

docs : REST API documentation

GET

/docs/index.html

Get Swagger UI main index page

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

[Hide Response](#)

Curl

curl -X GET --header 'Accept: text/html' 'http://192.168.186.138:9922/onos/v1/docs/index.html'

Request URL

http://192.168.186.138:9922/onos/v1/docs/index.html

Response Body

```
<!-- Adapted from Swagger UI -->
<!DOCTYPE html>
<html>
<head>
  <title>ONOS API Docs</title>
  <link rel="icon" type="image/png" href="images/onos-logo.png"/>

  <link href='css/typography.css' media='screen' rel='stylesheet' type='text/css'/>
  <link href='css/reset.css' media='screen' rel='stylesheet' type='text/css'/>
  <link href='css/screen.css' media='screen' rel='stylesheet' type='text/css'/>
  <link href='css/reset.css' media='print' rel='stylesheet' type='text/css'/>
  <link href='css/print.css' media='print' rel='stylesheet' type='text/css'/>
  <script src='lib/object-assign-pollyfill.js' type='text/javascript'></script>
  <script src='lib/jquery-1.8.0.min.js' type='text/javascript'></script>
  <script src='lib/jquery.slideto.min.js' type='text/javascript'></script>
  <script src='lib/jquery.wiggle.min.js' type='text/javascript'></script>
  <script src='lib/jquery.ba-bbq.min.js' type='text/javascript'></script>
  <script src='lib/handlebars-4.0.5.js' type='text/javascript'></script>
  <script src='lib/lodash.min.js' type='text/javascript'></script>
  <script src='lib/backbone.min.js' type='text/javascript'></script>
```

Response Code

200

Response Headers

```
{
  "server": "StarOS.xyz",
  "connection": "Close",
  "content-length": "4633",
  "content-type": "text/html"
}
```

GET /docs/apis/{key: .*?}/swagger.json

[Get Swagger UI JSON](#)

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|----------------------|----------------|-----------|
| key | <input type="text" value="(required)"/> | REST API web context | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /docs/apis/{key: .*?}/model.json

[Get REST API model schema](#)

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|----------------------|----------------|-----------|
| name | <input type="text" value="(required)"/> | REST API web context | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET </docs/apis>

[Get all registered REST API docs](#)

Implementation Notes

Returns array of all registered API docs.

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /docs

[Get Swagger UI main index page](#)

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /docs/{resource: .*}

[Get Swagger UI resource](#)

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-----------------------|----------------------|----------------|-----------|
| resource | <div>(required)</div> | path of the resource | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

applications : Manage inventory of applications

DELETE /applications/{name}

Uninstall application

Implementation Notes

Uninstalls the specified application deactivating it first if necessary.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------|-------------|----------------|-----------|
|-----------|-------|-------------|----------------|-----------|

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|------------------|----------------|-----------|
| name | <input type="text" value="(required)"/> | application name | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET

/applications/{name}

Get application details

Implementation Notes

Returns details of the specified application.

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "name": "org.onosproject.distributedprimitives",
  "id": 1,
  "version": "1.2.3",
  "category": "default",
  "description": "ONOS app to test distributed primitives",
  "readme": "ONOS app to test distributed primitives.",
  "origin": "ON.Lab",
  "url": "http://onosproject.org",
  "permissions": [
    ""
  ],
  "featuresRepo": "mvn:org.onosproject/onos-app-cip/1.2.3/xml/features",
  "features": [
    "onos-app-distributed-primitives"
  ],
  "requiredApps": [
    ""
  ],
  "state": "ACTIVE"
}
```

Response Content Type

application/json

▼

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------|-------------|----------------|-----------|
|-----------|-------|-------------|----------------|-----------|

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|------------------|----------------|-----------|
| name | <input type="text" value="(required)"/> | application name | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /applications/ids/entry [Gets applicationId entry by either id or name](#)

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "name": "org.onosproject.distributedprimitives",
  "id": 1
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|---------------------|----------------|-----------|
| id | <input type="text" value="(required)"/> | id of application | query | string |
| name | <input type="text" value="(required)"/> | name of application | query | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /applications [Get all installed applications](#)

Implementation Notes

Returns array of all installed applications.

Response Class (Status 200)

successful operation

Model Example Value

```
{
  "applications": [
    {
      "name": "org.onosproject.distributedprimitives",
      "id": 1,
      "version": "1.2.3",
      "category": "default",
      "description": "ONOS app to test distributed primitives",
      "readme": "ONOS app to test distributed primitives.",
      "origin": "ON.Lab",
      "url": "http://onosproject.org",
      "permissions": [
        ""
      ],
      "featuresRepo": "mvn:org.onosproject/onos-app-cip/1.2.3/xml/features",
      "features": [
        "onos-app-distributed-primitives"
      ],
      "requiredApps": [
        ""
      ],
      "state": "ACTIVE"
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

POST **/applications**

[Install a new application](#)

Implementation Notes

Uploads application archive stream and optionally activates the application.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|-----------------------------------|----------------------------------|----------------|-----------|
| activate | <input type="text" value="true"/> | true to activate app also | query | boolean |
| stream | | | body | undefined |

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-----------------------------------|----------------|-----------|
| | <div>(required)</div> | application archive stream | | |
| | Parameter content type: application/octet_stream ▾ | | | |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

POST /applications/{name}/register Registers an on or off platform application

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-----------------------|-------------------------|----------------|-----------|
| name | <div>(required)</div> | application name | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /applications/ids Gets a collection of application ids

Implementation Notes

Returns array of all registered application ids.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "applicationIds": [
```

```
{
  "name": "org.onosproject.distributedprimitives",
  "id": 1
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

| | | | |
|---------|------------------|--|--|
| default | Unexpected error | | |
|---------|------------------|--|--|

Try it out!

DELETE /applications/{name}/active

De-activate application

Implementation Notes

De-activates the specified application.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|------------------|----------------|-----------|
| name | <input type="text" value="(required)"/> | application name | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

| | | | |
|-----|----------------------|--|--|
| 200 | successful operation | | |
|-----|----------------------|--|--|

| | | | |
|---------|------------------|--|--|
| default | Unexpected error | | |
|---------|------------------|--|--|

Try it out!

POST /applications/{name}/active

Activate application

Implementation Notes

Activates the specified application.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|------------------|----------------|-----------|
| name | <input type="text" value="(required)"/> | application name | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

cluster : Manage cluster of ONOS instances

POST /cluster/configuration Forms cluster of ONOS instances

Implementation Notes

Forms ONOS cluster using the uploaded JSON definition.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|--------------------|----------------|---|
| config | <div>(required)</div> <div>Parameter content type: application/json ▾</div> | cluster definition | body | <div>Model</div> <div>Example Value</div> <pre>{ "nodes": [{ "id": "127.0.0.1", "ip": "127.0.0.1", "tcpPort": 9876, "status": "ACTIVE" }] }</pre> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

GET /cluster Get all cluster nodes

Implementation Notes

Returns array of all cluster nodes.

Response Class (Status 200)

successful operation

[Model](#)
[Example Value](#)

```
{
  "nodes": [
    {
      "id": "127.0.0.1",
      "ip": "127.0.0.1",
      "tcpPort": 9876,
      "status": "ACTIVE"
    }
  ]
}
```

Response Content Type

[Response Messages](#)

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

GET [/cluster/{id}](#)

[Get cluster node details](#)

[Implementation Notes](#)

Returns details of the specified cluster node.

[Response Class \(Status 200\)](#)

successful operation

[Model](#)
[Example Value](#)

```
{
  "id": "127.0.0.1",
  "ip": "127.0.0.1",
  "tcpPort": 9876,
  "status": "ACTIVE"
}
```

Response Content Type

[Parameters](#)

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------------|----------------|-----------|
| id | <input type="text" value="(required)"/> | cluster node identifier | path | string |

[Response Messages](#)

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

configuration : Manage component configurations

DELETE

/configuration/{component}

Selectively clears configuration properties

Implementation Notes

Clears only the properties present in the JSON request.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|--------------------|----------------|-----------|
| component | (required) | component name | path | string |
| request | (required) | JSON configuration | body | undefined |

Parameter content type:
application/json

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET

/configuration/{component}

Gets configuration of the specified component

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|----------------|----------------|-----------|
| component | (required) | component name | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

POST /configuration/{component} Selectively sets configuration properties

Implementation Notes

Sets only the properties present in the JSON request.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-----------------------------|----------------|-----------|
| component | <div>(required)</div> | component name | path | string |
| preset | <div>true ▾</div> | preset the property if true | query | boolean |
| request | <div>(required)</div> <div></div> <div>Parameter content type:<div>application/json ▾</div></div> | JSON configuration | body | undefined |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET

/configuration

Gets all component configurations

Implementation Notes

Returns collection of all registered component configurations.

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

keys : Query and Manage Device Keys

GET

/keys

Gets all device keys

Implementation Notes

Returns array of all device keys.

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "keys": [
    {
      "id": "Device key unique identifier",
      "type": "COMMUNITY_NAME, OR USERNAME_PASSWORD",
      "label": "Device key user label",
      "community_name": "Device key community name",
      "username": "Device key username",
      "password": "Device key password"
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST

/keys

Adds a new device key from the JSON input stream

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|------------------------|----------------|--|
| stream | (required) | device key JSON stream | body | Model Example Value |
| | | | | <pre>{ "id": "Device key unique identifier", "type": "COMMUNITY_NAME, OR USERNAME_PASSWORD", "label": "Device key user label",</pre> |

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|--|
| | <div>Parameter content type:<div>application/json ▾</div></div> | | | <pre>"community_name": "Device key community name", "username": "Device key username", "password": "Device key password" }</pre> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

DELETE

/keys/{id}

Removes a device key by device key identifier

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|-----------------------|----------------|-----------|
| id | (required) | device key identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET

/keys/{id}

Gets a single device key by device key unique identifier

Implementation Notes

Returns the specified device key.

Response Class (Status 200)

successful operation

Model

Example Value

```
{  
  "id": "Device key unique identifier",  
  "type": "COMMUNITY_NAME, OR USERNAME_PASSWORD",  
  "label": "Device key user label",  
}
```

```
"community_name": "Device key community name",
"username": "Device key username",
"password": "Device key password"
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-----------------------|----------------|-----------|
| id | <input type="text" value="(required)"/> | device key identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

devices : Manage inventory of infrastructure devices

GET /devices

[Gets all infrastructure devices](#)

Implementation Notes

Returns array of all discovered infrastructure devices.

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "devices": [
    {
      "id": "of:0000000000000001",
      "type": "SWITCH",
      "available": true,
      "role": "of:0000000000000001",
      "mfr": "Nicira, Inc.",
      "hw": "Open vSwitch",
      "sw": "2.3.1",
      "serial": "123",
      "driver": "ovs",
      "chassisId": "1",
      "annotations": {
        "managementAddress": "123",
        "protocol": "OF_13",
        "channelId": "10.128.12.4:34689"
      }
    }
  ]
}
```

```
}
]
}
```

Response Content Type application/json ▾

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

DELETE **/devices/{id}**

Removes infrastructure device

Implementation Notes

Administratively deletes the specified device from the inventory of known devices.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------------------------|--------------------------|----------------|-----------|
| id | (required) | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET **/devices/{id}**

Gets details of infrastructure device

Implementation Notes

Returns details of the specified infrastructure device.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "id": "of:0000000000000001",
  "type": "SWITCH",
  "available": true,
```



```
"role": "of:0000000000000001",
"mfr": "Nicira, Inc.",
"hw": "Open vSwitch",
"sw": "2.3.1",
"serial": "123",
"driver": "ovs",
"chassisId": "1",
"annotations": {
  "managementAddress": "123",
  "protocol": "OF_13",
  "driver": "ovs",
  "channelId": "10.128.12.4:34689"
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| id | <input type="text" value="(required)"/> | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

GET /devices/{id}/ports

Gets ports of infrastructure device

Implementation Notes

Returns details of the specified infrastructure device.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "id": "of:0000000000000001",
  "type": "SWITCH",
  "available": true,
  "role": "of:0000000000000001",
  "mfr": "Nicira, Inc.",
  "hw": "Open vSwitch",
  "sw": "2.3.1",
  "serial": "123",
  "driver": "ovs",
```

```
"chassisId": "1",
"annotations": {
  "managementAddress": "123",
  "protocol": "OF_13",
  "channelId": "10.128.12.4:34689"
},
"ports": [
  {
    "element": "of:0000000000000001",
    "port": "2",
    "isEnabled": true,
    "type": "copper",
    "portSpeed": 0,
    "annotations": {
      "portName": "s1"
    }
  }
]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| id | <input type="text" value="(required)"/> | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

dpis : Query the latest DPI statistics info

GET /dpis

[Gets the latest dpi statistics](#)

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "receivedTime": "2016-06-12 04:05:05",
  "dpiStatInfo": {
    "trafficStatistics": {
      "ethernetBytes": 69889,
      "discardedBytes": 69889,

```

```
"ipPackets": 69889,
"totalPackets": 69889,
"ipBytes": 69889,
"avgPktSize": 9889,
"uniqueFlows": 9889,
"tcpPackets": 69889,
"udpPackets": 69889,
"dpiThroughputPps": 69889.12,
"dpiThroughputBps": 69889.12,
"trafficThroughputPps": 69889.12,
"trafficThroughputBps": 69889.12,
"trafficDurationSec": 69889.12,
"guessedFlowProtos": 9889
},
"detectedProtos": [
  {
    "name": "TCP",
    "breed": "Acceptable",
    "packets": 69889,
    "bytes": 69889,
    "flows": 9889
  }
],
"knownFlows": [
  {
    "protocol": "TCP",
    "hostAName": "10.0.20.50",
    "hostAPort": 9889,
    "hostBName": "10.0.20.10",
    "hostBPort": 8181,
    "detectedProtocol": 80,
    "detectedProtocolName": "HTTP",
    "packets": 69889,
    "bytes": 69889,
    "hostSeverName": "raptor"
  }
],
"unknownFlows": [
  {
    "protocol": "TCP",
    "hostAName": "10.0.20.50",
    "hostAPort": 9889,
    "hostBName": "10.0.20.10",
    "hostBPort": 8181,
    "detectedProtocol": 80,
    "detectedProtocolName": "HTTP",
    "packets": 69889,
    "bytes": 69889,
    "hostSeverName": "raptor"
  }
]
}
}
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-------------|-------------------|-----------------|----------------|-----------|
| topn | (required) | max size | query | integer |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST /dpis

Add new dpi statistics entry at the end of list

Parameters

| Parameter | Value | Description | Parameter Type | Data Type | |
|---------------|------------|----------------------------|----------------|-----------|---------------|
| stream | (required) | dpi statistics JSON | body | Model | Example Value |

Parameter content type:

application/json

```
{  
  "receivedTime": "2016-06-12 04:05:  
  "dpiStatInfo": {  
    "trafficStatistics": {  
      "ethernetBytes": 69889,  
      "discardedBytes": 69889,  
      "ipPackets": 69889,  
      "totalPackets": 69889,  
      "ipBytes": 69889,  
      "avgPktSize": 9889,  
      "uniqueFlows": 9889,
```

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |

default

Unexpected error

Try it out!

```
GET /dpis/traffic
```

Gets the latest traffic statistics only

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "receivedTime": "2016-06-12 04:05:05",
  "dpiStatInfo": {
    "trafficStatistics": {
      "ethernetBytes": 69889,
      "discardedBytes": 69889,
      "ipPackets": 69889,
      "totalPackets": 69889,
      "ipBytes": 69889,
      "avgPktSize": 9889,
      "uniqueFlows": 9889,
      "tcpPackets": 69889,
      "udpPackets": 69889,
      "dpiThroughputPps": 69889.12,
      "dpiThroughputBps": 69889.12,
      "trafficThroughputPps": 69889.12,
      "trafficThroughputBps": 69889.12,
      "trafficDurationSec": 69889.12,
      "guessedFlowProtos": 9889
    }
  }
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /dpis/unknownFlows

Gets the latest unknown flows statistics only

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "receivedTime": "2016-06-12 04:05:05",
  "dpiStatInfo": {
    "unknownFlows": [
      {
        "protocol": "TCP",
        "hostAName": "10.0.20.50",
        "hostAPort": 9889,
        "hostBName": "10.0.20.10",

```

```
    "hostBPort": 8181,
    "detectedProtocol": 80,
    "detectedProtocolName": "HTTP",
    "packets": 69889,
    "bytes": 69889,
    "hostSeverName": "raptor"
  }
]
}
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|-----------|
| topn | <input type="text" value="(required)"/> | max size | query | integer |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET /dpis/protocols

Gets the latest detected protocol statistics only

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "receivedTime": "2016-06-12 04:05:05",
  "dpiStatInfo": {
    "detectedProtos": [
      {
        "name": "TCP",
        "breed": "Acceptable",
        "packets": 69889,
        "bytes": 69889,
        "flows": 9889
      }
    ]
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|-----------|
| topn | <input type="text" value="(required)"/> | max size | query | integer |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /dpis/knownFlows

Gets the latest known flows statistics only

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "receivedTime": "2016-06-12 04:05:05",
  "dpiStatInfo": {
    "knownFlows": [
      {
        "protocol": "TCP",
        "hostAName": "10.0.20.50",
        "hostAPort": 9889,
        "hostBName": "10.0.20.10",
        "hostBPort": 8181,
        "detectedProtocol": 80,
        "detectedProtocolName": "HTTP",
        "packets": 69889,
        "bytes": 69889,
        "hostSeverName": "raptor"
      }
    ]
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|-----------|
| topn | <input type="text" value="(required)"/> | max size | query | integer |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

flowobjectives : Manage flow objectives

POST /flowobjectives/policy

Installs the filtering rules onto the specified device

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|---------------------|----------------|---|
| stream | <div>(required)</div> <div>Parameter content type: application/json</div> | filtering rule JSON | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "policy": "policy expression"}</pre></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

POST /flowobjectives/{deviceId}/forward

Creates and installs a new forwarding objective for the specified device

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|---------------------------|----------------|-----------------------|
| appId | (required) | application identifier | query | string |
| deviceId | (required) | device identifier | path | string |
| stream | | forwarding objective JSON | body | Model Example Value |

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|--|
| | <div>(required)</div> <div>Parameter content type: application/json</div> | | | <pre>{ "flag": "SPECIFIC", "priority": 40000, "timeout": 0, "isPermanent": true, "deviceId": "of:0000000000000001", "operation": "ADD", "selector": { "criteria": [{ "type": "ETH_TYPE", "value": "00:00:00:00:00:00" }] } }</pre> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /flowobjectives/next Returns the globally unique nextId

Response Class (Status 200)
successful operation

Model
Example Value

```
{
  "nextId": 3
}
```

Response Content Type application/json

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST /flowobjectives/{deviceId}/filter Creates and installs a new filtering objective for the specified device

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------|-------------|----------------|-----------|
|-----------|-------|-------------|----------------|-----------|

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|--------------------------|----------------|--|
| appId | <div>(required)</div> | application identifier | query | string |
| deviceId | <div>(required)</div> | device identifier | path | string |
| stream | <div><div>(required)</div><div>Parameter content type:<div>application/json</div></div></div> | filtering objective JSON | body | <div>Model<div>Example Value</div><div><pre>{ "type": "PERMIT", "priority": 400000, "timeout": 0, "isPermanent": true, "deviceId": "of:0000000000000000", "operation": "ADD", "key": { "type": "ETH_TYPE", "ethType": "0x88cc", "mac": "00:00:11:00:00:01", } }</pre></div></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

POST /flowobjectives/{deviceId}/next

Creates and installs a new next objective for the specified device

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|--|------------------------|---|-----------|
| appId | <div>(required)</div> | application identifier | query | string |
| deviceId | <div>(required)</div> | device identifier | path | string |
| stream | <div><div>(required)</div><div>Parameter content type:<div>application/json</div></div></div> <div>next objective JSON</div> | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "type": "HASHED", "id": 1, "priority": 400000, "timeout": 0, "isPermanent": true, "deviceId": "of:0000000000000001", "operation": "ADD", "meta": { "criteria": [</pre></div> | |

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------|-------------|----------------|-----------|
|-----------|-------|-------------|----------------|-----------|

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

flows : Query and program flow rules

DELETE

/flows/application/{appId}

Removes flow rules by application ID

Implementation Notes

Removes a collection of flow rules generated by the given application.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|------------------------|----------------|-----------|
| appId | (required) | application identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

| | | |
|---|----------------------------|---|
| GET | /flows/application/{appld} | Gets flow rules generated by an application |
| <h3>Implementation Notes</h3> <p>Returns the flow rule specified by the application id.</p> | | |
| <h3>Response Class (Status 200)</h3> <p>successful operation</p> | | |
| <h3>Model</h3> <h4><u>Example Value</u></h4> <pre>{ "flows": [{ "id": "12103425214920339",</pre> | | |

```
"tableId": 3,
"appId": "org.onosproject.core",
"groupId": 0,
"priority": 40000,
"timeout": 0,
"isPermanent": true,
"deviceId": "of:0000000000000003",
"liveType": "UNKNOWN",
"treatment": {
  "instructions": [
    {
      "type": "OUTPUT",
      "port": "CONTROLLER"
    }
  ],
  "deferred": [
    "string"
  ]
},
"selector": {
  "criteria": [
    {
      "type": "ETH_TYPE",
      "ethType": "0x88cc",
      "mac": "00:00:11:00:00:01",
      "port": 1,
      "metadata": "0xabcdL",
      "vlanId": "0x1000",
      "priority": 1,
      "protocol": 1,
      "ip": "10.1.1.0/24",
      "tcpPort": 1,
      "udpPort": 1,
      "sctpPort": 1,
      "icmpType": 1,
      "icmpCode": 1,
      "flowLabel": "0xfffffe",
      "icmpv6Type": 1,
      "icmpv6Code": 1,
      "targetAddress": "10.1.1.0/24",
      "label": 1,
      "exthdrFlags": 1,
      "lambda": 1,
      "gridType": "DWDm",
      "channelSpacing": 100,
      "spacingMultiplier": 4,
      "slotGranularity": 8,
      "ochSignalId": 1,
      "tunnelId": 5,
      "ochSignalType": 1,
      "oduSignalId": 1,
      "tributaryPortNumber": 11,
      "tributarySlotLen": 80,
```

```
        "tributarySlotBitmap": [
          1
        ],
        "oduSignalType": 4
      }
    ]
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|------------------------|----------------|-----------|
| appId | <input type="text" value="(required)"/> | application identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|--|------------------|----------------|---------|
| default | Unexpected error | | |
| <input type="button" value="Try it out!"/> | | | |

DELETE /flows

Removes a batch of flow rules

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|------------------------|----------------|-----------|
| stream | <div><input type="text" value="(required)"/></div> <div>Parameter content type: <input type="text" value="application/json"/></div> | stream for posted JSON | body | undefined |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|--|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| <input type="button" value="Try it out!"/> | | | |

Implementation Notes

Returns array of all flow rules in the system.

Response Class (Status 200)

successful operation

Model Example Value

```
{
  "flows": [
    {
      "id": "12103425214920339",
      "tableId": 3,
      "appId": "org.onosproject.core",
      "groupId": 0,
      "priority": 40000,
      "timeout": 0,
      "isPermanent": true,
      "deviceId": "of:0000000000000003",
      "state": "ADDED",
      "life": 69889,
      "packets": 22546,
      "bytes": 1826226,
      "liveType": "UNKNOWN",
      "lastSeen": 1447892365670,
      "treatment": {
        "instructions": [
          {
            "type": "OUTPUT",
            "port": "CONTROLLER"
          }
        ],
        "deferred": [
          "string"
        ]
      },
      "selector": {
        "criteria": [
          {
            "type": "ETH_TYPE",
            "ethType": "0x88cc",
            "mac": "00:00:11:00:00:01",
            "port": 1,
            "metadata": "0xabcdL",
            "vlanId": "0x1000",
            "priority": 1,
            "protocol": 1,
            "ip": "10.1.1.0/24",
            "tcpPort": 1,
            "udpPort": 1,

```

```
    "sctpPort": 1,
    "icmpType": 1,
    "icmpCode": 1,
    "flowLabel": "0xfffffe",
    "icmpv6Type": 1,
    "icmpv6Code": 1,
    "targetAddress": "10.1.1.0/24",
    "label": 1,
    "extHdrFlags": 1,
    "lambda": 1,
    "gridType": "DWD",
    "channelSpacing": 100,
    "spacingMultiplier": 4,
    "slotGranularity": 8,
    "ochSignalId": 1,
    "tunnelId": 5,
    "ochSignalType": 1,
    "oduSignalId": 1,
    "tributaryPortNumber": 11,
    "tributarySlotLen": 80,
    "tributarySlotBitmap": [
      1
    ],
    "oduSignalType": 4
  }
]
}
}
]
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

POST /flows

Creates new flow rules

Implementation Notes

Creates and installs a new flow rules.

Flow rule criteria and instruction description: <https://wiki.onosproject.org/display/ONOS/Flow+Rules>

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|----------------|----------------|-----------|
| appId | <input type="text" value="(required)"/> | application id | query | string |

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-----------------|----------------|---|
| stream | <div>(required)</div> <div>Parameter content type: application/json</div> | flow rules JSON | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "flows": [{ "deviceid": "of: 00000000000000", "cookie": "0", "cookiemask": "0", "tableid": "0", "command": "0", "idletimeout": "0", "hardtimeout": "0", "priority": "40000", "..." }] }</pre></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

DELETE /flows/{deviceid}/{flowid} Removes flow rule

Implementation Notes

Removes the specified flow rule.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|----------------------|----------------|-----------|
| deviceId | (required) | device identifier | path | string |
| flowId | (required) | flow rule identifier | path | double |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /flows/{deviceid}/{flowid} Gets flow rules

Implementation Notes

Returns the flow entry specified by the device id and flow rule id.

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "flows": [
    {
      "id": "12103425214920339",
      "tableId": 3,
      "appId": "org.onosproject.core",
      "groupId": 0,
      "priority": 40000,
      "timeout": 0,
      "isPermanent": true,
      "deviceId": "of:0000000000000003",
      "state": "ADDED",
      "life": 69889,
      "packets": 22546,
      "bytes": 1826226,
      "liveType": "UNKNOWN",
      "lastSeen": 1447892365670,
      "treatment": {
        "instructions": [
          {
            "type": "OUTPUT",
            "port": "CONTROLLER"
          }
        ],
        "deferred": [
          "string"
        ]
      },
      "selector": {
        "criteria": [
          {
            "type": "ETH_TYPE",
            "ethType": "0x88cc",
            "mac": "00:00:11:00:00:01",
            "port": 1,
            "metadata": "0xabcdL",
            "vlanId": "0x1000",
            "priority": 1,
            "protocol": 1,
            "ip": "10.1.1.0/24",
            "tcpPort": 1,
            "udpPort": 1,
            "sctpPort": 1,

```

```
    "icmpType": 1,
    "icmpCode": 1,
    "flowLabel": "0xfffffe",
    "icmpv6Type": 1,
    "icmpv6Code": 1,
    "targetAddress": "10.1.1.0/24",
    "label": 1,
    "extHdrFlags": 1,
    "lambda": 1,
    "gridType": "DWDM",
    "channelSpacing": 100,
    "spacingMultiplier": 4,
    "slotGranularity": 8,
    "ochSignalId": 1,
    "tunnelId": 5,
    "ochSignalType": 1,
    "oduSignalId": 1,
    "tributaryPortNumber": 11,
    "tributarySlotLen": 80,
    "tributarySlotBitmap": [
      1
    ],
    "oduSignalType": 4
  }
]
}
}
]
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|----------------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |
| flowId | <input type="text" value="(required)"/> | flow rule identifier | path | double |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET /flows/{deviceId}

Gets flow entries of a device

Implementation Notes

Returns array of all flow rules for the specified device.

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "flows": [
    {
      "id": "12103425214920339",
      "tableId": 3,
      "appId": "org.onosproject.core",
      "groupId": 0,
      "priority": 40000,
      "timeout": 0,
      "isPermanent": true,
      "deviceId": "of:0000000000000003",
      "state": "ADDED",
      "life": 69889,
      "packets": 22546,
      "bytes": 1826226,
      "liveType": "UNKNOWN",
      "lastSeen": 1447892365670,
      "treatment": {
        "instructions": [
          {
            "type": "OUTPUT",
            "port": "CONTROLLER"
          }
        ],
        "deferred": [
          "string"
        ]
      },
      "selector": {
        "criteria": [
          {
            "type": "ETH_TYPE",
            "ethType": "0x88cc",
            "mac": "00:00:11:00:00:01",
            "port": 1,
            "metadata": "0xabcdL",
            "vlanId": "0x1000",
            "priority": 1,
            "protocol": 1,
            "ip": "10.1.1.0/24",
            "tcpPort": 1,
            "udpPort": 1,
            "sctpPort": 1,

```

```
    "icmpType": 1,
    "icmpCode": 1,
    "flowLabel": "0xfffffe",
    "icmpv6Type": 1,
    "icmpv6Code": 1,
    "targetAddress": "10.1.1.0/24",
    "label": 1,
    "extHdrFlags": 1,
    "lambda": 1,
    "gridType": "DWD",
    "channelSpacing": 100,
    "spacingMultiplier": 4,
    "slotGranularity": 8,
    "ochSignalId": 1,
    "tunnelId": 5,
    "ochSignalType": 1,
    "oduSignalId": 1,
    "tributaryPortNumber": 11,
    "tributarySlotLen": 80,
    "tributarySlotBitmap": [
      1
    ],
    "oduSignalType": 4
  }
]
}
}
]
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

POST /flows/{deviceId}

Creates new flow rule

Implementation Notes

Creates and installs a new flow rule for the specified device.

Flow rule criteria and instruction description: <https://wiki.onosproject.org/display/ONOS/Flow+Rules>

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|------------------------|----------------|---|
| deviceId | (required) | device identifier | path | string |
| appId | (required) | application identifier | query | string |
| stream | (required) | flow rule JSON | body | <div>Model</div> <div>Example Value</div> <pre>{ "flows": [{ "deviceid": "of: 00000000000000", "cookie": "0", "cookiemask": "0", "tableid": "0", "command": "0", "idletimeout": "0", "hardtimeout": "0", "priority": "40000", </pre> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| Try it out! | | | |

groups : Query and program group rules

| | | |
|---|---------|-----------------------------------|
| GET | /groups | Returns all groups of all devices |
| <div><h3>Response Class (Status 200)</h3><p>successful operation</p><p><u>Model</u></p><p><u>Example Value</u></p><pre>{ "groups": [{ "id": "1", "state": "PENDING_ADD", "life": 69889, "packets": 22546, "bytes": 1826226, "referenceCount": 1826226, "type": "ALL", </pre></div> | | |

```
{
  "deviceId": "of:0000000000000003",
  "appId": "org.onosproject.rest",
  "appCookie": "0x1234abcd",
  "buckets": [
    {
      "treatment": {
        "instructions": [
          {
            "type": "OUTPUT",
            "port": "2"
          }
        ]
      },
      "weight": "1.0",
      "watchPort": "2",
      "watchGroup": "1"
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /groups/{deviceId}

Returns all groups associated with the given device

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "groups": [
    {
      "id": "1",
      "state": "PENDING_ADD",
      "life": 69889,
      "packets": 22546,
      "bytes": 1826226,
      "referenceCount": 1826226,
      "type": "ALL",
      "deviceId": "of:0000000000000003",
      "appId": "org.onosproject.rest",

```

```
{
  "appCookie": "0x1234abcd",
  "buckets": [
    {
      "treatment": {
        "instructions": [
          {
            "type": "OUTPUT",
            "port": "2"
          }
        ]
      },
      "weight": "1.0",
      "watchPort": "2",
      "watchGroup": "1"
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

POST /groups/{deviceId} [Create new group rule](#)

Implementation Notes

Creates and installs a new group rule for the specified device.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |

| | | | | |
|--|-----------------|------|-------|---------------|
| stream | group rule JSON | body | Model | Example Value |
| <pre>{ "type": "ALL", "appCookie": "0x1234abcd", }</pre> | | | | |

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|--|
| | <div>(required)</div> <div>Parameter content type: application/json ▾</div> | | | <pre>"groupId": "1", "buckets": [{ "treatment": {</pre> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

DELETE

/groups/{deviceId}/{appCookie}

Removes the specified group

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|--|----------------|-----------|
| deviceId | (required) | device identifier | path | string |
| appCookie | (required) | application cookie to be used for lookup | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

| | | |
|-----------------------------|--------------------------------|---|
| GET | /groups/{deviceId}/{appCookie} | Returns a group with the given deviceId and appCookie |
| Response Class (Status 200) | | |
| successful operation | | |
| Model | | |
| Example Value | | |
| <pre>{ "id": "1",</pre> | | |


```
"state": "PENDING_ADD",
"life": 69889,
"packets": 22546,
"bytes": 1826226,
"referenceCount": 1826226,
"type": "ALL",
"deviceId": "of:000000000000003",
"appId": "org.onosproject.rest",
"appCookie": "0x1234abcd",
"buckets": [
  {
    "treatment": {
      "instructions": [
        {
          "type": "OUTPUT",
          "port": "2"
        }
      ]
    },
    "weight": "1.0",
    "watchPort": "2",
    "watchGroup": "1"
  }
]
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |
| appCookie | <input type="text" value="(required)"/> | group key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

hosts : Manage inventory of end-station hosts

DELETE /hosts/{mac}/{vlan}

Removes infrastructure device

Implementation Notes

Administratively deletes the specified device from the inventory of known devices.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "id": "46:E4:3C:A4:17:C8/-1",
  "mac": "46:E4:3C:A4:17:C8",
  "vlan": "-1",
  "ipAddresses": [
    "127.0.0.1"
  ],
  "location": {
    "elementId": "of:0000000000000002",
    "port": "3"
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|----------------------|----------------|-----------|
| mac | <input type="text" value="(required)"/> | host MAC address | path | string |
| vlan | <input type="text" value="(required)"/> | host VLAN identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET /hosts/{mac}/{vlan} Get details of end-station host with MAC/VLAN

Implementation Notes

Returns detailed properties of the specified end-station host.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "id": "46:E4:3C:A4:17:C8/-1",
  "mac": "46:E4:3C:A4:17:C8",
```

```
"vlan": "-1",
"ipAddresses": [
  "127.0.0.1"
],
"location": {
  "elementId": "of:0000000000000002",
  "port": "3"
}
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|----------------------|----------------|-----------|
| mac | <input type="text" value="(required)"/> | host MAC address | path | string |
| vlan | <input type="text" value="(required)"/> | host VLAN identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET /hosts/{id}

[Get details of end-station host](#)

Implementation Notes

Returns detailed properties of the specified end-station host.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "id": "46:E4:3C:A4:17:C8/-1",
  "mac": "46:E4:3C:A4:17:C8",
  "vlan": "-1",
  "ipAddresses": [
    "127.0.0.1"
  ],
  "location": {
    "elementId": "of:0000000000000002",
    "port": "3"
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-----------------|----------------|-----------|
| id | <input type="text" value="(required)"/> | host identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

GET **/hosts**

[Get all end-station hosts](#)

Implementation Notes

Returns array of all known end-station hosts.

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "hosts": [
    {
      "id": "46:E4:3C:A4:17:C8/-1",
      "mac": "46:E4:3C:A4:17:C8",
      "vlan": "-1",
      "ipAddresses": [
        "127.0.0.1"
      ],
      "location": {
        "elementId": "of:0000000000000002",
        "port": "3"
      }
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST **/hosts**

Creates a new host based on JSON input and adds it to the current host inventory

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|---------------|---|-------------------|----------------|--|
| stream | (required) <div></div> <div>Parameter content type: application/json ▾</div> | input JSON | body | Model <div><pre>{ "mac": "46:E4:3C:A4:17:C8", "vlan": "-1", "ipAddresses": ["127.0.0.1"], "location": { "elementId": "of:0000000000000002", "port": "3" } }</pre></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

intents : Query, submit and withdraw network intents

GET **/intents/relatedflows/{appId}/{key}**

Gets all related flow entries created by a particular intent

Implementation Notes

Returns all flow entries of the specified intent.

Response Class (Status 200)

successful operation

Model Example Value

```
{
  "id": "0x5",
  "appId": "org.onosproject.ovsdb",
  "type": "HostToHostIntent",
  "paths": [
    [
```

```
{
  "id": "12103425214920339",
  "tableId": 3,
  "appId": "org.onosproject.core",
  "groupId": 0,
  "priority": 40000,
  "timeout": 0,
  "isPermanent": true,
  "deviceId": "of:0000000000000003",
  "state": "ADDED",
  "life": 69889,
  "packets": 22546,
  "bytes": 1826226,
  "lastSeen": 1447892365670,
  "treatment": {
    "instructions": [
      {
        "type": "OUTPUT",
        "port": "CONTROLLER"
      }
    ],
    "deferred": [
      "string"
    ]
  },
  "selector": {
    "criteria": [
      {
        "type": "ETH_TYPE",
        "ethType": "0x88cc",
        "mac": "00:00:11:00:00:01",
        "port": 1
      }
    ]
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|------------------------|----------------|-----------|
| appId | <input type="text" value="(required)"/> | application identifier | path | string |
| key | <input type="text" value="(required)"/> | intent key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

GET /intents Gets all intents

Implementation Notes

Returns array containing all the intents in the system.

Response Class (Status 200)

successful operation

Model

Example Value

```
{  "intents": [    {      "type": "HostToHostIntent",      "id": "0x6",      "appId": "org.onosproject.ovsdb",      "priority": 55,      "resources": [        "46:E4:3C:A4:17:C8/-1"      ]    }  ]}
```

Response Content Type

application/json ▾

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

POST /intents Submits a new intent

Implementation Notes

Creates and submits intent from the JSON request.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------|-------------|----------------|-----------|
|-----------|-------|-------------|----------------|-----------|

| Parameter | Value | Description | Parameter Type | Data Type | |
|---------------|---|-------------------|----------------|---|---------------|
| stream | <div>(required)</div> <div></div> <div>Parameter content type: application/json ▾</div> | input JSON | body | Model | Example Value |
| | | | | <pre>{ "type": "HostToHostIntent", "appId": "org.onosproject.ovsdb", "priority": 55, "one": "46:E4:3C:A4:17:C8/-1", "two": "08:00:27:56:8a:15/-1"}</pre> | |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

DELETE /intents/{appId}/{key} Withdraws intent

Implementation Notes

Withdraws the specified intent from the system.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|------------------------|----------------|-----------|
| appId | (required) | application identifier | path | string |
| key | (required) | intent key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

| | | | | | |
|-----|------------------------|------------------------------------|--|--|--|
| GET | /intents/{appId}/{key} | Gets intent by application and key | | | |
|-----|------------------------|------------------------------------|--|--|--|

Implementation Notes

Returns details of the specified intent.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "intents": [
    {
      "type": "HostToHostIntent",
      "id": "0x6",
      "appId": "org.onosproject.ovsdb",
      "priority": 55,
      "resources": [
        "46:E4:3C:A4:17:C8/-1"
      ]
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|------------------------|----------------|-----------|
| appId | <input type="text" value="(required)"/> | application identifier | path | string |
| key | <input type="text" value="(required)"/> | intent key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

links : Manage inventory of infrastructure links

GET

/links

Gets infrastructure links

Implementation Notes

Returns array of all links, or links for the specified device or port.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "links": [
    {
      "src": {
        "port": "3",
        "device": "of:0000000000000002"
      },
      "dst": {
        "port": "2",
        "device": "of:0000000000000003"
      },
      "type": "DIRECT",
      "state": "ACTIVE"
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|----------------------|--------------------------------|----------------|-----------|
| device | <input type="text"/> | (optional) device identifier | query | string |
| port | <input type="text"/> | (optional) port number | query | string |
| direction | <input type="text"/> | (optional) direction qualifier | query | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

mastership : Manage the mastership of ONOS instances

GET

/mastership/{deviceId}/role

Returns controllers connected to a given device, in order of preference

Implementation Notes

The first entry in the list is the current master.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "master": "1",
  "backups": [
    "1"
  ]
}
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------------------------|-------------------|----------------|-----------|
| deviceId | (required) | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /mastership Balances the mastership to be shared as evenly as possibly by all online instances

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

PUT /mastership Applies the current mastership role for the specified device

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------|--|----------------|---|
| stream | | JSON representation of device, node, mastership info | body | <div>Model Example Value<div>{ "deviceId": "of:0000000000000001",</div></div> |

| Parameter | Value | Description | Parameter Type | Data Type |
|--|-----------------------|-------------|----------------|--|
| | <div>(required)</div> | | | <pre>"nodeId": "1", "role": "MASTER" }</pre> |
| Parameter content type: <div>application/json</div> | | | | |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /mastership/{deviceId}/request

Returns the mastership status of the local controller for a given device forcing master selection if necessary

Response Class (Status 200)

successful operation

Model

Example Value

```
{  
  "role": "MASTER"  
}
```

Response Content Type

application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-----------------------|-------------------|----------------|-----------|
| deviceId | <div>(required)</div> | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /mastership/{deviceId}/master

Returns the current master for a given device

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "nodeId": "1"
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET /mastership/{nodeId}/device

Returns the devices for which a controller is master

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "deviceIds": [
    "of:0000000000000001"
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-----------------------|----------------|-----------|
| nodeId | <input type="text" value="(required)"/> | controller identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

GET

/mastership/{deviceId}/relinquish

Abandons mastership of the specified device on the local node thus forcing selection of a new master

Implementation Notes

If the local node is not a master for this device, no master selection will occur.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|-------------------|----------------|-----------|
| deviceId | (required) | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET

/mastership/{deviceId}/local

Returns the role of the local node for the specified device

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "role": "MASTER"
}
```

Response Content Type

application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|-------------------|----------------|-----------|
| deviceId | (required) | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

meters : Query and program meter rules

GET /meters Returns all meters of all devices

Response Class (Status 200)

successful operation

Model

Example Value

```
{  "meters": [    {      "id": "1",      "appId": "org.onosproject.rest",      "deviceId": "of:0000000000000001",      "unit": "KB_PER_SEC",      "burst": true,      "state": "ADDED",      "life": 0,      "refCount": 0,      "packets": 0,      "bytes": 0,      "bands": [        {          "type": "REMARK",          "rate": 0,          "burstSize": 0,          "prec": 0,          "packets": 0,          "bytes": 0        }      ]    }  ]}
```

Response Content Type application/json

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

DELETE /meters/{deviceId}/{meterId}

Removes the specified meter

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |
| meterId | <input type="text" value="(required)"/> | meter identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /meters/{deviceId}/{meterId}

Returns a meter by the meter id

Response Class (Status 200)

successful operation

Model Example Value

```
{
  "id": "1",
  "appId": "org.onosproject.rest",
  "deviceId": "of:0000000000000001",
  "unit": "KB_PER_SEC",
  "burst": true,
  "state": "ADDED",
  "life": 0,
  "refCount": 0,
  "packets": 0,
  "bytes": 0,
  "bands": [
    {
      "type": "REMARK",
      "rate": 0,
      "burstSize": 0,
      "prec": 0,
      "packets": 0,
      "bytes": 0
    }
  ]
}
```


Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |
| meterId | <input type="text" value="(required)"/> | meter identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

GET /meters/{deviceId}

Returns a collection of meters by the device id

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "meters": [
    {
      "id": "1",
      "appId": "org.onosproject.rest",
      "deviceId": "of:0000000000000001",
      "unit": "KB_PER_SEC",
      "burst": true,
      "state": "ADDED",
      "life": 0,
      "refCount": 0,
      "packets": 0,
      "bytes": 0,
      "bands": [
        {
          "type": "REMARK",
          "rate": 0,
          "burstSize": 0,
          "prec": 0,
          "packets": 0,
          "bytes": 0
        }
      ]
    }
  ]
}
```

Response Content Type application/json ▾

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------------------------|-------------------|----------------|-----------|
| deviceId | (required) | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

POST **/meters/{deviceId}**

Creates new meter rule

Implementation Notes

Creates and installs a new meter rule for the specified device.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------------------------|-------------------|----------------|-----------|
| deviceId | (required) | device identifier | path | string |
| stream | (required) | meter rule JSON | body | Model |

Parameter content type:
application/json ▾

```
{
  "deviceId": "of:0000000000000001",
  "unit": "KB_PER_SEC",
  "burst": true,
  "bands": [
    {
      "type": "REMARK",
      "rate": "0",
      "burstSize": "0",
      "prec": "0"
    }
  ]
}
```

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

[Try it out!](#)

metrics : Query metrics

GET

/metrics/{metricName}

Gets stats information of a metric

Implementation Notes

Returns array of all information for the specified metric.

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "metric": {
    "timer": {
      "counter": "1",
      "mean_rate": "1.0",
      "1_min_rate": "1.0",
      "5_min_rate": "1.0",
      "15_min_rate": "1.0",
      "mean": "1.0",
      "min": "1.0",
      "max": "1.0",
      "stddev": "1.0"
    }
  }
}
```

Response Content Type

application/json ▾

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|------------|-------------|----------------|-----------|
| metricName | (required) | metric name | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET

/metrics

Gets stats information of all metrics

Implementation Notes

Returns array of all information for all metrics.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "metrics": [
    {
      "metric": {
        "timer": {
          "counter": "1",
          "mean_rate": "1.0",
          "1_min_rate": "1.0",
          "5_min_rate": "1.0",
          "15_min_rate": "1.0",
          "mean": "1.0",
          "min": "1.0",
          "max": "1.0",
          "stddev": "1.0"
        }
      }
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

mcast : Manage the multicast routing information

DELETE /mcast

Remove a multicast route

Implementation Notes

Removes a route from the multicast RIB.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "source": "10.1.1.0",
```

```
"group": "10.1.1.0"
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|--|----------------------|----------------|-----------|
| stream | (required) <div></div> <div>Parameter content type: <input type="text" value="application/json"/></div> | multicast route JSON | body | undefined |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /mcast

[Get all multicast routes](#)

Implementation Notes

Returns array of all known multicast routes.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "routes": [
    {
      "source": "10.1.1.0",
      "group": "10.1.1.0",
      "type": "STATIC"
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

POST /mcast Create new multicast route

Implementation Notes

Creates a new route in the multicast RIB.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|----------------------|----------------|---|
| stream | <div>(required)</div> <div>Parameter content type: application/json ▾</div> | multicast route JSON | body | <div>Model Example Value</div> <div><pre>{ "source": "10.1.1.0", "group": "10.1.1.0"}</pre></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

POST /mcast/sinks/{group}/{source} Create a sink for a multicast route

Implementation Notes

Creates a new sink for an existing multicast route.

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|-------------------|----------------|-----------|
| group | (required) | group IP address | path | string |
| source | (required) | source IP address | path | string |

| Parameter | Value | Description | Parameter Type | Data Type | | | | |
|-----------|---|-------------|----------------|--|-------|---------------|--|---|
| stream | <div>(required)</div> <div>Parameter content type: application/json ▾</div> | sink JSON | body | <table><thead><tr><th>Model</th><th>Example Value</th></tr></thead><tbody><tr><td></td><td><pre>{ "sinks": ["of:0000000000000001/1"] }</pre></td></tr></tbody></table> | Model | Example Value | | <pre>{ "sinks": ["of:0000000000000001/1"] }</pre> |
| Model | Example Value | | | | | | | |
| | <pre>{ "sinks": ["of:0000000000000001/1"] }</pre> | | | | | | | |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

network/configuration : Manage network configurations

DELETE /network/configuration Clear entire network configuration base

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /network/configuration Gets entire network configuration base

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

POST **/network/configuration**

Uploads bulk network configuration

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|----------------|---|--|----------------|-----------|
| request | (required) <div></div> <div>Parameter content type: application/json ▾</div> | network configuration JSON rooted at the top node | body | undefined |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

DELETE **/network/configuration/{subjectClassKey}**

Clear all network configurations for a subject class

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------------------|---------------------------|--------------------------|----------------|-----------|
| subjectClassKey | (required) <div></div> | subject class key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET **/network/configuration/{subjectClassKey}**

Gets all network configuration for a subject class

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------------------|---------------------------|--------------------------|----------------|-----------|
| subjectClassKey | (required) <div></div> | subject class key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

POST /network/configuration/{subjectClassKey}

Upload multiple network configurations for a subject class

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|---|---|----------------|-----------|
| subjectClassKey | <div>(required)</div> | subject class key | path | string |
| request | <div><div>(required)</div><div>Parameter content type:<div>application/json</div></div></div> | network configuration JSON rooted at the top node | body | undefined |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

DELETE /network/configuration/{subjectClassKey}/{subjectKey}/{configKey}

Clear specific network configuration for a subjectKey

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|------------|-------------------------|----------------|-----------|
| subjectClassKey | (required) | subjectKey class key | path | string |
| subjectKey | (required) | subjectKey key | path | string |
| configKey | (required) | configuration class key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

GET

/network/configuration/{subjectClassKey}/{subjectKey}/{configKey}

Gets specific network configuration for a subjectKey

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|-----------------------|-------------------------|----------------|-----------|
| subjectClassKey | <div>(required)</div> | subjectKey class key | path | string |
| subjectKey | <div>(required)</div> | subjectKey key | path | string |
| configKey | <div>(required)</div> | configuration class key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

POST

/network/configuration/{subjectClassKey}/{subjectKey}/{configKey}

Upload specific network configuration for a subjectKey

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|------------|---|----------------|-----------|
| subjectClassKey | (required) | subjectKey class key | path | string |
| subjectKey | (required) | subjectKey key | path | string |
| configKey | (required) | configuration class key | path | string |
| request | (required) | network configuration JSON rooted at the top node | body | undefined |

| Parameter | Value | Description | Parameter Type | Data Type |
|-------------------|---|----------------|----------------|-----------|
| | Parameter content type: application/json ▾ | | | |
| Response Messages | | | | |
| HTTP Status Code | Reason | Response Model | | Headers |
| 200 | successful operation | | | |
| default | Unexpected error | | | |
| Try it out! | | | | |

DELETE /network/configuration/{subjectClassKey}/{subjectKey} Clear all network configurations for a subjectKey

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|------------|----------------------|----------------|-----------|
| subjectClassKey | (required) | subjectKey class key | path | string |
| subjectKey | (required) | subjectKey key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /network/configuration/{subjectClassKey}/{subjectKey} [Gets all network configuration for a subjectKey](#)

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|-----------------------|----------------------|----------------|-----------|
| subjectClassKey | <div>(required)</div> | subjectKey class key | path | string |
| subjectKey | <div>(required)</div> | subjectKey key | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| <div></div> | | | |

Try it out!

POST /network/configuration/{subjectClassKey}/{subjectKey}

Upload mutiple network configurations for a subjectKey

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|--|---|----------------|-----------|
| subjectClassKey | <input type="text" value="(required)"/> | subjectKey class key | path | string |
| subjectKey | <input type="text" value="(required)"/> | subjectKey key | path | string |
| request | <div><input type="text" value="(required)"/> <div>Parameter content type: <input type="text" value="application/json"/></div></div> | network configuration JSON rooted at the top node | body | undefined |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

paths : Compute paths in the network graph

GET /paths/{src}/{dst}/disjoint

Gets all shortest disjoint path pairs between any two hosts or devices

Implementation Notes

Returns array of all shortest disjoint path pairs between any two elements.

Response Class (Status 200)

successful operation

Model Example Value

```
{
  "paths": [
    {
      "cost": 2,
      "links": [
```

```
{
  "src": {
    "port": "3",
    "device": "of:0000000000000002"
  },
  "dst": {
    "port": "2",
    "device": "of:0000000000000003"
  },
  "type": "DIRECT",
  "state": "ACTIVE"
}
]
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|---|-------------------------------|----------------|-----------|
| src | <input type="text" value="(required)"/> | source identifier | path | string |
| dst | <input type="text" value="(required)"/> | destination identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET **/paths/{src}/{dst}**

[Gets all shortest paths between any two hosts or devices](#)

Implementation Notes

Returns array of all shortest paths between any two elements.

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "paths": [
    {
      "cost": 2,
      "links": [
        {
```

```
{
  "src": {
    "port": "3",
    "device": "of:0000000000000002"
  },
  "dst": {
    "port": "2",
    "device": "of:0000000000000003"
  },
  "type": "DIRECT",
  "state": "ACTIVE"
}
]
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|---|-------------------------------|----------------|-----------|
| src | <input type="text" value="(required)"/> | source identifier | path | string |
| dst | <input type="text" value="(required)"/> | destination identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

regions : Manages region and device membership

GET /regions

Returns set of all regions

Response Class (Status 200)

successful operation

[Model](#)
[Example Value](#)

```
{
  "regions": [
    {
      "id": "1",
      "name": "region",
      "type": "ROOM",
      "masters": [

```

```
    "1"
  ]
}
}
```

Response Content Type application/json ▾

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

| | |
|---------|------------------|
| default | Unexpected error |
|---------|------------------|

Try it out!

POST **/regions**

Creates a new region using the supplied JSON input stream

Parameters

| Parameter | Value | Description | Parameter Type | Data Type | | | | |
|-----------|--|--------------------|----------------|---|-------|---------------|--|--|
| stream | <div>(required)</div> <div>Parameter content type: application/json</div> | region JSON stream | body | <table><tr><th>Model</th><th>Example Value</th></tr><tr><td></td><td><pre>{ "id": "1", "name": "region", "type": "ROOM", "masters": ["1"] }</pre></td></tr></table> | Model | Example Value | | <pre>{ "id": "1", "name": "region", "type": "ROOM", "masters": ["1"] }</pre> |
| Model | Example Value | | | | | | | |
| | <pre>{ "id": "1", "name": "region", "type": "ROOM", "masters": ["1"] }</pre> | | | | | | | |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

| | |
|-----|----------------------|
| 200 | successful operation |
|-----|----------------------|

| | |
|---------|------------------|
| default | Unexpected error |
|---------|------------------|

Try it out!

DELETE **/regions/{regionId}**

Removes the specified region using the given region identifier

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------|-------------|----------------|-----------|
|-----------|-------|-------------|----------------|-----------|

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| regionId | <input type="text" value="(required)"/> | region identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

GET

/regions/{regionId}

Returns the region with the specified identifier

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "id": "1",
  "name": "region",
  "type": "ROOM",
  "masters": [
    [
      "1"
    ]
  ]
}
```

Response Content Type

application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|-------------------|----------------|-----------|
| regionId | (required) | region identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

| | | |
|-----|---------------------|---|
| PUT | /regions/{regionId} | Updates the specified region using the supplied JSON input stream |
| | | |

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|--|--------------------|----------------|--|
| regionId | (required) | region identifier | path | string |
| stream | (required) <div>Parameter content type: application/json ▾</div> | region JSON stream | body | <div>Model</div> <div>Example Value</div> <pre>{ "id": "1", "name": "region", "type": "ROOM", "masters": [["1"]] }</pre> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

DELETE /regions/{regionId}/devices

Removes the specified collection of devices from the region

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "deviceIds": [
    "of:0000000000000001"
  ]
}
```

Response Content Type application/json ▾

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|-------------------|----------------|-----------|
| regionId | (required) | region identifier | path | string |
| stream | | | body | undefined |

| Parameter | Value | Description | Parameter Type | Data Type |
|---|-----------------------|------------------------------|----------------|-----------|
| | <div>(required)</div> | deviceIds JSON stream | | |
| Parameter content type: application/json ▾ | | | | |
| Response Messages | | | | |
| HTTP Status Code | Reason | Response Model | Headers | |
| default | Unexpected error | | | |
| <div>Try it out!</div> | | | | |

GET /regions/{regionId}/devices Returns the set of devices that belong to the specified region

Response Class (Status 200)
successful operation

Model
Example Value

```
{
  "deviceIds": [
    "of:0000000000000001"
  ]
}
```

Response Content Type application/json ▾

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------------|-----------------------|--------------------------|----------------|-----------|
| regionId | <div>(required)</div> | region identifier | path | string |

Response Messages

| | | | |
|------------------|------------------|----------------|---------|
| HTTP Status Code | Reason | Response Model | Headers |
| default | Unexpected error | | |

Try it out!

POST /regions/{regionId}/devices Adds the specified collection of devices to the region

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|--|------------------------|----------------|--|
| regionId | (required) | region identifier | path | string |
| stream | (required) <div>Parameter content type: application/json ▾</div> | deviceId's JSON stream | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "deviceIds": ["of:0000000000000001"] }</pre></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

statistics : Query flow statistics

GET /statistics/ports/{deviceId} Gets port statistics of a specified devices

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "statistics": [
    {
      "deviceId": "of:0000000000000001",
      "ports": [
        {
          "port": 1,
          "packetsReceived": 98,
          "packetsSent": 98,
          "bytesReceived": 9162,
          "bytesSent": 9162,
          "packetsRxDropped": 0,
          "packetsTxDropped": 0,

```

```
        "packetsRxErrors": 0,
        "packetsTxErrors": 0,
        "durationSec": 90
    }
  ]
}
]
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device ID | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET /statistics/delta/ports/{deviceId}/{port}

Gets port delta statistics of a specified device and port

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "statistics": [
    {
      "deviceId": "of:0000000000000001",
      "ports": [
        {
          "port": 1,
          "packetsReceived": 98,
          "packetsSent": 98,
          "bytesReceived": 9162,
          "bytesSent": 9162,
          "packetsRxDropped": 0,
          "packetsTxDropped": 0,
          "packetsRxErrors": 0,
          "packetsTxErrors": 0,
          "durationSec": 90
        }
      ]
    }
  ]
}
```

```
]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device ID | path | string |
| port | <input type="text" value="(required)"/> | port | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET /statistics/ports

Gets port statistics of all devices

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "statistics": [
    {
      "deviceId": "of:0000000000000001",
      "ports": [
        {
          "port": 1,
          "packetsReceived": 98,
          "packetsSent": 98,
          "bytesReceived": 9162,
          "bytesSent": 9162,
          "packetsRxDropped": 0,
          "packetsTxDropped": 0,
          "packetsRxEerrors": 0,
          "packetsTxErrors": 0,
          "durationSec": 90
        }
      ]
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

GET /statistics/delta/ports/{deviceId}

Gets port delta statistics of a specified devices

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "statistics": [
    {
      "deviceId": "of:0000000000000001",
      "ports": [
        {
          "port": 1,
          "packetsReceived": 98,
          "packetsSent": 98,
          "bytesReceived": 9162,
          "bytesSent": 9162,
          "packetsRxDropped": 0,
          "packetsTxDropped": 0,
          "packetsRxEErrors": 0,
          "packetsTxErrors": 0,
          "durationSec": 90
        }
      ]
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device ID | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET

/statistics/ports/{deviceId}/{port}

Gets port statistics of a specified device and port

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "statistics": [
    {
      "deviceId": "of:0000000000000001",
      "ports": [
        {
          "port": 1,
          "packetsReceived": 98,
          "packetsSent": 98,
          "bytesReceived": 9162,
          "bytesSent": 9162,
          "packetsRxDropped": 0,
          "packetsTxDropped": 0,
          "packetsRxEerrors": 0,
          "packetsTxErrors": 0,
          "durationSec": 90
        }
      ]
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------|----------------|-----------|
| deviceId | <input type="text" value="(required)"/> | device ID | path | string |
| port | <input type="text" value="(required)"/> | port | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /statistics/flows/tables/{deviceId}

Gets table statistics for all tables of a specified device

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "statistics": [
    {
      "deviceId": "of:0000000000000001",
      "table": [
        {
          "tableId": 0,
          "deviceId": "of:0000000000000001",
          "activeEntries": 3,
          "packetsLookedUp": 458530,
          "packetsMathced": 458501
        }
      ]
    }
  ]
}
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|-------------|----------------|-----------|
| deviceId | (required) | device ID | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

default Unexpected error

Try it out!

GET /statistics/delta/ports

Gets port delta statistics of all devices

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "statistics": [
    {
```



```
{
  "deviceId": "of:0000000000000001",
  "ports": [
    {
      "port": 1,
      "packetsReceived": 98,
      "packetsSent": 98,
      "bytesReceived": 9162,
      "bytesSent": 9162,
      "packetsRxDropped": 0,
      "packetsTxDropped": 0,
      "packetsRxEerrors": 0,
      "packetsTxErrors": 0,
      "durationSec": 90
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /statistics/flows/link

Gets load statistics for all links or for a specific link

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "loads": [
    {
      "rate": 0,
      "latest": "of:0000000000000001",
      "valid": false,
      "time": 1447980119542,
      "link": "http://10.128.12.1:8181/onos/v1/links?device=of:0000000000000002&port=3"
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|----------------------|---|----------------|-----------|
| device | <input type="text"/> | (optional) device ID for a specific link | query | string |
| port | <input type="text"/> | (optional) port number for a specified link | query | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /statistics/flows/tables

Gets table statistics for all tables of all devices

Response Class (Status 200)

successful operation

Model Example Value

```
{
  "statistics": [
    {
      "deviceId": "of:0000000000000001",
      "table": [
        {
          "tableId": 0,
          "deviceId": "of:0000000000000001",
          "activeEntries": 3,
          "packetsLookedUp": 458530,
          "packetsMathced": 458501
        }
      ]
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

tenants : Query and manage tenants of virtual networks

GET

/tenants

Returns all tenant identifiers

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "tenants": [
    {
      "id": "Tenant123"
    }
  ]
}
```

Response Content Type

application/json

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST

/tenants

Creates a tenant with the given tenant identifier

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|----------------------|----------------|---|
| stream | <div>(required)</div> <div>Parameter content type: application/json</div> | TenantId JSON stream | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "id": "Tenant123" }</pre></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

DELETE

/tenants/{tenantId}

Removes the specified tenant with the specified tenant identifier

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|-------------------|----------------|-----------|
| tenantId | (required) | tenant identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

topology : Query network topology graph and its components

GET

/topology/clusters

Gets overview of topology SCCs

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "clusters": [
    {
      "id": 1,
      "deviceCount": 3,
      "linkCount": 4,
      "root": "of:0000000000000001"
    }
  ]
}
```

Response Content Type

application/json

Response Messages

HTTP Status Code

Reason

Response Model

Headers

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

GET /topology/clusters/{id}/devices

Gets devices in a specific SCC

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "devices": [
    "of:0000000000000001"
  ]
}
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|----------------------------|----------------|-----------|
| id | (required) | id of the cluster to query | path | integer |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /topology

Gets overview of current topology

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "path": {
    "dst": {
      "dpid": "2222",
      "port": "3"
    },

```

```
{
  "src": {
    "dpid": "1111",
    "port": "2"
  }
}
```

Response Content Type application/json ▼

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /topology/clusters/{id}/links

Gets links in specific SCC

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "links": [
    {
      "src": {
        "port": "3",
        "device": "of:0000000000000002"
      },
      "dst": {
        "port": "2",
        "device": "of:0000000000000003"
      },
      "type": "DIRECT",
      "state": "ACTIVE"
    }
  ]
}
```

Response Content Type application/json ▼

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|-------------------------|----------------------------|----------------|-----------|
| id | (required) | id of the cluster to query | path | integer |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

GET /topology/infrastructure/{connectPoint}

Tests if a connect point is infrastructure or edge

Response Class (Status 200)

successful operation

Model

Example Value

```
{  "infrastructure": true}
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|--------------|------------|---------------------|----------------|-----------|
| connectPoint | (required) | deviceid:portnumber | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /topology/broadcast/{connectPoint}

Tests if a connect point is in broadcast set

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "broadcast": true
}
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|--------------|---|---------------------|----------------|-----------|
| connectPoint | <input type="text" value="(required)"/> | deviceid:portnumber | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /topology/clusters/{id}

Gets details of a specific SCC

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "id": 1,
  "deviceCount": 3,
  "linkCount": 4,
  "root": "of:0000000000000001"
}
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|----------------------------|----------------|-----------|
| id | (required) | id of the cluster to query | path | integer |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

vnets : Query and Manage Virtual Network elements

| | | |
|-------------------|--------------------|---|
| DELETE | /vnets/{networkId} | Removes the virtual network with the specified network identifier |
| Parameters | | |

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|--------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

DELETE

/vnets/{networkId}/devices/{deviceId}/ports/{portNum}

Removes the virtual network port from the virtual device in a virtual network

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|---------------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |
| deviceId | <input type="text" value="(required)"/> | virtual device identifier | path | string |
| portNum | <input type="text" value="(required)"/> | virtual port number | path | double |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

| | | |
|---|-----------------------------------|--|
| GET | /vnets/{networkId}/devices | Returns all virtual network devices in a virtual network |
| Response Class (Status 200) | | |
| successful operation | | |
| <u>Model</u> <u>Example Value</u> | | |
| <pre>{ "devices": [{ "networkId": 3, "deviceId": "of:0000000000000042" }] }</pre> | | |

```
]
}
```

Response Content Type application/json ▾

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------------|-------------------------|---------------------------|----------------|-----------|
| networkId | (required) | network identifier | path | double |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

POST **/vnets/{networkId}/devices**

Creates a virtual device from the JSON input stream

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------------|-------------------------|---------------------------|----------------|-----------|
| networkId | (required) | network identifier | path | double |

| | | | | | |
|---------------|-------------------------------------|-----------------------------------|------|---|--|
| stream | (required) <div></div> | virtual device JSON stream | body | Model Example Value | |
| | | | | <div><pre>{ "networkId": 3, "deviceId": "of:0000000000000042" }</pre></div> | |

Parameter content type:
application/json ▾

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |

default Unexpected error

[Try it out!](#)

DELETE **/vnets/{networkId}/devices/{deviceId}**

Removes the virtual network device from the virtual network

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|--------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |
| deviceId | <input type="text" value="(required)"/> | device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

GET /vnets/{networkId}/devices/{deviceId}/ports

Returns all virtual network ports in a virtual device in a virtual network

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "ports": [
    {
      "networkId": 3,
      "deviceId": "of:0000000000000042",
      "portNum": 34,
      "physDeviceId": "of:0000000000000003",
      "physPortNum": 2
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|---------------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |
| deviceId | <input type="text" value="(required)"/> | virtual device identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

POST

/vnets/{networkId}/devices/{deviceId}/ports

Creates a virtual network port in a virtual device in a virtual network

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|---------------------------|----------------|---|
| networkId | (required) | network identifier | path | double |
| deviceId | (required) | virtual device identifier | path | string |
| stream | (required) | virtual port JSON stream | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "networkId": 3, "deviceId": "of:0000000000000042", "portNum": 34, "physDeviceId": "of:0000000000000003", "physPortNum": 2}</pre></div> |

Parameter content type:

application/json

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

| | | |
|-----------------------------|-------------------|---|
| GET | /vnets/{tenantId} | Returns the virtual networks with the specified tenant identifier |
| Response Class (Status 200) | | |
| successful operation | | |
| Model | | |
| <u>Example Value</u> | | |

```
{
  "vnets": [
    {
      "networkId": 3,
      "tenantId": "Tenant123"
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|-------------------|----------------|-----------|
| tenantId | <input type="text" value="(required)"/> | tenant identifier | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

GET /vnets

Returns all virtual networks

Response Class (Status 200)

successful operation

[Model](#)
[Example Value](#)

```
{
  "vnets": [
    {
      "networkId": 3,
      "tenantId": "Tenant123"
    }
  ]
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

POST /vnets

Creates a virtual network from the JSON input stream

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|------------|----------------------------------|----------------|------------------------------------|
| stream | (required) | tenant identifier JSON stream | body | Model |
| | | | | Example Value |
| | | | | <pre>{ "id": "Tenant123" }</pre> |

Parameter content type:

application/json ▾

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

DELETE /vnets/{networkId}/links

Removes the virtual network link from the JSON input stream

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "networkId": 3,
  "src": {
    "port": 3,
    "device": "of:0000000000000002"
  },
  "dst": {
    "port": 2,
    "device": "of:0000000000000003"
  },
  "type": "DIRECT",
  "state": "ACTIVE"
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------------|---|---------------------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |
| stream | <div><div><div>(required)</div></div></div> | virtual link JSON stream | body | undefined |

Parameter content type:

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /vnets/{networkId}/links

Returns all virtual network links in a virtual network

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "links": [
    {
      "networkId": 3,
      "src": {
        "port": 3,
        "device": "of:0000000000000002"
      },
      "dst": {
        "port": 2,
        "device": "of:0000000000000003"
      },
      "type": "VIRTUAL",
      "state": "ACTIVE"
    }
  ]
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|--------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST /vnets/{networkId}/links

Creates a virtual network link from the JSON input stream

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|--------------------------|----------------|--|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |
| stream | <div><div><input type="text" value="(required)"/></div><div>Parameter content type: <input type="text" value="application/json"/></div></div> | virtual link JSON stream | body | <div>Model Example Value</div> <div><pre>{ "networkId": 3, "src": { "port": 3, "device": "of:0000000000000000", }, "dst": { "port": 2, "device": "of:0000000000000000", }, "type": "DIRECT", }</pre></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

DELETE /vnets/{networkId}/hosts

Removes the virtual network host from the JSON input stream

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "networkId": 3,
  "id": "46:E4:3C:A4:17:C8/-1",
  "mac": "46:E4:3C:A4:17:C8",
  "vlan": "-1",
  "ipAddresses": [
    "127.0.0.1"
  ],
  "location": {
    "elementId": "of:000000000000002",
    "port": "3"
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|--|--------------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |
| stream | <div><div><div>(required)</div></div></div> <div>Parameter content type: <input type="text" value="application/json"/></div> | virtual host JSON stream | body | undefined |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /vnets/{networkId}/hosts

Returns all virtual network hosts in a virtual network

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "hosts": [
    {
      "networkId": 3,
```

```
{
  "id": "46:E4:3C:A4:17:C8/-1",
  "mac": "46:E4:3C:A4:17:C8",
  "vlan": "-1",
  "ipAddresses": [
    "127.0.0.1"
  ],
  "location": {
    "elementId": "of:0000000000000002",
    "port": "3"
  }
}
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|--------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST /vnets/{networkId}/hosts Creates a virtual network host from the JSON input stream

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|---|--------------------|----------------|-----------|
| networkId | <input type="text" value="(required)"/> | network identifier | path | double |

| | | | | |
|--------|---|--------------------------|------|---|
| stream | <div><div><input type="text" value="(required)"/></div><div>Parameter content type: application/json</div></div> | virtual host JSON stream | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "networkId": 3, "mac": "46:E4:3C:A4:17:C8", "vlan": "-1", "ipAddresses": ["127.0.0.1"], "location": { "elementId": "of:0000000000000000", "port": "3" } }</pre></div> |
|--------|---|--------------------------|------|---|

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|----------------------|----------------|---------|
| 200 | successful operation | | |
| default | Unexpected error | | |

Try it out!

topologyservice : Query and Manage Topology Service

DELETE /topologyservice/topology/{fromdevice}/{todevice}

Removes one Topology

Response Class (Status 200)

successful operation

Model Example Value

```
{
  "result": {
    "code": "0",
    "description": "ok"
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|---|--|----------------|-----------|
| fromdevice | <input type="text" value="(required)"/> | Delete one Topology by fromdevice&todevice | path | string |
| todevice | <input type="text" value="(required)"/> | Delete one Topology by fromdevice&todevice | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET /topologyservice/topology/{fromdevice}/{todevice}

Returns one Topology

Response Class (Status 200)

successful operation

[Model](#)

[Example Value](#)

```
{
  "path": {
    "dst": {
      "dpid": "2222",
      "port": "3"
    },
    "src": {
      "dpid": "1111",
      "port": "2"
    }
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|---|---|----------------|-----------|
| fromdevice | <input type="text" value="(required)"/> | Get one topology by fromdevice & todevice | path | string |
| todevice | <input type="text" value="(required)"/> | Get one topology by fromdevice & todevice | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

POST /topologyservice/topology/{fromdevice}/{todevice}

[Install a new Topology](#)

Response Class (Status 200)

successful operation

[Model](#)

[Example Value](#)

```
{
  "result": {
    "code": "0",
    "description": "ok"
  }
}
```

```
}  
}
```

Response Content Type application/json ▾

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|---|-----------------------------------|----------------|-----------|
| fromdevice | <div>(required)</div> | Create one Topology by fromdevice | path | string |
| todevice | <div>(required)</div> | Create one Topology by todevice | path | string |
| stream | <div><div>(required)</div><div>Parameter content type:<div>application/octet_stream</div></div></div> <div><div>Model</div><div>Example Value</div><div><pre>{ "path": { "dst": { "dpid": "2222", "port": "3" }, "src": { "dpid": "1111", "port": "2" } } }</pre></div></div> | Create one topology | body | |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

GET /topologyservice/topologys

Returns all Topologys

Response Class (Status 200)

successful operation

Model
Example Value

```
{  
  "paths": [  
    {  
      "dst": {  
        "dpid": "2222",  
        "port": "3"  
      }  
    }  
  ]  
}
```

```
    },
    "src": {
      "dpid": "1111",
      "port": "2"
    }
  },
  ],
  "resultCode": "1",
  "resultDesc": "ok"
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

[Try it out!](#)

tunnelservice : Query and Manage Tunnel Service

DELETE /tunnelservice/tunnel/{resourceid}

Removes one Tunnel

Response Class (Status 200)

successful operation

Model

Example Value

```
{
  "result": {
    "resourceId": "5",
    "code": "0",
    "description": "ok"
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|---|---------------------------------|----------------|-----------|
| resourceid | <input type="text" value="(required)"/> | Delete one tunnel by resourceid | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|--------|----------------|---------|
|------------------|--------|----------------|---------|

| HTTP Status Code | Reason | Response Model | Headers |
|------------------------|------------------|----------------|---------|
| default | Unexpected error | | |
| <div>Try it out!</div> | | | |

GET

/tunnelservice/tunnel/{resourceid}

Returns one Tunnel

Response Class (Status 200)

successful operation

Model

Example Value

```
{  "username": "test",  "type": "vlan",  "inputPort": "of:1234/33",  "outputPort": "of:1234/33",  "inputtunnelport": "of:1234/33",  "outputtunnelport": "of:1234/33",  "inputvxlanport": "of:1234/33",  "outputvxlanport": "of:1234/33",  "inputvlanid": "12",  "outputvlanid": "12",  "bandwidth": "44333"}
```

Response Content Type

application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|------------|------------------------------|----------------|-----------|
| resourceid | (required) | Get one tunnel by resourceid | path | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

| | | |
|--|------------------------------------|---------------------------------|
| PUT | /tunnelservice/tunnel/{resourceid} | Update a Tunnel |
| <div>Response Class (Status 200)</div> <div>successful operation</div> | | |

Model
Example Value

```
{
  "result": {
    "resourceId": "5",
    "code": "0",
    "description": "ok"
  }
}
```

Response Content Type

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|------------|---|---------------------------------|----------------|-----------|
| resourceid | <input type="text" value="(required)"/> | Updata one tunnel by resourceid | path | string |

| | | | | |
|--------|--|-------------------|------|---|
| stream | <div><div><div>(required)</div></div></div> <div>Parameter content type: <input type="text" value="application/json"/></div> | Update one tunnel | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "username": "test", "type": "vlan", "inputPort": "of:1234/33", "outputPort": "of:1234/33", "inputtunnelport": "of:1234/33", "outputtunnelport": "of:1234/33", "inputvxlanport": "of:1234/33", "outputvxlanport": "of:1234/33", "inputvlanid": "12", "outputvlanid": "12", "inputvlanname": "VLAN12", "outputvlanname": "VLAN12" }</pre></div> |
|--------|--|-------------------|------|---|

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

DELETE /tunnelservice/tunnels/active

Deactive all Tunnels

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "result": {
    "resourceId": "5",
```



```
"code": "0",
"description": "ok"
}
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST /tunnelservice/tunnels/active

Active all Tunnels

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "result": {
    "resourceId": "5",
    "code": "0",
    "description": "ok"
  }
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

POST /tunnelservice/tunnel

Install a new Tunnel

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "result": {
```

```
{
  "resourceId": "5",
  "code": "0",
  "description": "ok"
}
```

Response Content Type application/json ▾

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|-----------|--|-------------------|----------------|---|
| stream | (required) <div><div>Parameter content type: application/octet_stream ▾</div></div> | Create one tunnel | body | <div>Model</div> <div>Example Value</div> <div><pre>{ "username": "test", "type": "vlan", "inputPort": "of:1234/33", "outputPort": "of:1234/33", "inputtunnelport": "of:1234/33", "outputtunnelport": "of:1234/33", "inputvxlanport": "of:1234/33", "outputvxlanport": "of:1234/33", "inputvlanid": "12", "outputvlanid": "12", }</pre></div> |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

GET **/tunnelservice/tunnels**

Returns all Tunnels

Response Class (Status 200)

successful operation

Model
Example Value

```
{
  "tunnels": [
    {
      "id": "1",
      "username": "admin",
      "inputPort": "23",
      "inputvlanid": "2",
      "outputPort": "3",
      "outputvlanid": "3",
      "inputtunnelport": "of:1234/33",
      "outputtunnelport": "of:1234/33",
    }
  ]
}
```

```
{
  "inputvxlanport": "of:1234/33",
  "outputvxlanport": "of:1234/33",
  "bandwidth": "222",
  "type": "vlan",
  "state": "ACTIVE"
},
{
  "resultCode": "1",
  "resultDesc": "ok"
}
```

Response Content Type

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|------------------|------------------|----------------|---------|
| default | Unexpected error | | |

Try it out!

[BASE URL: /onos/v1 , API VERSION: 1.0]
[Swagger UI](#) (v2.2.10) adaptation