

# Maven 设置阿里云中央仓库的settings.xml



小波同学 [关注](#) IP属地: 广东

0.668 2022.07.13 16:58:01 字数 216 阅读 4,499

## Maven 中央仓库地址大全

### 1、阿里中央仓库（首推1）

```
1 <repository>
2   <id>aliyun</id>
3   <name>aliyun maven Repository</name>
4   <url>https://maven.aliyun.com/repository/public</url>
5 </repository>
```

### 2、camunda.com 中央仓库（首推2）

```
1 <repository>
2   <id>activiti-repos2</id>
3   <name>Activiti Repository 2</name>
4   <url>https://app.camunda.com/nexus/content/groups/public</url>
5 </repository>
```

### 3、spring.io 中央仓库

```
1 <repository>
2   <id>springsource-repos</id>
3   <name>SpringSource Repository</name>
4   <url>http://repo.spring.io/release/</url>
5 </repository>
```

### 4、maven.apache.org 中央仓库

```
1 <repository>
2   <id>central-repos</id>
3   <name>Central Repository</name>
4   <url>http://repo.maven.apache.org/maven2</url>
5 </repository>
```

## 5、maven.org 中央仓库

```
1 <repository>
2   <id>central-repos1</id>
3   <name>Central Repository 2</name>
4   <url>http://repo1.maven.org/maven2/</url>
5 </repository>
```

## 6、alfresco.com 中央仓库（首推3）

```
1 <repository>
2   <id>activiti-repos</id>
3   <name>Activiti Repository</name>
4   <url>https://maven.alfresco.com/nexus/content/groups/public</url>
5 </repository>
```

## 7、oschina 中央仓库（需要翻墙）

```
1 <repository>
2   <id>oschina-repos</id>
3   <name>Oschina Releases</name>
4
5 </repository>
```

## 8、oschina thinkgem 中央仓库（需要翻墙）

```
1 <repository>
2   <id>thinkgem-repos</id>
3   <name>ThinkGem Repository</name>
4   <url>http://git.oschina.net/thinkgem/repos/raw/master</url>
5 </repository>
```

## 9、java.net 中央仓库（需要翻墙）

```
1 <repository>
2   <id>java-repos</id>
3   <name>Java Repository</name>
4   <url>http://download.java.net/maven/2/</url>
5 </repository>
```

## 10、github.com 中央仓库（需要翻墙）

```
1 </repository>
2 <id>thinkgem-repos</id>
3 <name>ThinkGem Repository 2</name>
4 <url>https://raw.github.com/thinkgem/repository/master</url>
5 </repository>
```

## 11、ibiblio 镜像地址（这个也比较快）

```
1 <mirror>
2   <id>ibiblio</id>
3   <name>Mirror from Maven ibiblio</name>
4   <url>http://mirrors.ibiblio.org/pub/mirrors/maven2/</url>
5   <mirrorOf>central</mirrorOf>
6 </mirror>
```

## 12、repo1.maven.apache.org 镜像地址

```
1 <mirror>
2   <id>central</id>
3   <name>Maven Repository Switchboard</name>
4   <url>http://repo1.maven.apache.org/maven2/</url>
5   <mirrorOf>central</mirrorOf>
6 </mirror>
```

## 13、repo2 镜像地址

```
1 <mirror>
2   <id>repo2</id>
3   <name>Mirror from Maven Repo2</name>
4   <url>http://repo2.maven.org/maven2/</url>
5   <mirrorOf>central</mirrorOf>
6 </mirror>
```

## 14、UK 镜像地址

```
1 <mirror>
2   <id>uk central</id>
3   <name>Mirror from UK</name>
4   <url>http://uk.maven.org/maven2/</url>
5   <mirrorOf>central</mirrorOf>
6 </mirror>
```

## 15、JBoos 镜像地址

```
1 <mirror>
2   <id>jboss-public-repository-group</id>
3   <name>jboss-public-repository-group</name>
4   <url>http://repository.jboss.org/nexus/content/groups/public</url>
5   <mirrorOf>central</mirrorOf>
6 </mirror>
```

## 16、Google 镜像地址

```
1 <mirror>
2   <id>google</id>
3   <name>google maven</name>
4   <url>https://maven.google.com/</url>
5   <mirrorOf>central</mirrorOf>
6 </mirror>
```

## 17、Maven china镜像地址

```
1 <mirror>
2   <id>maven.net.cn</id>
3   <name>Mirror from Maven in china</name>
4   <url>http://maven.net.cn/content/groups/public/</url>
5   <mirrorOf>central</mirrorOf>
6 </mirror>
```

## settings.xml配置

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0 http://maven.apache.org/xsd/settings-1.0.0.xsd">
5
6   <!-- 默认的值是${user.home}/.m2/repository -->
7   <localRepository>E:\apache\repository</localRepository>
8
9   <!-- 如果Maven要试图与用户交互来得到输入就设置为true，否则就设置为false，默认为true。 -->
10  <interactiveMode>true</interactiveMode>
11
12  <!-- 如果Maven使用${user.home}/.m2/plugin-registry.xml来管理plugin的版本，就设置为true，默认为false -->
13  <usePluginRegistry>false</usePluginRegistry>
14
15  <!-- 如果构建系统要在离线模式下工作，设置为true，默认为false。 如果构建服务器因为网络故障或者安全问题 -->
16  <offline>false</offline>
17
18  <mirrors>
```

```

19     <mirror>
20         <id>aliyunmaven</id>
21         <mirrorOf>*</mirrorOf><!--*代表所有的jar包都到阿里云下载-->
22         <!--<mirrorOf>central</mirrorOf>--><!--central代表只有中央仓库的jar包才到阿里云下载-->
23         <name>阿里云公共仓库</name>
24         <url>https://maven.aliyun.com/repository/public</url>

```

```

26
27     <!--老版阿里云公共仓库-->
28     <!--
29     <mirror>
30         <id>nexus-aliyun</id>
31         <mirrorOf>central</mirrorOf>
32         <name>Nexus aliyun</name>
33         <url>http://maven.aliyun.com/nexus/content/groups/public/</url>
34     </mirror>
35     -->
36
37     <!-- 中央仓库在中国的镜像 -->
38     <mirror>
39         <id>maven.net.cn</id>
40         <name>Mirror from Maven in china</name>
41         <url>http://maven.net.cn/content/groups/public/</url>
42         <mirrorOf>central</mirrorOf>
43     </mirror>
44 </mirrors>
45
46
47
48
49     <!-- settings.xml中的profile是pom.xml中的profile的简洁形式。 它包含了激活(activation)，仓库(repository)
50     profile元素仅包含这四个元素是因为他们涉及到整个的构建系统，而不是个别的POM配置。 如果settings.xml
51     <profiles>
52         <profile>
53             <id>default</id>

```

```

55         <activeByDefault>true</activeByDefault>
56         <jdk>1.8</jdk>
57     </activation>
58     <!-- 远程仓库-->
59     <repositories>
60         <!-- 阿里云远程仓库-->
61         <repository>
62             <id>aliyun</id>
63             <name>aliyun maven Repository</name>
64             <url>https://maven.aliyun.com/repository/public</url>
65             <releases>
66                 <enabled>true</enabled>
67             </releases>
68             <snapshots>
69                 <enabled>false</enabled>
70             </snapshots>
71         </repository>
72         <repository>
73             <id>spring-milestone</id>
74             <name>Spring Milestone Repository</name>

```

```

75         <url>http://repo.spring.io/milestone</url>
76     </releases>
77     <enabled>true</enabled>
78 </releases>
79 <snapshots>
80     <enabled>false</enabled>
81 </snapshots>
82 <layout>default</layout>
83 </repository>
84 <repository>
85     <id>spring-snapshot</id>
86     <name>Spring Snapshot Repository</name>
87     <url>http://repo.spring.io/snapshot</url>
88     <releases>
89         <enabled>false</enabled>
90     </releases>
91     <snapshots>
92         <enabled>true</enabled>
93     </snapshots>
94     <layout>default</layout>
95 </repository>
96 </repositories>
97 <!-- 镜像，添加了会先从阿里云仓库下载-->
98 <pluginRepositories>
99     <pluginRepository>
100         <id>aliyun-plugin</id>
101         <url>https://maven.aliyun.com/repository/public</url>
102         <releases>
103             <enabled>true</enabled>
104         </releases>
105         <snapshots>
106             <enabled>false</enabled>
107         </snapshots>
108     </pluginRepository>
109 </pluginRepositories>
110 </profile>

```

```

112 <!-- activations是profile的关键，就像POM中的profiles，profile的能力在于它在特定情况下可以修改一些
113      而这些情况是通过activation来指定的。 -->
114 <!-- <activeProfiles/> -->
115
116 </settings>

```

参考：

<https://blog.csdn.net/wtl1992/article/details/121314010>

<http://t.zoukankan.com/larryzeal-p-6149086.html>