

Teaching brigade, ONF



Developing ONOS Apps

Andrea Campanella, ONF, USA

Abdulhalim Dandoush, ESME-SUDRIA, France

Lefteris Manassakis, ICS FORTH, Greece

ONOS Build, SAMSUNG Seoul

September 20, 2017

Slides Links



Slides : <https://goo.gl/fB5yfj>

Outline



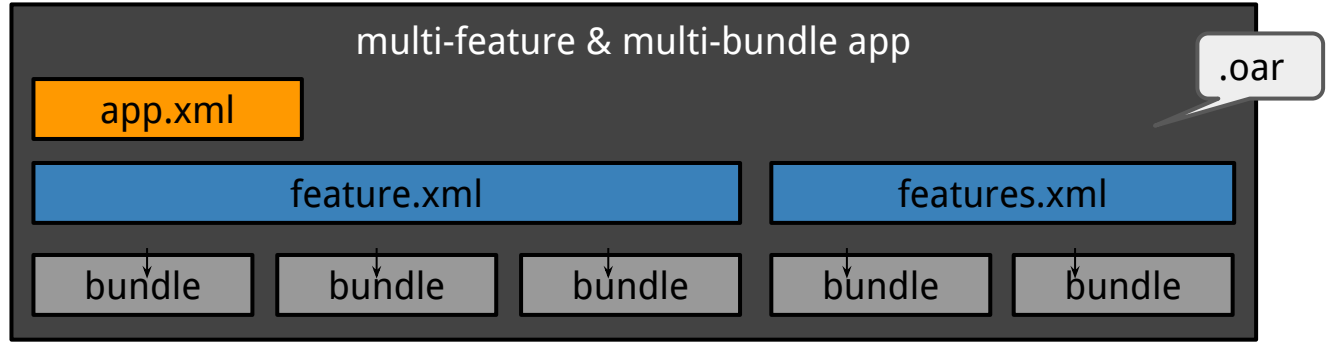
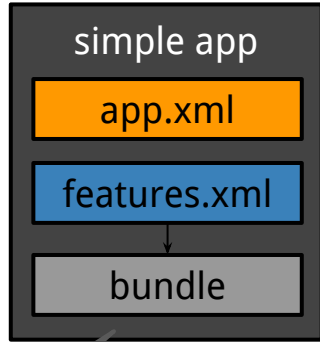
ONOS Applications

OSGi bundles, Karaf features, ONOS apps & OAR files
application lifecycle - install, activate, deactivate, uninstall
ONOS CLI & GUI

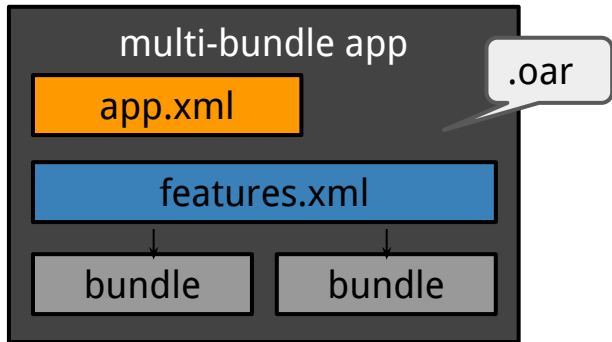
Developing ONOS apps

use `mvn archetype:generate` and `onos-create-app`
archetype overlays for CLI and UI - generated & pre-canned
iterative and demonstrating use of `onos-app` tool or GUI

Bundles, Features & ONOS Apps



.oar



Bundles, Features & ONOS Apps



- **Apps are delivered via ONOS App Archive (.oar) files**
 - OAR is a JAR with `app.xml`, `features.xml` and bundle artifacts
 - `onos-maven-plugin` generates an `*.oar` file as part of Maven build
- **Apps are managed on the entire ONOS cluster**
 - via REST API: `GET | POST | DELETE /onos/v1/applications`
 - via shell tool: `onos-app {install | activate | deactivate | uninstall}`
 - via CLI: `onos:app {install | activate | deactivate | uninstall}`
 - via GUI Applications view
- **Back-end installation and activation is done via normal feature & bundle services**

Developing ONOS apps



- **Maven archetypes**

- `onos-api-archetype` - basis for an app Java API bundle
- `onos-bundle-archetype` - basis for an ONOS bundle or an app
- `onos-cli-archetype` - overlay for apps with CLI extensions
- `onos-ui-archetype` - overlay for apps with GUI extensions
- `onos-uitab-archetype` - basis for an app Java API bundle
- `onos-uitopo-archetype` - basis for an app Java API bundle

- **Creates a working minimal project module**

- via `onos-create-app` shell tool
- convenience wrapper for `mvn archetype:generate`

One Ping Only ONOS app



Let's develop an app of our own...

One Ping Only ONOS app



Environment Setup



Bash_Profile setup:

```
vi ~/.bashrc
export ONOS_ROOT=~/onos
source $ONOS_ROOT/tools/dev/bash_profile
```

Maven:

```
sudo apt install maven
```

ONOS Maven Artifacts:

```
onos-buck-publish-local
```

Minimal ONOS app project



Make directory for app

```
$ cd ~
```

```
$ mkdir oneping
```

Create minimal app project via **onos-create-app** tool

```
$ onos-create-app app org.oneping oneping 1.0-SNAPSHOT
```

Minimal ONOS app project



Import into IDE and Edit our app `pom.xml` file

Minimal ONOS app project



Build the app via Maven and deploy it via **onos-app** tool

```
$ mvn clean install -U
```

```
$ onos-app 127.0.0.1 install! target/oneping*.oar
```

Minimal ONOS app project



Clean existing topology (if any)

```
sudo mn -c
```

Start Mininet topology

```
sudo mn --controller=remote,ip=127.0.0.1,port=6633 --mac  
--topo=tree,2,2 --switch=ovsk,protocols=OpenFlow13
```

Ping in Between mininet hosts does NOT work

```
mininet> h1 ping h2
```

Code & deploy OnePingOnly app



Code our app as **OnePing.java**

<https://github.com/opennetworkinglab/onos-app-samples/tree/master/oneping>

Build the app via Maven

```
$ cd oneping  
$ mvn clean install
```

Code & deploy OnePingOnly app



Deploy the app to ONOS cluster via **onos-app** tool

```
$ onos-app 127.0.0.1 reinstall! target/oneping*.oar
```

Show one-ping-only enforcement



Issue one ping between two hosts

```
mininet> h1 ping -c 1 h2
```

```
mininet> h1 ping -c 1 h3
```

Issue another ping between the same two hosts

```
mininet> h1 ping -c 1 h2
```

Second ping fails
Messages in the logs

ONOS app overlay archetypes



Some ONOS archetypes can be used to enhance an existing project with additional capabilities

- add CLI extensions
- add GUI extensions
- others in future

Simply run **onos-create-app** tool in the same project

Sample command does not exist

```
onos> sample
```

Overlay CLI archetype



Overlay CLI additions via **onos-create-app**

```
$ onos-create-app cli org.oneping oneping 1.0-SNAPSHOT
```

Rebuild and redeploy the enhanced app

```
$ mvn clean install
```

```
$ onos-app 127.0.0.1 reinstall! target/oneping*.oar
```

Show that sample command now exists

```
onos> sample
```

Q&A



Andrea: andrea@opennetworking.org

Lefteris: manassakis@gmail.com

Abdulhalim: adandoush@gmail.com

Documentation



- [Get Started with ONOS](#)
- [Creating and deploying and ONOS App](#)
- [Creating and deploying and ONOS App \(video\)](#)
- [Template application tutorial](#)



ONOS
Open Network Operating System