

Configuring multiple repositories

[Configuring JMS resources](#)

[Configuring services](#)

The advantage of having multiple repositories is that they can reside on different server instances. You can create multiple repositories on different server instances, deploy a single repository on a single server instance, and deploy applications on multiple repositories on a single server instance.

This document is organized into the following sections:

- [Creating multiple repositories on a single server instance](#)
 - [Creating a repository besides the default repository](#)
 - [Deploying an application to the repository](#)
- [Creating multiple repositories on multiple server instances](#)
 - [Creating a repository for server instance2](#)
 - [Deploying a sample application to and undeploying it from the repository](#)

As an alternative to the steps provided in this section, there is a **server-repo** plugin available that can perform this configuration for you. Visit the [Samples](#) section for further details.

Creating multiple repositories on a single server instance

You can create multiple repositories on Geronimo. You can add a second repository to deploy your applications and leave Geronimo in its default repository.

Creating a repository besides the default repository

To add a second repository besides the default repository, take the following steps:

1. Start Geronimo.
2. Create a directory `repo2` under `<geronimo_home>`, for example, `<geronimo_home>/repo2`.
3. Create a file `repo2.xml` under the `<geronimo_home>/repo2` directory, for example:

repo2.xml

```
<module xmlns="http://geronimo.apache.org/xml/ns/deployment-1.2">
  <environment>
    <moduleId>
      <groupId>org.example.configs</groupId>
      <artifactId>myrepo</artifactId>
      <version>2.2</version>
      <type>car</type>
    </moduleId>
    <dependencies>
      <dependency>
        <groupId>org.apache.geronimo.framework</groupId>
        <artifactId>j2ee-system</artifactId>
        <version>2.2</version>
        <type>car</type>
      </dependency>
    </dependencies>
    <hidden-classes/>
    <non-overridable-classes/>
  </environment>
  <gbean name="Repo2" class="org.apache.geronimo.system.repository.Maven2Repository">
    <attribute name="root">repo2</attribute>
    <attribute name="resolveToServer">false</attribute>
    <reference name="ServerInfo">
      <name>ServerInfo</name>
    </reference>
  </gbean>
  <gbean name="Local2" class="org.apache.geronimo.system.configuration.RepositoryConfigurationStore">
    <reference name="Repository">
      <name>Repo2</name>
    </reference>
  </gbean>
</module>
```

4. Deploy the `repo2.xml` from the command prompt:

```
deploy(.bat|.sh) deploy <Geronimo_HOME>/repo2/repo2.xml
```

To verify that the repository is successfully deployed, you can list the targets on the current server via the command:
`deploy(.bat|.sh) list-targets`, and you can see the following information:

```
Available Targets:  
org.apache.geronimo.framework/j2ee-system/2.2/car?ServiceModule=org.apache.geronimo.framework/j2ee-system/2.2  
/car,j2eeType=ConfigurationStore,name=Local  
org.example.configs/myrepo/2.2/car?ServiceModule=org.example.configs/myrepo/2.2/car,  
j2eeType=ConfigurationStore,name=Local2
```

Deploying an application to the repository

Here is an example of deploying a sample jsp application to the repository repo2.

1. List the targets on the current server via the command: `deploy(.bat|.sh) list-targets`, and you can see the following information:

```
Available Targets:  
org.example.configs/myrepo/2.2/car?ServiceModule=org.example.configs/myrepo/2.2/car,  
j2eeType=ConfigurationStore,name=Local2  
org.apache.geronimo.configs/j2ee-system/2.2/car?ServiceModule=org.apache.geronimo.configs/j2ee-system/2.2  
/car,j2eeType=ConfigurationStore,name=Local
```

2. Set the environment variable REPO2:

On a Windows system `set REPO2=org.example.configs/myrepo/2.2/car?ServiceModule=org.example.configs/myrepo/2.2/car,
j2eeType=ConfigurationStore,name=Local2`

On a non-Windows system `export REPO2=org.example.configs/myrepo/2.2/car?ServiceModule=org.example.configs/myrepo/2.2/car,
j2eeType=ConfigurationStore,name=Local2`

3. Deploy the application to the repository repo2 via the following deploy (.bat or .sh) command:

On a Windows system `deploy.bat deploy --targets %REPO2% <SAMPLE_HOME>/applications/geronimo-jsp-examples/target/geronimo-jsp-examples-2.2.war`

On a non-Windows system `deploy.sh deploy --targets $REPO2 <SAMPLE_HOME>/applications/geronimo-jsp-examples/target/geronimo-jsp-examples-2.2.war`

where `<SAMPLE_HOME>` is the directory of your samples. Then, the jsp example is deployed to the new repository repo2. The following message is displayed:

```
Deployed  
org.apache.geronimo.applications.examples/geronimo-jsp-examples/2.2/war  
@ /jsp-examples
```

4. Access the application via `http://host:port/jsp-examples`.

To undeploy the application from the repository repo2, run the following deploy (.bat or .sh) command:

On a Windows system `deploy.bat undeploy "%REPO2%|org.apache.geronimo.applications.examples/geronimo-jsp-examples/2.2/war"`

On a non-Windows system `deploy.sh undeploy "$REPO2|org.apache.geronimo.applications.examples/geronimo-jsp-examples/2.2/war"`

Creating multiple repositories on multiple server instances

You first need to create several server instances, and start the server instances. For information about creating server instances, see [Running Multiple Geronimo Instances](#). Then you need to create several repositories on these server instances. You can test the repositories by deploying an application to it and undeploying the application from them.

Creating a repository for server instance2

After you create more than one server instance, you can create repositories and deploy the repositories to different server instances. For information about how to create and deploy a repository on a server instance, for example, create repo22 and deploy it on server2.

1. Create a directory `repo22` under `<geronimo_home>`, for example, `<geronimo_home>/repo22`.
2. Create a file `repo22.xml` under the `<geronimo_home>/repo22` directory. For detailed information about the file, see [creating multiple repositories](#).
3. Deploy the `repo22.xml` by running the following command from the command prompt:
`deploy(.bat|.sh) -port instance2_RMI_port deploy <geronimo_home>/server2/repo22/repo22.xml`

You can follow the same steps to create repositories for server instance3.

Deploying a sample application to and undeploying it from the repository

To test the repository that you create, you can deploy a sample application to and undeploying it from the repository.

1. Set the environment variable repo22 for the repository repo22 that you have previously created.
2. Deploy the application to the repository repo2 via the following deploy (.bat or .sh) command:
On a Windows system **deploy.bat -port instance2_RMI_port deploy --targets %repo22% <SAMPLE_HOME>/applications/geronimo-jsp-examples/target/geronimo-jsp-examples-2.2.war**
On a non-Windows system **deploy.sh -port instance2_RMI_port deploy --targets \$repo22 <SAMPLE_HOME>/applications/geronimo-jsp-examples/target/geronimo-jsp-examples-2.2.war**
3. Undeploy the jsp application from repo22 repository via the following command:
On a Windows system **deploy.bat -port instance2_RMI_port undeploy %repo22%|org.apache.geronimo.applications.examples/geronimo-jsp-examples/2.2/war**
On a non-Windows system **deploy.sh -port instance2_RMI_port undeploy \$repo22|org.apache.geronimo.applications.examples/geronimo-jsp-examples/2.2/war**

You can follow the same steps to deploy applications to and undeploy applications from repositories on server instance3.