

OAM Repositories

- [Overview](#)
 - [Structure and Jenkins jobs](#)
 - [Artifacts groupId "org.o-ran-sc.oam.features"](#).
- [Directory oam:features containing specific devicemanagers](#)
 - [Artifact ids](#)

Overview

Repositories managed by OAM project

- [oam](#) - main repo
- [oam-nf-oam-adopter](#) - Network Function Adopter for model and protocol conversion according to O-RAN OAM specifications.
- [oam-tr069-adapter](#) - Converts BBF-TR069 oam SOAP interface into NETCONF/YANG
- [scp/oam/modeling](#) - yang data models and yang-tools generated classes

Repositories are used to provide osgi bundles and docker images for **nonrtic-o1-controller**.

- feature parent for the feature bundles and distribution.
- docker images for nonrtic-o1-controller
- feature bundles for specific devicemanagers
- model bundles with classes for generated classes from yang files, representing specific standards

Setup tasks/sub-tasks of jenkins are bundled via

[OAM-83](#) - Getting issue details...

STATUS

Target environments

- ONAP Honolulu: Maven 3.8, Java 11
- ONAP Istanbul: Maven 3.8 Java 11

Structure and Jenkins jobs

The following sub structure is used:

| repo:path | Artifact group | Build results | description | Jenkins jobs |
|----------------------------|--|---------------------------------|--------------------------------------|---|
| oam:distribution/ | | docker images nexus | Source for image creation | ci-management/jjb/oam/oam-distribution.yaml |
| oam:docs/ | | Documentation readTheDocs | readTheDocs source | |
| oam:features/ | org.o-ran-sc.oam.features | feature bundles nexus | Feature bundles | ci-management/jjb/oam/oam-features.yaml oam-features-maven-verify-master-mvn35-openjdk8 oam-features-maven-merge-master oam-features-maven-stage-master oam-features-release-merge oam-features-release-verify |
| oam:parent/ | | feature parent bundles nexus | Parents depending on ONAP Parents | ci-management/jjb/oam/oam-parent.yaml |
| oam:info.yaml | | | | ci-management/jjb/oam/oam.yaml |
| scp/oam/modeling:features/ | org.o-ran-sc.scp.oam.modeling.features | feature bundles nexus | yang model generated code | |
| scp/oam/modeling:info.yaml | | | | ci-management/jjb/scp-oam-modeling/scp-oam-modeling.yaml |

Artifacts groupId "org.o-ran-sc.oam.features".

List of all artifacts as provided under this group in repository layout

```
org.o-ran-sc.oam.features
devicemanager
  devicemanager-oran-ru-fh-feature
  devicemanager-oran-ru-fh-installer
  devicemanager-oran-ru-fh-model
  devicemanager-oran-ru-fh-provider
  devicemanager-oran-ru-fh-top
  devicemanager-xran-ru-fh-feature
  devicemanager-xran-ru-fh-installer
  devicemanager-xran-ru-fh-model
  devicemanager-xran-ru-fh-provider
  devicemanager-xran-ru-fh-top
features-top
  0.7.1-SNAPSHOT
  maven-metadata-local.xml
parent
binding-parent
odlparent-lite
parent-top
single-feature-parent
```

Directory oam:features containing specific devicemanagers

devicemanager - top level for devicemanager

- o-ran - Open RAN standard
 - ru-fh - device related radio unit - front haule

- x-ran - X RAN standard
 - ru-fh - radio unit - front haule

- tr069 - TR069 standard

Artifact ids

Using ONAP Frankfurt

Example with devicemanager xran/ru-fh for top pom file.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <parent>
    <groupId>org.onap.ccsdk.parent</groupId>
    <artifactId>odlparent-lite</artifactId>
    <version>1.5.2-SNAPSHOT</version>
    <relativePath/>
  </parent>

  <groupId>org.o-ran-sc.oam.features.devicemanager</groupId>
  <artifactId>devicemanager-oran-ru-fh-top</artifactId>
  <version>0.7.1-SNAPSHOT</version>
  <packaging>pom</packaging>

  <name>o-ran-sc-features :: ${project.artifactId}</name>

  <properties>
    <feature-name>devicemanager-oran-ru-fh</feature-name>
  </properties>

  <modules>
    <module>model</module>
    <module>provider</module>
    <module>feature</module>
    <module>installer</module>
  </modules>

<
/project>
```