

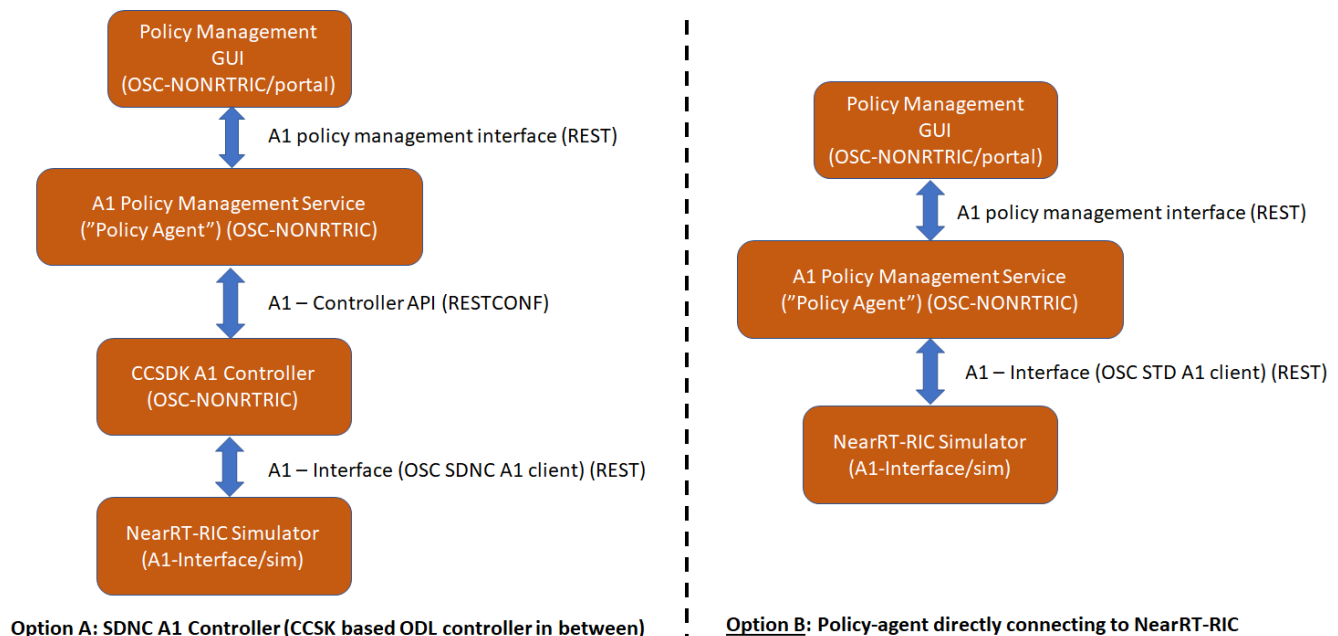
Testing End to End call in release A

This page describes how to demonstrate the entire flow from Non-Rt RIC Controlpanel -> Policy Agent -> (A1 controller in SDNC) -> Near Realtime Ric Simulator. Optionally A1 controller can be left out.

- [Overview of the End to End call](#)
- [End to End call from Dashboard to NearRT-RIC simulator](#)
 - [Preparation](#)
 - [Perform Policy Operations via Dashboard](#)
- [Call from CCSDK A1 Controller to NearRT-RIC simulator](#)
 - [Preparation](#)
 - [Perform Policy Operations via Open Daylight GUI](#)
 - [Get policy types](#)
 - [Create policy](#)
 - [Get policy instances](#)

Overview of the End to End call

Details of the architecture can be found from the following link: [Release A](#)



End to End call from Dashboard to NearRT-RIC simulator

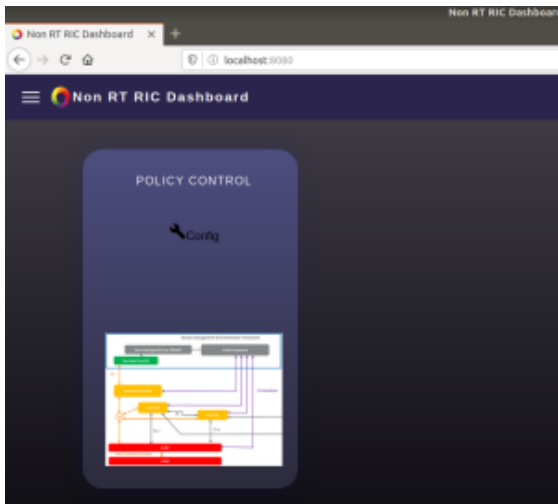
Preparation

See [Release A - Build/Run](#) for building and starting the environment.

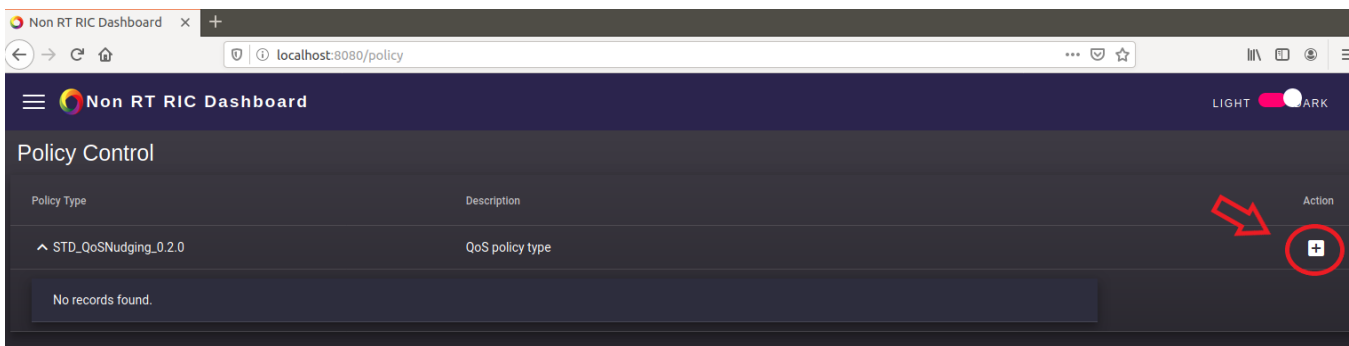
Either option A or B can be setup and all end to end actions from Dashboard should work same way.

Perform Policy Operations via Dashboard

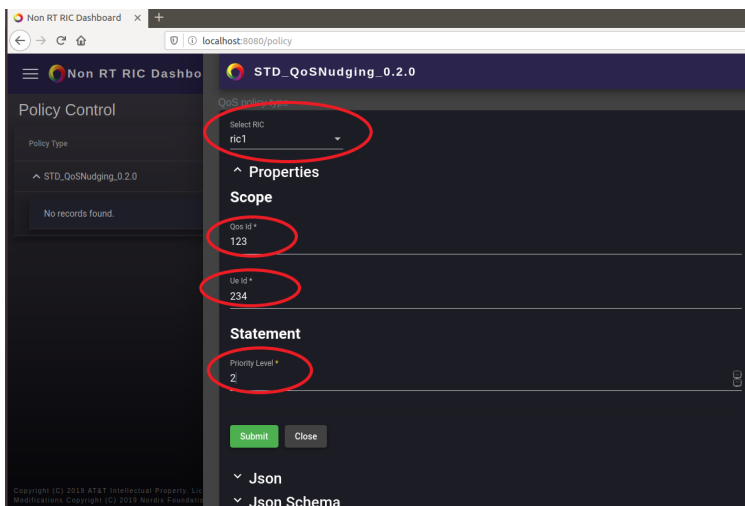
- Open Dashboard UI at <http://localhost:8080/>



- Click on Config and the screen below with STD_QoS Nudging_0.2.0 policy type should open



- Click on '+' icon to create new policy



- Choose your ric (ric1 or ric2) from drop-down menu and provide any arbitrary values for QoS Id, Ue Id and Priority

Dashboard

8b12866d-8f7d-4d99-84d2-5341e8d79600

STD_QoS Nudging_0.2.0

QoS policy type

^ Properties

Scope

QoS Id *

123

Ue Id *

234

Statement

Priority Level *

-2

Submit Close

Json

Json Schema

Policy

STD_QoS Nudging_0.2.0:8b1286

6d-8f7d-4d99-84d2-5341e8d79600

submitted

- Submit the policy (you then need to also press Close to return from creation view)
- An entry to the chosen nearRT-RIC simulator console/log should appear (see, for example, **docker logs ric1** in case ric1 policy was created):

```
172.18.0.3 - - [02/Apr/2020 07:33:57] "PUT /A1-P/v1/policies/ac3e914a-b39c-4d93-b77c-d2ebac756267?policyTypeId=STD_QoS Nudging_0.2.0 HTTP/1.1" 201 -
```

- You may repeat the creation process a few times for different rics - below we see one policy created to ric1 and two to ric2

Non RT RIC Dashboard

localhost:8080/policy

Non RT RIC Dashboard

LIGHT

ARK

Policy Control

Policy Type

Description

Action

STD_QoS Nudging_0.2.0

QoS policy type

+

Instance	Ric	Owner	Last modified	Action
fbf586cd-6839-4057-914f-4fff385f6da4	ric2	dashboard	4/1/2020, 6:37:27 PM	
ac96b6cf-12da-43a4-8d0c-89f277f2a89a	ric1	dashboard	4/1/2020, 6:37:02 PM	
0f3539c2-4977-41eb-a5fd-dcdeaeffd50c	ric2	dashboard	4/1/2020, 6:36:47 PM	

- Click the pen icon to modify any policy

Dashboard

8b12866d-8f7d-4d99-84d2-5341e8d79600

STD_QoS Nudging_0.2.0

QoS policy type

^ Properties

Scope

QoS Id *

123

Ue Id *

234

Statement

Priority Level *

3

Submit Close

Policy

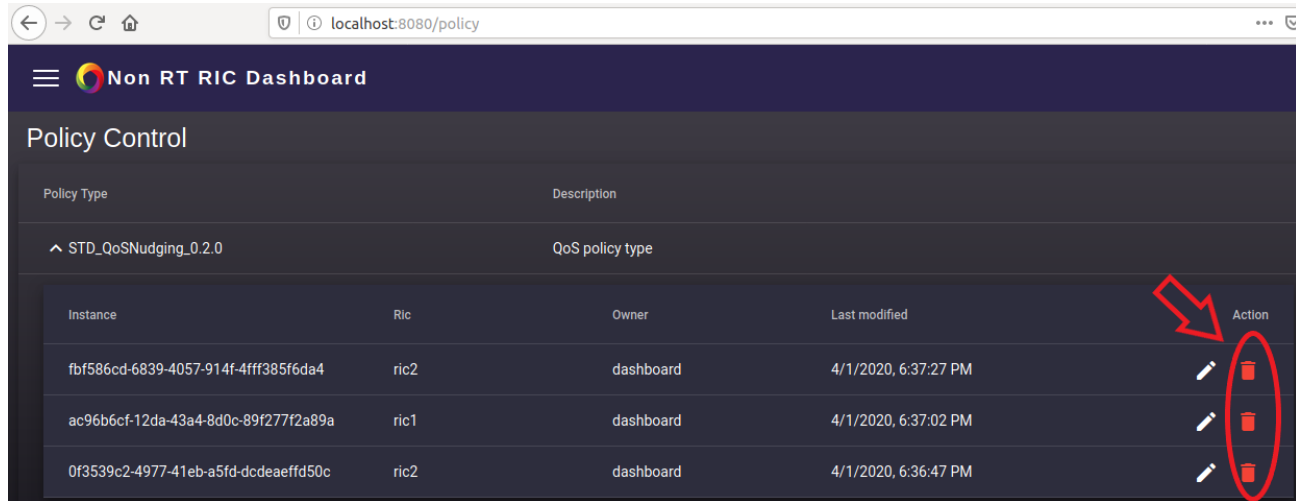
STD_QoS Nudging_0.2.0:8b1286

6d-8f7d-4d99-84d2-5341e8d79600

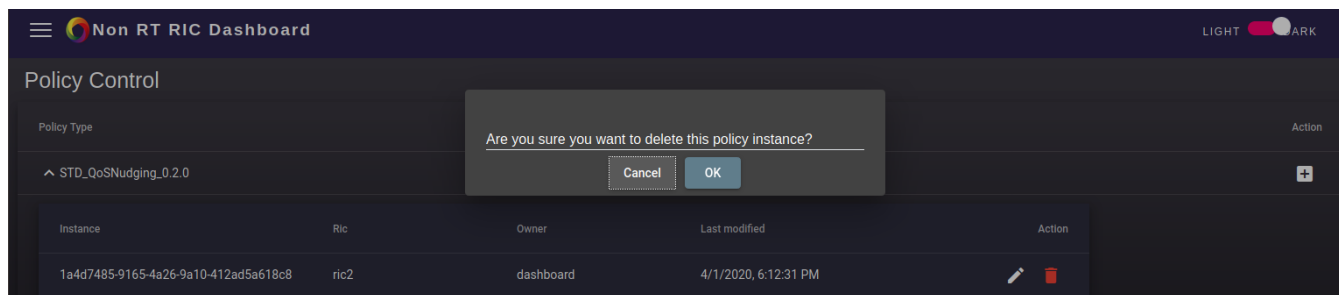
submitted

- Even though the dashboard allows changing any value, the simulator will only accept Priority changes
- Modifications are submitted in the same way as in creation, and the resulting simulator log should also be identical to the one we saw in creation

```
172.18.0.3 - - [02/Apr/2020 07:51:19] "PUT /A1-P/v1/policies/ac3e914a-b39c-4d93-b77c-d2ebac756267?policyTypeId=STD_QoSNUdging_0.2.0 HTTP/1.1" 201 -
```



- Delete any policy by clicking the red trashcan icon



- Confirm deletion and verify the log on simulator side:

```
172.18.0.3 - - [02/Apr/2020 08:02:21] "DELETE /A1-P/v1/policies/ac3e914a-b39c-4d93-b77c-d2ebac756267 HTTP/1.1" 204 -
```

Call from CCSDK A1 Controller to NearRT-RIC simulator

Limited tests to verify just OSC CCSDK A1 Controller calls towards NearRT-RIC simulator can be done by using ODL GUI. Dashboard and policy-agent are not needed in this case.

Preparation

See [Release A - Build/Run](#) for building and starting the environment. If following those instructions, just **skip the following steps**:

- [Release A - Build/Run#RunPolicy-agentDockerContainer](#)
- [Release A - Build/Run#RunNon-RT-RICDashboardDockerContainer](#)

Now only "a1-controller-container" and "sdnc_db_container" and simulators are running.

Perform Policy Operations via Open Daylight GUI

- Open GUI: [Release A - Build/Run#OpenDaylightGUI](#)
- Click Sub API:
[A1-ADAPTER-API\(2020-01-22\)](#)



OpenDaylight RestConf API Documentation

Controller Resources

Mounted Resources

Below are the list of APIs supported by the Controller.

A1-ADAPTER-API(2020-01-22)

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)

POST /restconf/operations/A1-ADAPTER-API:deletePolicy

POST /restconf/operations/A1-ADAPTER-API:getPolicyTypeIdentities

POST /restconf/operations/A1-ADAPTER-API:putPolicy

POST /restconf/operations/A1-ADAPTER-API:getPolicyType

POST /restconf/operations/A1-ADAPTER-API:getPolicyIdentities

ASDC-API(2017-02-01)

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)

DataChange(2015-05-19)

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)

Get policy types

Payload

POST /restconf/operations/A1-ADAPTER-API:getPolicyTypeIdentities

```
{
  "A1-ADAPTER-API:input": {
    "near-rt-ric-url": "http://ric1:8085/"
  }
}
```

POST /restconf/operations/A1-ADAPTER-API:getPolicyTypeIdentities

Response Class

Model | Model Schema

```
(getPolicyTypeIdentities)output-TOP {  
  A1-ADAPTER-API:output (object[(getPolicyTypeIdentities)output], optional)  
}  
(getPolicyTypeIdentities)output {  
  A1-ADAPTER-API:policy-type-id-list (array[Optional.empty], optional)  
}
```

Response Content Type application/json ▼

Parameters

Parameter	Value	Description	Parameter Type	Data Type
(getPolicyTypeIdentities)input-TOP	<pre>{ "A1-ADAPTER-API:input": { "near-rt-ric-url": "http://ric1:8085/" } }</pre> Parameter content type: application/json ▼		body	Model Model Schema <pre>(getPolicyTypeIdentities)input-TOP { A1-ADAPTER-API:input (object[(getPolicyTypeIdentities)input], optional) } (getPolicyTypeIdentities)input { A1-ADAPTER-API:near-rt-ric-url (Optional.empty, optional) }</pre>

Try it out!

[Hide Response](#)

Request URL

http://localhost:8282/restconf/operations/A1-ADAPTER-API:getPolicyTypeIdentities

Response Body

```
{  
  "output": {  
    "policy-type-id-list": [  
      "STD_QoS_Nudging_0.2.0"  
    ]  
  }  
}
```

Response Code

200

Response Headers

```
{  
  "Access-Control-Allow-Credentials": "true",  
  "Access-Control-Allow-Origin": "http://localhost:8282",  
  "Content-Length": "59",  
  "Content-Type": "application/json"  
}
```

Create policy

Payload

POST /restconf/operations/A1-ADAPTER-API:putPolicy

```
{
  "A1-ADAPTER-API:input": {
    "near-rt-ric-url": "http://ric1:8085/",
    "policy-id": "my-policy-instance1",
    "policy-type-id": "STD_QoSNUdging_0.2.0",
    "policy": '{
      "scope": {
        "qosId": "999",
        "ueId": "5001"
      },
      "statement": {
        "priorityLevel": 5
      }
    }'
  }
}
```

POST /restconf/operations/A1-ADAPTER-API:putPolicy

Response Class

Model [Model Schema](#)

```
(putPolicy)output-TOP {  
  A1-ADAPTER-API:output (object{(putPolicy)output}, optional)  
}  
(putPolicy)output {  
  A1-ADAPTER-API:returned-policy (Optional.empty, optional)  
}
```

Response Content Type application/json ▼

Parameters

Parameter	Value	Description	Parameter Type	Data Type
(putPolicy)input-TOP	<pre>{ "A1-ADAPTER-API:input": { "near-rt-ric-url": "http://rid1:8085/", "policy-id": "my-policy-instance1", "policy-type-id": "STD_QoSNUdding_0.2.0", "policy": '{ "scope": { "qosId": "999", "ueId": "5001" }, "statement": { "priorityLevel": 5 } }' } }</pre>		body	Model Model Schema <pre>(putPolicy)input-TOP { A1-ADAPTER-API:input (object{(putPolicy)input}, optional) } (putPolicy)input { A1-ADAPTER-API:near-rt- ric-url (Optional.empty, optional), A1-ADAPTER-API:policy- id (Optional.empty, optional), A1-ADAPTER-API:policy- type-id (Optional.empty, optional), A1-ADAPTER-API:policy (Optional.empty, optional) }</pre>

Parameter content type: application/json ▼

[Try it out!](#)

[Hide Response](#)

Request URL

http://localhost:8282/restconf/operations/A1-ADAPTER-API:putPolicy

Response Body

```
{  
  "output": {  
    "returned-policy": "{\n\"scope\":{\n\"qosId\":\n\"999\", \"ueId\":\n\"5001\"},\n\"statement\":{\n\"priorityLevel\":5\n}}\n"}  
}
```

Response Code

200

Response Headers

```
{  
  "Access-Control-Allow-Credentials": "true",  
  "Access-Control-Allow-Origin": "http://localhost:8282",  
  "Content-Length": "120",  
  "Content-Type": "application/json"  
}
```

Get policy instances

Payload

POST /restconf/operations/A1-ADAPTER-API:getPolicyIdentities

```
{
  "A1-ADAPTER-API:input": {
    "near-rt-ric-url": "http://ric1:8085/"
  }
}
```

POST /restconf/operations/A1-ADAPTER-API:getPolicyIdentities

Response Class

Model [Model Schema](#)

```
(getPolicyIdentities)output-TOP {
  A1-ADAPTER-API:output (object[(getPolicyIdentities)output], optional)
}
(getPolicyIdentities)output {
  A1-ADAPTER-API:policy-id-list (array[Optional.empty], optional)
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
(getPolicyIdentities)input-TOP	<pre>{ "A1-ADAPTER-API:input": { "near-rt-ric-url": "http://ric1:8085/" } }</pre> Parameter content type: application/json ▾		body	Model Model Schema (getPolicyIdentities)input-TOP { A1-ADAPTER-API:input (object[(getPolicyIdentities)input], optional) } (getPolicyIdentities)input { A1-ADAPTER-API:near-rt-ric-url (Optional.empty, optional) }

[Try it out!](#) [Hide Response](#)

Request URL

http://localhost:8282/restconf/operations/A1-ADAPTER-API:getPolicyIdentities

Response Body

```
{
  "output": {
    "policy-id-list": [
      "my-policy-instance1",
      "bbab23de-8979-45fe-9458-f0167c721a11"
    ]
  }
}
```

Response Code

200

Response Headers

```
{
  "Access-Control-Allow-Credentials": "true",
  "Access-Control-Allow-Origin": "http://localhost:8282",
  "Content-Length": "92",
  "Content-Type": "application/json"
}
```