

Configuring Repositories

Overview

Artifactory hosts three types of repository:

- [Local](#)
- [Remote](#)
- [Virtual](#)

Local and remote repositories are true physical repositories, while a virtual repository is actually an aggregation of them used to create controlled domains for search and resolution of artifacts.

To configure repositories, in the **Admin** module, select **Repositories**.

Repositories can be created, deleted, edited, ordered and aggregated.

Single Package Type

When creating any repository, you must specify its package type; this is a fundamental characteristic of the repository and can not be changed later. Once the repository type is set, Artifactory will index artifacts and calculate the corresponding metadata for every package uploaded which optimizes performance when resolving artifacts. Note that virtual repositories can only include repositories of the same type.

Wrong Package Type

While Artifactory will not prevent you from uploading a package of the wrong type to a repository, we strongly recommend maintaining consistency between the repository type and packages you upload.

if you do upload packages of the wrong type to a repository, Artifactory will not index the package or update the metadata for the repository.

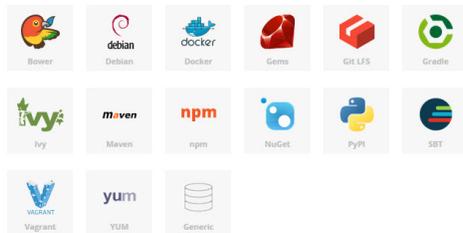
Page Contents

- [Overview](#)
- [Single Package Type](#)
 - [Generic Repositories](#)
- [Local Repositories](#)
- [Remote Repositories](#)
- [Virtual Repositories](#)
 - [The Default Virtual Repository \(Deprecated\)](#)
 - [Virtual Resolution Order](#)
- [General Resolution Order](#)

Read More

- [Common Settings](#)
- [Local Repositories](#)
- [Remote Repositories](#)
- [Smart Remote Repositories](#)
- [Virtual Repositories](#)

Select Package Type



Generic Repositories

You may define a repository as **Generic** in which case it has no particular type, and you may upload packages of any type. Generic repositories do not maintain separate package indexes. For using a client associated with a specific package type (e.g. yum, gem) you should create a matching repository.

Local Repositories

Local repositories are physical, locally-managed repositories into which you can deploy artifacts.

Artifacts in a local repository can be accessed directly using the following URL:

```
http://<host>:<port>/artifactory/<local-repository-name>/<artifact-path>
```

Artifactory is deployed with a number of pre-configured local repositories which can be used for internal and external releases, snapshots and plugins.

For full details on configuring local repositories, please refer to [Local Repositories](#).

Remote Repositories

A remote repository serves as a caching proxy for a repository managed at a remote URL (which may itself be another Artifactory remote repository).

Artifacts are stored and updated in remote repositories according to various configuration parameters that control the caching and proxying behavior. You can remove artifacts from a remote repository cache but you cannot actually deploy a new artifact into a remote repository.

Artifacts in a remote repository can be accessed directly using the following URL:

```
http://<host>:<port>/artifactory/<remote-repository-name>/<artifact-path>
```

This URL will fetch a remote artifact to the cache if it has not yet been stored.

In some cases it is useful to directly access artifacts that are already stored in the cache (for example to avoid remote update checks).

To directly access artifacts that are already stored in the cache you can use the following URL:

```
http://<host>:<port>/artifactory/<remote-repository-name>-cache/<artifact-path>
```

Artifactory is deployed with a number of pre-configured, remote repositories which are in common use. Of course you can change these according to the needs of your organization.



Proxy vs. Mirror

A remote repository acts as a **proxy** not as a mirror. Artifacts are not pre-fetched to a remote repository cache. They are only fetched and stored *on demand* when requested by a client. Therefore, a remote repository should not contain any artifacts in its cache immediately after creation. Artifacts will only be fetched to the cache once clients start working with the remote repository and issuing requests.

For full details on configuring remote repositories please refer to [Remote Repositories](#).

Virtual Repositories

A virtual repository (or "repository group") aggregates several repositories with the same package type under a common URL.



Generic Virtual Repositories

By their nature, Virtual Repositories whose package type has been specified as **Generic** can aggregate repositories of any type, however generic virtual repositories do not maintain any metadata

The Default Virtual Repository (Deprecated)

Artifactory offers an option to use a global virtual, which contains all local and remote repositories.

By default this option is disabled, to enable the Default Virtual Repository edit the 'artifactory.system.properties' located at \$ARTIFACTORY_HOME/etc and set the following flag to *false*:

```
## Disable the download access to the global 'repo'  
artifactory.repo.global.disabled=false
```

This change requires you restart your Artifactory service.

Once enabled the repository is available at:

```
http://<hostname>:<port>/artifactory/repo
```

Virtual Resolution Order

When an artifact is requested from a virtual repository, the order in which repositories are searched or resolved is local repositories first, then remote repository caches, and finally remote repositories themselves.

Within each of these, the order by which repositories are queried is determined by the order in which they are listed in the configuration as described in [General Resolution Order](#) below.

For a virtual repository, you can see the effective search and resolution order in the **Included Repositories** list view in the **Basic** settings tab. This is particularly helpful when nesting virtual repositories. For more details on configuring a virtual repository please refer to [Virtual Repositories](#).

General Resolution Order

You can set the order in which repositories of each type (local, remote and virtual) are searched and resolved by simply ordering them accordingly within the corresponding section of the **Configure Repositories** page. To set the order you need to add the repositories to the list of selected repositories in the order in which they should be searched to resolve artifacts.

The order in which repositories are searched is also affected by additional factors such as security privileges, include/exclude patterns and policies for handling snapshots and releases.