Setting Up New Connector Project

Create the Project

Create a source code repository for a new connector. We strongly prefer if you use Github to maintain connector source code, but any reasonable way will do. E.g. connectors maintained under the Polygon project are maintained in the Evolveum area on Github. The naming convention that we use is `connector-targetsystem`.

Choose a license. We recommend to use Apache 2.0 license. But any reasonable license will do.

Choose a maven group ID and artifact ID.

- Group IDs are usually based on Java package names which are in turn usually based on DNS names, We use `com.evolveum.polygon`. But you are free to use your own.
- Artifact IDs are short component names. We use the same name as name of the repository, e.g. `connector-gitlab`.

Use our Maven archetype to create an empty project structure.

- Get the polygon framework source code and install it:

```
git clone https://github.com/Evolveum/polygon.git
cd polygon
mvn clean install
```

- Generate your project structure using polygon archetype:

```
cd your-project-folder
mvn archetype:generate -DarchetypeGroupId=com.evolveum.polygon -DarchetypeArtifactId=basic-connector-archetype -
DarchetypeVersion=1.4.0.49 -DgroupId=your.group.id -DartifactId=your-artifact-id
```

The archetype will ask several questions such as these:

```
Define value for property 'version':  1.0-SNAPSHOT: : 1.4.1.0-SNAPSHOT
[INFO] Using property: package = com.evolveum.polygon
Define value for property 'connectorName': : Foo
Confirm properties configuration:
groupId: com.evolveum.polygon
artifactId: connector-ldap
version: 1.4.1.0-SNAPSHOT
package: com.evolveum.polygon
connectorName: Foo
 Y: :
```

Checkout the empty repository before the you use the artifact and generate the files directly it in. Or use the generated directory for an initial import. Or simply move the generated files where you need them. It all depends on the source control system that you are using and your development process.

The resulting file structure should look like this:

```
pom.xml
src
  main
    ...
  test
    ...
LICENSE (optional, but this is a good practice)
README  (optional, but this is a good practice)
```

Do not forget to add the generated files under source control, e.g. for git:

```
git add pom.xml src
```

Now you can import the project into your IDE and start the development.