 Why Is Error 401 Returned When Uploading Maven Components to Self-Hosted Repos?

Symptom

Failed to upload Maven components to self-hosted repos from the local IDE, and **401-Insufficient Permission** is displayed.

Cause Analysis

The self-hosted repo information configured in the **pom.xml** file does not match the **settings.xml** file.

Solution

When uploading components, replace the **repository_id** value in the **distributionManagement** element of the **pom.xml** file with the **repository_id** value in the **server** element of the **settings.xml** file.

The uploading process is as follows:

- **Step 1** Go to the self-hosted repo page, and choose Maven from the repository list.
- **Step 2** Click **Tutorial** in the upper right corner.
- **Step 3** Configure the local Maven tool by following the configuration guide.
- Step 4 Run mvn deploy to upload the Maven project.
 - In the Maven CLI, access the directory where the pom.xml file of the Maven project is stored, check whether the repository_id value in the distributionManagement element of the pom.xml file matches the repository_id value in the server element of the settings.xml file.

setting.xml

<server></server>	RETERING REPORTED AND A DESCRIPTION OF THE DESCRIPT
In release ch-north and a sharp and	
<username>cn=norun=@www.erawaerawaerawaerawaerawaerawaerawae</username>	ALONIFLONIFLONIFLONIFLONIFLONIFLONIFLONIF
<pre><password>200092009300</password></pre>	t i i i i i i i i i i i i i i i i i i i
<server></server>	
<id>snapshot cn-north-Zakkakakakakakakakakakakakakakakakakaka</id>	stands and the stand
<username>cn-north-</username>	were were were were were were were were
<pre><pre>cpassword >ws=tax=tax=</pre>/password></pre>	
(aprilar)	
(/ SELVEL/	
<server></server>	
<id>z_mirrors</id>	
pom.xml₄	
	Ĭ
<dependencies></dependencies>	
<groupid>a</groupid>	
<artifactid>a</artifactid>	
<pre><version>a</version></pre>	
(distributionManagement)	
<repository></repository>	
(id)release cn-north-MCARCARCARC	X CAN AN A CANCELE AND A CANCELE AND A CANCELE AND A CANCELED AND A CANCEL
<ur><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><url><lu><url><url><lu><url><url><lu><url><lu><url><lu><url><lu><url><lu><url><lu><url><lu><url><lu><url><lu><url><lu><url><lu><url><lu><url><lu><url><url< td=""><td></td></url<></url></lu></url></lu></url></lu></url></lu></url></lu></url></lu></url></lu></url></lu></url></lu></url></lu></url></lu></url></lu></url></lu></url></url></lu></url></url></lu></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></url></ur>	
<pre> <smapshotrepository></smapshotrepository></pre>	
<id>/id>//#E/DIME/JOIN/JOIN/JOIN/JOIN/JOIN/JOIN/JOIN/JOIN</id>	SINTATINELASINELASINELASINELASINELASINELASINELASINELASINELASINELASINELASINELASINELASINELASINELASINELASINELASINE
Curl AND	ĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸĸ
<pre>//distributionManagement> //project></pre>	
a brolloos	0

2. Upload the local JAR package: mvn deploy:deploy-file -DgroupId=com.huawei -DartifactId=aopalliance -Dversion=1.0 -Dpackagi=jar

----End