Public Repository

Introduction

ChemAxon provides a public repository at https://hub.chemaxon.com for our products and modules. It is designed for integrators and resellers but is available for any registered ChemAxon user as well. How to acquire the required credentials in order to use this repository are described below.

Create your ChemAxon Account

If you do not already have a ChemAxon Account, you may create it via filling the form at the https://accounts.chemaxon.com/register page. Your e-mail address will serve as the username when setting up the repository.

Generate your API key

To generate your API key simply visit your settings page at https://accounts.chemaxon.com/my/settings. Under the ‘Public Repository’ section, you will find a button to ‘Generate API Key’ or your exact key having generated one already.

Configure your project

Debian

First, download the public GPG key:

```
wget -qO - https://<E-MAIL>:<API-KEY>@hub.chemaxon.com/artifactory/api
/gpg/key/public | apt-key add -;
```

To use the Artifactory repository to install a Debian package you need to add it to your sources. list file. You can do that using the following command:

```
sudo echo 'deb https://<E-MAIL>:<API-KEY>@hub.chemaxon.com/artifactory
/cxn-deb-release-local all main' >> /etc/apt/sources.list
```

Using the apt command will install the package:

```
sudo apt install <PACKAGE>
```

Search for other packages:

```
apt search chemaxon
```

Red Hat

First, you need to add the Artifactory repository to your YUM repositories. Edit the following file with your editor of choice:

```
sudo vim /etc/yum.repos.d/artifactory.repo
```

And insert the following to the file:

```
[Artifactory]
name=ChemAxon Artifactory
baseurl=https://<E-MAIL>:<API-KEY>@hub.chemaxon.com/artifactory/cxn-
rpm-release-local
enabled=1
gpgcheck=0
#Optional - if you have GPG signing keys installed, use the below
flags to verify the repository metadata signature:
gpgkey=https://<E-MAIL>:<API-KEY>@hub.chemaxon.com/artifactory/api/gpg
/key/public
repo_gpgcheck=1
```

Install your package with the command:

```
sudo yum install <PACKAGE>
```

Search for other packages:

```
yum search chemaxon
```

Docker

First, you need to login to the repository:

```
echo <API-KEY> | docker login -u <E-MAIL> --password-stdin hub.chemaxon.com
```

Alternatively, you can paste the following into the `~/.docker/config.json` file (in Windows `%USERPROFILE%/.docker/config.json`):

```
{
  "auths": {
    "hub.chemaxon.com": {
      "auth": <BASE64(<E-MAIL>:<API-KEY>)>
    }
  }
}
```

After that, you can pull the images:

```
docker pull hub.chemaxon.com/<IMAGE>:<TAG>
```

Browse Docker Tags

Though Docker itself doesn't have a convenient way to browse image tags, there are workarounds like Node Package Manager (8.0+).

```
npx docker-browse tags <IMAGE> -r hub.chemaxon.com
```

NPM

First, you need to store your credentials and repository in your .npmrc file:

```
curl -o ~/.npmrc -u<E-MAIL>:<API-KEY> https://hub.chemaxon.com
/artifactory/api/npm/auth
```

For your npm command line client to work with Artifactory, you first need to set the default npm registry with an Artifactory npm repository using the following command:

```
npm config set registry https://hub.chemaxon.com/artifactory/api/npm
/npm/
```

```
Alternatively, you can paste the following into the ~/.npmrc file (in
Windows %USERPROFILE%/.npmrc):
_auth=<BASE64(<E-MAIL>:<API-KEY>)>
always-auth=true
email=<E-MAIL>
registry=https://hub.chemaxon.com/artifactory/api/npm/npm/
```

To install the packages run:

```
npm install <PACKAGE>
```

Gradle

If you would like to use Gradle as your build tool, you can follow these instructions to access ChemAxon’s Maven repository. Or you can use the quickstart project as an example.

In your Gradle project, create (or edit if already exists) a `gradle.properties` file and add these contents. You can also move this file outside your project, into <user home>/.gradle directory.

```
chemaxonRepositoryUser=<E-MAIL>
chemaxonRepositoryPassword=<API-KEY>
chemaxonRepositoryUrl=https://hub.chemaxon.com/artifactory
```

In the `build.gradle` file, add our Maven repository in the repositories block. For example:

```
repositories {
  maven {
     url "${chemaxonRepositoryUrl}/libs-release"
     credentials {
         username = chemaxonRepositoryUser
         password = chemaxonRepositoryPassword
     }
  }
}
```

After applying these settings, you can add ChemAxon modules as dependencies.

If you had a problem with resolving the ojdbc8 Oracle JDBC driver but you do not want to use Oracle then omitting the driver solves the problem. To omit ojdbc8 add this setting to the `build.gradle` file:

```
configurations.all {
    exclude group: 'com.oracle.jdbc', module: 'ojdbc8'
}
```

Quickstart Projects

Gradle

A simple, self-contained quickstart project example using Gradle:

MSketchGradleQuickstart.zip

Maven

A simple, self-contained quickstart project example using Apache Maven:

MSketchMavenQuickstart.zip

Top Level Modules

To integrate a product, you only need to add its top-level module as a dependency, all required modules will be downloaded transitively (with the exception of the module **naming** which also needs to be added separately).

Product	Top-level module name in the repository
Marvin	marvin-app
JChem	jchem-main

Example Projects

JKlustor Web Example

An open-source API usage example for JKlustor, which uses ChemAxon's Maven Repository. It is available on GitHub: https://github.com/ChemAxon/jklustor-web-example/