



OPEN NETWORKING  
FOUNDATION

# Real Time Media NBI REST Specification

Version 1.0  
March 2015

TR-517



ONF Document Type: Technical Recommendation

ONF Document Name: Real Time Media NBI REST Specification

## DISCLAIMER

THIS SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

Any marks and brands contained herein are the property of their respective owners.

Open Networking Foundation  
2275 E. Bayshore Road, Suite 103, Palo Alto, CA 94303  
[www.opennetworking.org](http://www.opennetworking.org)

©2015 Open Networking Foundation. All rights reserved.

Open Networking Foundation, the ONF symbol, and OpenFlow are registered trademarks of the Open Networking Foundation, in the United States and/or in other countries. All other brands, products, or service names are or may be trademarks or service marks of, and are used to identify, products or services of their respective owners.

## TABLE OF CONTENTS

Table of Contents .....	3
Introduction .....	5
SAMPLE USE CASES .....	6
Sample Use Case .....	6
SESSION ELEMENT MODEL .....	7
SessionElement .....	7
UML Representation .....	7
JSON REpresentation .....	7
Field Descriptions .....	8
PolicyElement .....	9
UML Representation .....	9
JSON REpresentation .....	9
Field Descriptions .....	10
API OPERATION TEMPLATES .....	12
GET /api/network/SessionElement .....	13
HEAD /api/network/SessionElement .....	16
POST /api/network/SessionElement.....	17
PUT /api/network/SessionElement .....	19
PATCH /api/network/SessionElement .....	20
DELETE /api/network/SessionElement .....	21
GET /api/network/SessionElement/{ID} .....	23
POST /api/network/SessionElement/{ID}.....	25
PUT /api/network/SessionElement/{ID} .....	26
PATCH /api/network/SessionElement/{ID} .....	27
DELETE /api/network/SessionElement/{ID} .....	30
GET /api/network/PolicyElement .....	31
HEAD /api/network/PolicyElement.....	33
POST /api/network/PolicyElement.....	34

PUT /api/network/PolicyElement ..... 36

PATCH /api/network/PolicyElement ..... 37

DELETE /api/network/PolicyElement..... 38

GET /api/network/PolicyElement/{ID} ..... 39

HEAD /api/network/PolicyElement/{ID}..... 41

POST /api/network/PolicyElement/{ID}..... 42

PUT /api/network/PolicyElement/{ID} ..... 43

PATCH /api/network/PolicyElement/{ID} ..... 44

DELETE /api/network/PolicyElement/{ID}..... 46

NOTIFICATION..... 47

    Resource creation notification ..... 47

    resource Modification Notification ..... 48

    resource Deletion Notification ..... 48

Swagger Specification ..... 50

    PolicyElement ..... 50

    SessionElement ..... 59

    open items ..... 70

    Authors and Contributors ..... 70

    Release History..... 71

## INTRODUCTION

The following document is intended to provide details of a REST API specification for Real Time Media as a North Bound Interface (NBI). The intent of this API is to provide a consistent/standardized interface between a Real Time Media application like UC&C and a SDN Network Service Module that satisfies the requirements of the IMTC Automating QoE version 2 use-case document.

This specification was created as a liaison between the IMTC and ONF and readers/implementers should first read the [IMTC Automating QoE use-case](#) to get familiar with the constructs and concepts before utilizing this specification. The sample use-case section below illustrates only a general high-level approach of how this API will operate and not the necessary details that implementers require.

This specification is applicable for any kind of deployment but was mainly intended for an enterprise solution within a single autonomy. Multi-autonomy implementations are possible (ex: MPLS WAN provider to enterprise LAN) but is considered out of scope for this specification

## SAMPLE USE CASES

### Sample Use Case

UC&C discovers the Network Service Module via DNS and establishes a persistent HTTPS connection. Upon a secured connection is established (mutual authentication), a UC&C will use HTTP GET method to read meta-data and learn what resources and operations are supported by the Network Service Module.

UC&C may read policies that are stored in the Network Service Module, using HTTP GET, and may add or update policies, using HTTP PUT/POST as specified in the operations section of this document. If a policy is accepted then it will be stored and enforced by the Network Service Module . Otherwise an attempt to add/update policy via HTTP POST/PUT will be rejected.

If a new session is created by UC&C then it may invoke a custom CAC operation, as specified in the operations section of this document. UC&C will send a request to add/update a session filled with session attributes. While the Network Service Module may reject the request or grant committed bandwidth for each media flow in the session. In addition upon success the Network Service Module will construct a SessionElement with URL that enables UC&C to monitor it, as specified in the operations section.

If a session is updated in a way that affects QoS then UC&C may invoke a custom CAC operation with the requested change. The semantics and handling is similar to the operation to establish a session. If the session update does not affect CAC then UC&C may update the session directly via HTTP PUT, as specified in the operations section.

If a session is terminated then UC&C will update the session SessionElement state accordingly. UC&C SHOULD NOT DELETE the session because it may be consumed for post processing purposes, such as reports and diagnostics. If UC&C will try to delete the session then the Network Service Module will accept or reject the request based on its policy.

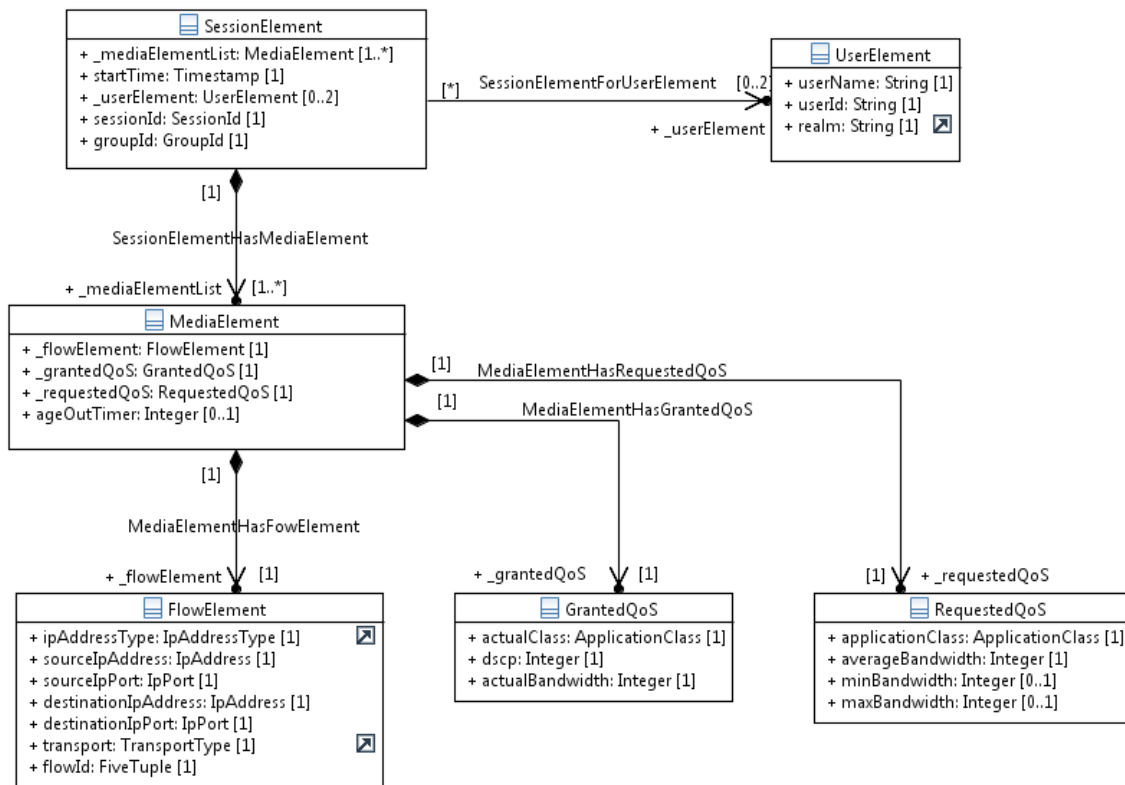
If UC&C is interested to monitor asynchronous events that affects QOS then it should subscribe to Notifications as specified in the Notification section of this document.

## SESSION ELEMENT MODEL

### SessionElement

A REST SessionElement being managed by this API is the SessionElement.

### UML REPRESENTATION



### JSON REPRESENTATION

```

{
  "mediaElement": [
    {
      "flowElement": {
        "ipAddressType": "",
        "sourceIpAddress": "",
        "sourceIpPort": "",
        "destinationIpAddress": "",
        "destinationIpPort": "",
        "transportType": "",
        "flowId": ""
      },
      "grantedQoS": {
        "actualClass": "",
        "dscp": 0,
        "actualBandwidth": 0
      },
      "requestedQoS": {
        "applicationClass": "",
        "averageBandwidth": 0,
        "minBandwidth": 0,
        "maxBandwidth": 0
      }
    }
  ]
}
    
```

```

    "requestedQoS": {
      "applicationClass": "",
      "averageBandwidth": 0,
      "maxBandwidth": 0,
      "minBandwidth": 0
    },
    "ageOutTimer": 0
  },
  "startTime": "",
  "userElement": [
    {
      "name": "",
      "id": "",
      "realm": ""
    }
  ],
  "groupId": "",
  "name": "",
  "id": "",
  "type": "",
  "alias": [
    {
      "name": "",
      "value": ""
    }
  ],
  "href": "",
  "association": [
    {
      "name": "",
      "role": "",
      "href": "",
      "validFor": ""
    }
  ],
  "resourceCharacteristic": [
    {
      "name": "",
      "value": ""
    }
  ],
  "resourceSpecification": [
    {
      "href": "",
      "role": ""
    }
  ]
}

```

---

## FIELD DESCRIPTIONS

Field	Description
<b>mediaElement</b>	The array of MediaElement that are supporting the session..
<b>startTime</b>	The QoS that was requested.
<b>userElement</b>	The UserElement for the user who created the session.
<b>groupId</b>	The group ID.

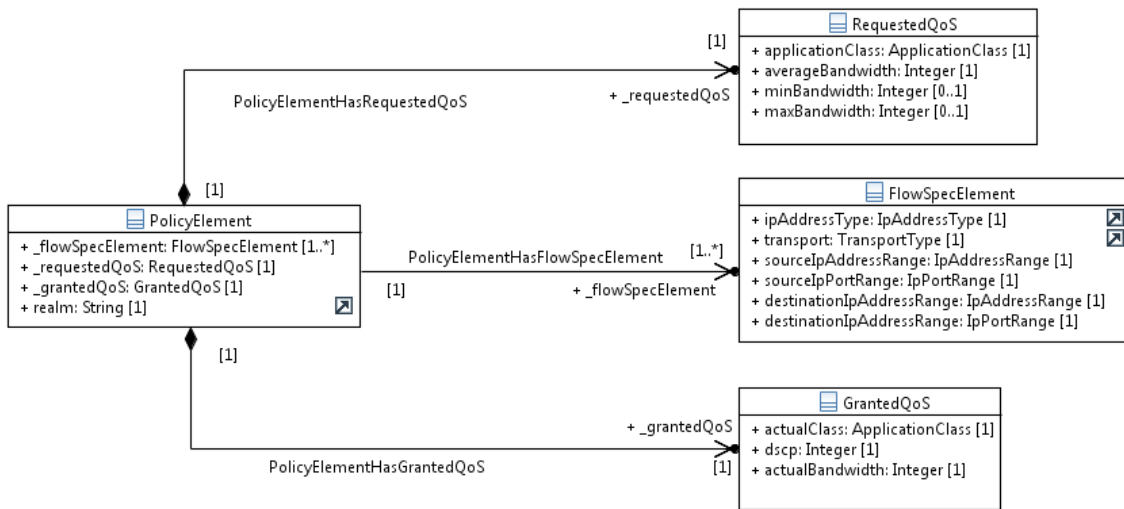


<b>id</b>	The unique identifier that is used for identification of the ManagedElement resources in the API.
<b>type</b>	Const string "managedElement".
<b>name</b>	The common name of the managed element.
<b>description</b>	Not supported in this release. Set to null.
<b>lifeCycleState</b>	The life cycle state of the ME. Defaults to NULL.
<b>alias</b>	Contains the alternative identifiers for the ME such as GUI name, TID, MTOSI name.
<b>assoication</b>	List of assoications to other REST managed resources. In this release only contains one relationship. That of the managed element to its AlarmCounts SessionElement.
<b>isBundle</b>	Always false in this release,
<b>bundle</b>	List of bundled REST managed resources. Defaults to NULL
<b>resourceSpecification</b>	An href to a SessionElement specification. SessionElement specifications are not support in this release so this will always will be NULL.
<b>characteristic</b>	A list of values that are valid for the instance of the ME. These values can come from definitions in characteristic specifications (not supported in this release) or ad-hoch. It will also be used to hold informaion the underlying system has about the managed elements that is useful to clients and that does not have an attribute in the managed element resorce.

## PolicyElement

A REST SessionElement being managed by this API is the PolicyElement.

## UML REPRESENTATION



## JSON REPRESENTATION

```

{
  "flowSpecElement": [
    {
      "someFlowSpecAttribute": ""
    }
  ],
  "requestedQoS": {
    "applicationClass": "",
  }
}

```

```

    "averageBandwidth": 0,
    "maxBandwidth": 0,
    "minBandwidth": 0
  },
  "grantedQoS": {
    "actualClass": "",
    "dscp": 0,
    "actualBandwidth": 0
  },
  "realm": "",
  "name": "",
  "id": "",
  "type": "",
  "alias": [
    {
      "name": "",
      "value": ""
    }
  ],
  "href": "",
  "association": [
    {
      "name": "",
      "role": "",
      "href": "",
      "validFor": ""
    }
  ],
  "resourceCharacteristic": [
    {
      "name": "",
      "value": ""
    }
  ],
  "resourceSpecification": [
    {
      "href": "",
      "role": ""
    }
  ]
}

```

---

## FIELD DESCRIPTIONS

Field	Description
<b>flowSpecElement</b>	An array of the FlowSpecElement that are supporting the PolicyElement.
<b>requestedQoS</b>	The QoS that was requested.
<b>grantedQoS</b>	The QoS that was granted.
<b>realm</b>	The realm.
<b>id</b>	The unique identifier that is used for identification of the ManagedElement resources in the API.
<b>type</b>	Const string "managedElement".
<b>name</b>	The common name of the managed element.
<b>description</b>	Not supported in this release. Set to null.
<b>lifeCycleState</b>	The life cycle state of the ME. Defaults to NULL.

---

<b>alias</b>	Contains the alternative identifiers for the ME such as GUI name, TID, MTOSI name.
<b>assoication</b>	List of assoications to other REST managed resources. In this release only contains one relationship. That of the managed element to its AlarmCounts SessionElement.
<b>isBundle</b>	Always false in this release,
<b>bundle</b>	List of bundled REST managed resources. Defaults to NULL
<b>resourceSpecification</b>	An href to a SessionElement specification. SessionElement specifications are not support in this release so this will always will be NULL.
<b>characteristic</b>	A list of values that are valid for the instance of the ME. These values can come from definitions in characteristic specifications (not supported in this release) or ad-hoch. It will also be used to hold informaion the underlying system has about the managed elements that is useful to clients and that does not have an attribute in the managed element resorce.

---

## API OPERATION TEMPLATES

For every single of operation on the entities use the following templates and provide sample REST requests and responses.

Remember that the following Uniform Contract rules must be used :

Operation on Entities	Uniform API Operation	Description
Query Entities	GET SessionElement GET PolicyElement	GET must be used to retrieve a representation of a managed resource.
Create Entity	POST SessionElement POST PolicyElement	POST must be used to create a new SessionElement
Partial Update of an Entity	PATCH SessionElement PATCH PolicyElement	PATCH must be used to partially update a SessionElement
Complete Update of an Entity	PUT SessionElement PUT PolicyElement	PUT must be used to completely update a SessionElement identified by its SessionElement URI
Remove an Entity	DELETE SessionElement DELETE PolicyElement	DELETE must be used to remove a SessionElement

Filtering and attribute selection rules are described in the REST Design Guidelines.

Notifications are also described in a subsequent section.

## GET /API/NETWORK/SESSIONELEMENT

This operation is used to get a collection of SessionElements.

Supported Query Parameters: **query, sort, fields, depth**

REQUEST
GET /api/network/sessionElement Accept: application/json
RESPONSE
200 Content-Type: application/json Content-Range:0-1/2 Items  [ { "id": "1", "name": "", "type": "SessionElement", "alias": [ { "name": "", "value": "" } ], "resourceCharacteristic": [ { "name": "", "value": "" } ], "resourceSpecification": [ { "href": "", "role": "" } ], "association": [ { "name": "", "role": "", "href": "", "validFor": "" } ], "href": "http://localhost:8080/api/sessionElement/1", "mediaElement": [ { "flowElement": { "ipAddressType": "", "sourceIpAddress": "", "sourceIpPort": "", "destinationIpAddress": "", } } ] } ]

```

        "destinationIpPort": "",
        "transportType": "",
        "flowId": ""
    },
    "grantedQoS": {
        "actualClass": "",
        "dscp": 0,
        "actualBandwidth": 0
    },
    "requestedQoS": {
        "applicationClass": "",
        "averageBandwidth": 0,
        "maxBandwidth": 0,
        "minBandwidth": 0
    },
    "ageOutTimer": 0
}
],
"startTime": "",
"userElement": [
    {
        "name": "",
        "id": "",
        "realm": ""
    }
],
"groupId": ""
},
{
    "id": "2",
    "name": "",
    "type": "SessionElement",
    "alias": [
        {
            "name": "",
            "value": ""
        }
    ],
    "resourceCharacteristic": [
        {
            "name": "",
            "value": ""
        }
    ],
    "resourceSpecification": [
        {
            "href": "",
            "role": ""
        }
    ],
    "association": [
        {
            "name": "",
            "role": "",
            "href": "",
            "validFor": ""
        }
    ],
    "href": "http://localhost:8080/api/sessionElement/2",

```

```
    "mediaElement": [
      {
        "flowElement": {
          "ipAddressType": "",
          "sourceIpAddress": "",
          "sourceIpPort": "",
          "destinationIpAddress": "",
          "destinationIpPort": "",
          "transportType": "",
          "flowId": ""
        },
        "grantedQoS": {
          "actualClass": "",
          "dscp": 0,
          "actualBandwidth": 0
        },
        "requestedQoS": {
          "applicationClass": "",
          "averageBandwidth": 0,
          "maxBandwidth": 0,
          "minBandwidth": 0
        },
        "ageOutTimer": 0
      }
    ],
    "startTime": "",
    "userElement": [
      {
        "name": "",
        "id": "",
        "realm": ""
      }
    ],
    "groupId": ""
  }
]
```

## HEAD /API/NETWORK/SESSIONELEMENT

This operation is used to get the headers that would be returned by the corresponding GET.

Supported Query Parameters: **query**

REQUEST
HEAD /api/network/sessionElement Accept: application/json
RESPONSE
Content-Range: 0-1/2



## POST /API/NETWORK/SESSIONELEMENT

POST operations are used to create new instances of the resource.

### REQUEST

```
POST /api/network/sessionElement
Accept: application/json
```

```
{
  "name": "test",
  "groupId": "group12",
  "mediaElement": [
    {
      "flowElement": {
        "ipAddressType": "",
        "sourceIpAddress": "",
        "sourceIpPort": "",
        "destinationIpAddress": "",
        "destinationIpPort": "",
        "transportType": "",
        "flowId": ""
      },
      "grantedQoS": {
        "actualClass": "",
        "dscp": 0,
        "actualBandwidth": 0
      },
      "requestedQoS": {
        "applicationClass": "",
        "averageBandwidth": 0,
        "maxBandwidth": 0,
        "minBandwidth": 0
      },
      "ageOutTimer": 0
    }
  ],
  "startTime": "",
  "userElement": [
    {
      "name": "",
      "realm": ""
    }
  ]
}
```

### RESPONSE

```
200 OK
Location: http://localhost:8080/api/sessionElement/1
```

```
{
  "id": "1",
  "type": "sessionElement",
  "name": "test",
  "groupId": "group12",
```

```
"mediaElement": [
  {
    "flowElement": {
      "ipAddressType": "",
      "sourceIpAddress": "",
      "sourceIpPort": "",
      "destinationIpAddress": "",
      "destinationIpPort": "",
      "transportType": "",
      "flowId": ""
    },
    "grantedQoS": {
      "actualClass": "",
      "dscp": 0,
      "actualBandwidth": 0
    },
    "requestedQoS": {
      "applicationClass": "",
      "averageBandwidth": 0,
      "maxBandwidth": 0,
      "minBandwidth": 0
    },
    "ageOutTimer": 0
  }
],
"startTime": "",
"userElement": [
  {
    "name": "",
    "id": "",
    "realm": ""
  }
],
"href": "http://localhost:8080/api/sessionElement/1",
}
```

**PUT /API/NETWORK/SESSIONELEMENT**

Not Applicable

<b>REQUEST</b>
<code>PUT /api/network/sessionElement</code> <code>Accept: application/json</code>
<b>RESPONSE</b>
<code>405 Method Not Allowed</code>

**PATCH /API/NETWORK/SESSIONELEMENT**

Not Applicable

<b>REQUEST</b>
<code>PUT /api/network/sessionElement</code> <code>Accept: application/json</code>
<b>RESPONSE</b>
<code>405 Method Not Allowed</code>

**DELETE /API/NETWORK/SESSIONELEMENT**

Not Applicable

<b>REQUEST</b>
DELETE /api/network/sessionElement Accept: application/json
<b>RESPONSE</b>
405 Method Not Allowed



## GET /API/NETWORK/SESSIONELEMENT/{ID}

This operation is used to retrieve the specified SessionElement.

REQUEST
<pre>GET /api/network/sessionElement/2 Accept: application/json</pre>
RESPONSE
<pre>200 Content-Type: application/json ETag: "asdfadfasdfaera"  {   "id": "2",   "name": "",   "type": "SessionElement",   "alias": [     {       "name": "",       "value": ""     }   ],   "resourceCharacteristic": [     {       "name": "",       "value": ""     }   ],   "resourceSpecification": [     {       "href": "",       "role": ""     }   ],   "association": [     {       "name": "",       "role": "",       "href": "",       "validFor": ""     }   ],   "href": "http://localhost:8080/api/sessionElement/2",   "mediaelement": [     {       "flowElement": {         "ipAddressType": "",         "sourceIpAddress": "",         "sourceIpPort": "",         "destinationIpAddress": "",         "destinationIpPort": "",         "transportType": "",         "flowId": ""       }     }   ] }</pre>

```
    },
    "grantedQoS": {
      "actualClass": "",
      "dscp": 0,
      "actualBandwidth": 0
    },
    "requestedQoS": {
      "applicationClass": "",
      "averageBandwidth": 0,
      "maxBandwidth": 0,
      "minBandwidth": 0
    },
    "ageOutTimer": 0
  }
],
"startTime": "",
"userElement": [
  {
    "name": "",
    "id": "",
    "realm": ""
  }
],
"groupId": ""
}
```



**POST /API/NETWORK/SESSIONELEMENT/{ID}**

No Applicable

<b>REQUEST</b>
POST /api/network/sessionElement/1 Accept: application/json
<b>RESPONSE</b>
405 Method Not Allowed

**PUT /API/NETWORK/SESSIONELEMENT/{ID}**

PUT operations are used to replace one object instance with another. There is currently no use case which requires this.

<b>REQUEST</b>
PUT /api/network/sessionElement Accept: application/json
<b>RESPONSE</b>
405 Method Not Allowed

## PATCH /API/NETWORK/SESSIONELEMENT/{ID}

PATCH operations are used to modify resources.

### REQUEST

```
PATCH /api/network/sessionElement/1
Accept: application/json
```

```
{
  "name": "new name",
  "requestedQoS": {
    "applicationClass": "",
    "averageBandwidth": 3,
    "maxBandwidth": 4,
    "minBandwidth": 0
  }
}
```

### RESPONSE

```
200 OK
ETag: "asdfaetdfsrtfg"
```

```
{
  "id": "1",
  "type": "sessionElement",
  "name": "new name",
  "groupId": "group12",
  "mediaelement": [
    {
      "flowElement": {
        "ipAddressType": "",
        "sourceIpAddress": "",
        "sourceIpPort": "",
        "destinationIpAddress": "",
        "destinationIpPort": "",
        "transportType": "",
        "flowId": ""
      },
      "grantedQoS": {
        "actualClass": "",
        "dscp": 0,
        "actualBandwidth": 0
      },
      "requestedQoS": {
        "applicationClass": "",
        "averageBandwidth": 3,
        "maxBandwidth": 4,
        "minBandwidth": 0
      },
      "ageOutTimer": 0
    }
  ],
  "startTime": "",
  "userElement": [
```

```

    {
      "name": "",
      "id": "",
      "realm": ""
    }
  ],
  "href": "http://localhost:8080/api/sessionElement/1",
}

```

**REQUEST**

```

PATCH /api/network/sessionElement/1
Accept: application/json+patch

```

```

[
  { "op": "replace", "path": "/name", "value": "another new name" }
]

```

**RESPONSE**

```

200 OK
ETag: "dfertsdfawersdf"

```

```

{
  "id": "1",
  "type": "sessionElement",
  "name": "another new name",
  "groupId": "group12",
  "mediaelement": [
    {
      "flowElement": {
        "ipAddressType": "",
        "sourceIpAddress": "",
        "sourceIpPort": "",
        "destinationIpAddress": "",
        "destinationIpPort": "",
        "transportType": "",
        "flowId": ""
      },
      "grantedQoS": {
        "actualClass": "",
        "dscp": 0,
        "actualBandwidth": 0
      },
      "requestedQoS": {
        "applicationClass": "",
        "averageBandwidth": 0,
        "maxBandwidth": 0,
        "minBandwidth": 0
      },
      "ageOutTimer": 0
    }
  ],
  "startTime": "",
}

```

```
"userElement": [  
  {  
    "name": "",  
    "id": "",  
    "realm": ""  
  }  
],  
"href": "http://localhost:8080/api/sessionElement/1",  
}
```

**DELETE /API/NETWORK/SESSIONELEMENT/{ID}**

This operation causes the delete of the specified SessionElement.

<b>REQUEST</b>
DELETE /api/network/sessionElement/2 Accept: application/json
<b>RESPONSE</b>
204 No content

## GET /API/NETWORK/POLICYELEMENT

This operation is used to get a collection of PolicyElements.

Supported Query Parameters: **query, sort, fields, depth**

### REQUEST

```
GET /api/network/policyElement
Accept: application/json
```

### RESPONSE

```
200
Content-Type: application/json
Content-Range 0-0/1 Items
```

```
[
  {
    "id": "3",
    "name": "",
    "type": "PolicyElement",
    "alias": [
      {
        "name": "",
        "value": ""
      }
    ],
    "resourceCharacteristic": [
      {
        "name": "",
        "value": ""
      }
    ],
    "resourceSpecification": [
      {
        "href": "",
        "role": ""
      }
    ],
    "association": [
      {
```

```
        "name": "",
        "role": "",
        "href": "",
        "validFor": ""
    }
],
"href": "http://localhost:8080/api/network/policyElement/3",
"flowSpecElement": [
    {
        "someFlowSpecAttribute": ""
    }
],
"requestedQoS": {
    "applicationClass": "",
    "averageBandwidth": 0,
    "maxBandwidth": 0,
    "minBandwidth": 0
},
"grantedQoS": {
    "actualClass": "",
    "dscp": 0,
    "actualBandwidth": 0
},
"realm": ""
}
]
```



## HEAD /API/NETWORK/POLICYELEMENT

This operation is used to get the headers that would be returned by the corresponding GET without retrieving the PolicyElements itself.

Supported Query Parameters: **query**

REQUEST
HEAD /api/network/policyElement Accept: application/json
RESPONSE
Content-Range: 0-0/4

## POST /API/NETWORK/POLICYELEMENT

POST operations are used to create new instances of the resource.

### REQUEST

```
POST /api/network/policyElement
Accept: application/json
```

```
{
  "name": "myepe",
  "flowSpecElement": [
    {
      "someFlowSpecAttribute": ""
    }
  ],
  "requestedQoS": {
    "applicationClass": "",
    "averageBandwidth": 0,
    "maxBandwidth": 0,
    "minBandwidth": 0
  },
  "grantedQoS": {
    "actualClass": "",
    "dscp": 0,
    "actualBandwidth": 0
  },
  "realm": ""
}
```

### RESPONSE

```
200 OK
Location: http://localhost:8080/api/network/policyElement/3
```

```
{
  "id": "3",
  "name": "myepe",
  "type": "policyElement",
  "flowSpecElement": [
    {
      "someFlowSpecAttribute": ""
    }
  ],
  "requestedQoS": {
    "applicationClass": "",
    "averageBandwidth": 0,
    "maxBandwidth": 0,
    "minBandwidth": 0
  },
  "grantedQoS": {
    "actualClass": "",
    "dscp": 0,
    "actualBandwidth": 0
  },
  "realm": ""
}
```

```
"href": "http://localhost:8080/api/network/policyElement/3"  
}
```

**PUT /API/NETWORK/POLICYELEMENT**

Not Applicable

<b>REQUEST</b>
PUT /api/network/policyElement Accept: application/json
<b>RESPONSE</b>
405 Method Not Allowed

**PATCH /API/NETWORK/POLICYELEMENT**

Not Applicable

<b>REQUEST</b>
PUT /api/network/policyElement Accept: application/json
<b>RESPONSE</b>
405 Method Not Allowed

**DELETE /API/NETWORK/POLICYELEMENT**

Not Applicable

<b>REQUEST</b>
DELETE /api/network/policyElement Accept: application/json
<b>RESPONSE</b>
405 Method Not Allowed

**GET /API/NETWORK/POLICYELEMENT/{ID}**

This operation is used to retrieve the specified PolicyElement.

**REQUEST**

```
GET /api/network/policyElement/3
Accept: application/json
```

**RESPONSE**

```
200
Content-Type: application/json
ETag: "asdlkjadlhasdfh"
```

```
{
  "id": "3",
  "name": "",
  "type": "PolicyElement",
  "alias": [
    {
      "name": "",
      "value": ""
    }
  ],
  "resourceCharacteristic": [
    {
      "name": "",
      "value": ""
    }
  ],
  "resourceSpecification": [
    {
      "href": "",
      "role": ""
    }
  ],
  "association": [
    {
      "name": "",
      "role": "",

```

```
    "href": "",
    "validFor": ""
  }
],
"href": "http://localhost:8080/api/network/policyElement/3",
"flowSpecElement": [
  {
    "someFlowSpecAttribute": ""
  }
],
"requestedQoS": {
  "applicationClass": "",
  "averageBandwidth": 0,
  "maxBandwidth": 0,
  "minBandwidth": 0
},
"grantedQoS": {
  "actualClass": "",
  "dscp": 0,
  "actualBandwidth": 0
},
"realm": ""
}
```



**HEAD /API/NETWORK/POLICYELEMENT/{ID}**

This operation is used to get the headers that would be returned by the corresponding GET without having to retrieve the data.

<b>REQUEST</b>
HEAD /api/network/policyElement/2 Accept: application/json
<b>RESPONSE</b>
200 OK ETag: "jdfjffjhfhddf"

**POST /API/NETWORK/POLICYELEMENT/{ID}**

No Applicable

<b>REQUEST</b>
POST /api/network/policyElement/4 Accept: application/json
<b>RESPONSE</b>
405 Method Not Allowed

**PUT /API/NETWORK/POLICYELEMENT/{ID}**

Not Applicable

<b>REQUEST</b>
PUT /api/network/policyElement/3 Accept: application/json
<b>RESPONSE</b>
405 Method Not Allowed

**PATCH /API/NETWORK/POLICYELEMENT/{ID}**

PATCH operations are used to modify resources.

REQUEST
<pre>PATCH /api/network/policyElement/3 Accept: application/json  {   "name": "new name", }</pre>
RESPONSE
<pre>200 OK ETag: "asdfasfdfjhweklh"  {   "id": "3",   "name": "new name",   "type": "PolicyElement",   "flowSpecElement": [     {       "someFlowSpecAttribute": ""     }   ],   "requestedQoS": {     "applicationClass": "",     "averageBandwidth": 0,     "maxBandwidth": 0,     "minBandwidth": 0   },   "grantedQoS": {     "actualClass": "",     "dscp": 0,     "actualBandwidth": 0   },   "realm": ""   "href": "http://localhost:8080/api/network/policyElement/3" }</pre>

REQUEST
<pre>PATCH /api/network/policyElement/3 Accept: application/json+patch  [   { "op": "replace", "path": "/name", "value": "another new name" } ]</pre>
RESPONSE

```
200 OK
ETag: "asdfasdfkrweklh"

{
  "id": "3",
  "name": "another new name",
  "type": "PolicyElement",
  "flowSpecElement": [
    {
      "someFlowSpecAttribute": ""
    }
  ],
  "requestedQoS": {
    "applicationClass": "",
    "averageBandwidth": 0,
    "maxBandwidth": 0,
    "minBandwidth": 0
  },
  "grantedQoS": {
    "actualClass": "",
    "dscp": 0,
    "actualBandwidth": 0
  },
  "realm": ""
  "href": "http://localhost:8080/api/network/policyElement/3"
}
```

**DELETE /API/NETWORK/POLICYELEMENT/{ID}**

DELETE operations are not supported against /api/network/sessionElement.

**REQUEST**

```
DELETE /api/network/policyElement/3  
Accept: application/json
```

**RESPONSE**

```
204 No content
```

## NOTIFICATION

## RESOURCE CREATION NOTIFICATION

Used to notify that a resource has been created (showing a sessionElement case).

```
{
  "sequence": 1,
  "event": {
    "id": "1562231",
    "type": "SessionElementCreateNotification",
    "time": "2014-09-27T09:39:25.0Z",
    "SessionElement": {
      "id": "1",
      "type": "sessionElement",
      "name": "another new name",
      "groupId": "group12",
      "mediaelement": [
        {
          "flowElement": {
            "ipAddressType": "",
            "sourceIpAddress": "",
            "sourceIpPort": "",
            "destinationIpAddress": "",
            "destinationIpPort": "",
            "transportType": "",
            "flowId": ""
          },
          "grantedQoS": {
            "actualClass": "",
            "dscp": 0,
            "actualBandwidth": 0
          },
          "requestedQoS": {
            "applicationClass": "",
            "averageBandwidth": 0,
            "maxBandwidth": 0,
            "minBandwidth": 0
          },
          "ageOutTimer": 0
        }
      ],
      "startTime": "",
      "userElement": [
        {
          "name": "",
          "id": "",
          "realm": ""
        }
      ],
      "href": "http://localhost:8080/api/network/sessionElement/1"
    }
  }
}
```

## RESOURCE MODIFICATION NOTIFICATION

Used to notify that a resource has been modified (showing a sessionElement case).

```
{
  "sequence": 1,
  "event": {
    "id": "1562231",
    "type": "SessionElementModificationNotification",
    "time": "2014-09-27T09:39:25.0Z",
    "SessionElement": {
      "id": "1",
      "type": "sessionElement",
      "name": "another new name",
      "groupId": "group12",
      "mediaelement": [
        {
          "flowElement": {
            "ipAddressType": "",
            "sourceIpAddress": "",
            "sourceIpPort": "",
            "destinationIpAddress": "",
            "destinationIpPort": "",
            "transportType": "",
            "flowId": ""
          },
          "grantedQoS": {
            "actualClass": "",
            "dscp": 0,
            "actualBandwidth": 0
          },
          "requestedQoS": {
            "applicationClass": "",
            "averageBandwidth": 0,
            "maxBandwidth": 0,
            "minBandwidth": 0
          },
          "ageOutTimer": 0
        }
      ],
      "startTime": "",
      "userElement": [
        {
          "name": "",
          "id": "",
          "realm": ""
        }
      ],
      "href": "http://localhost:8080/api/network/sessionElement/1"
    }
  }
}
```

## RESOURCE DELETION NOTIFICATION

Used to notify that a resource has been deleted (showing a policyElement case)..



```
{
  "sequence": 1,
  "event": {
    "id": "1562231",
    "type": "PolicyElementDeleteNotification",
    "time": "2014-09-27T09:39:25.0Z",
    "PolicyElement": {
      "id": "77",
      "href": "http://server/api/network/policyElement/77",
    }
  }
}
```

## SWAGGER SPECIFICATION

### POLICYELEMENT

```

{
  "apiVersion": "1.0.0",
  "swaggerVersion": "1.2",
  "basePath": "http://localhost:8080/ONFNBIUIC/webresources",
  "resourcePath": "/api/policyElement",
  "apis": [
    {
      "path": "/api/policyElement/{id}",
      "operations": [
        {
          "method": "DELETE",
          "summary": "Delete the specified Link",
          "notes": "Delete the specified link.",
          "type": "void",
          "nickname": "remove",
          "parameters": [
            {
              "name": "id",
              "description": "link ID",
              "required": true,
              "type": "string",
              "paramType": "path",
              "allowMultiple": false
            }
          ],
          "responseMessages": [
            {
              "code": 204,
              "message": "No Content"
            },
            {
              "code": 404,
              "message": "Not found"
            },
            {
              "code": 500,
              "message": "Internal Server Error"
            }
          ]
        }
      ],
      {
        "method": "GET",
        "summary": "Retrieve a link by ID",
        "notes": "Retrieves a specific link by ID.",
        "type": "PolicyElement",
        "nickname": "findWithFields",
        "produces": [
          "application/json"
        ],
        "parameters": [
          {
            "name": "id",
            "description": "link ID",
            "required": true,

```

```

        "type": "string",
        "paramType": "path",
        "allowMultiple": false
    },
    {
        "name": "fields",
        "description": "Field Specification",
        "required": false,
        "type": "string",
        "paramType": "query",
        "allowMultiple": false
    }
],
"responseMessages": [
    {
        "code": 200,
        "message": "OK"
    },
    {
        "code": 404,
        "message": "Not found"
    },
    {
        "code": 500,
        "message": "Internal Server Error"
    }
]
},
{
    "method": "PATCH",
    "summary": "Modify a Link",
    "notes": "Modify the specified link.",
    "type": "PolicyElement",
    "nickname": "edit",
    "produces": [
        "application/json"
    ],
    "consumes": [
        "application/json"
    ],
    "parameters": [
        {
            "name": "id",
            "description": "link ID",
            "required": true,
            "type": "string",
            "paramType": "path",
            "allowMultiple": false
        },
        {
            "name": "body",
            "description": "Link Attributes to Modify",
            "required": true,
            "type": "PolicyElement",
            "paramType": "body",
            "allowMultiple": false
        }
    ],
    "responseMessages": [
        {

```

```

        "code": 200,
        "message": "OK"
    },
    {
        "code": 500,
        "message": "Internal Server Error"
    }
]
},
{
    "method": "PATCH",
    "summary": "Modify a Link",
    "notes": "Modify the specified link.",
    "type": "PolicyElement",
    "nickname": "patch",
    "produces": [
        "application/json"
    ],
    "consumes": [
        "application/json+patch"
    ],
    "parameters": [
        {
            "name": "id",
            "description": "link ID",
            "required": true,
            "type": "string",
            "paramType": "path",
            "allowMultiple": false
        },
        {
            "name": "body",
            "description": "Patch Request",
            "required": true,
            "type": "org.tmf.dsmapi.common.model.JsonPatch",
            "paramType": "body",
            "allowMultiple": false
        }
    ],
    "responseMessages": [
        {
            "code": 200,
            "message": "OK"
        },
        {
            "code": 500,
            "message": "Internal Server Error"
        }
    ]
},
{
    "method": "HEAD",
    "summary": "Retrieve the HTTP Header",
    "notes": "Retrieve the HTTP header that would have been returned
by the coresponding GET.",
    "type": "void",
    "nickname": "getHTTPHeadersForInstance",
    "produces": [
        "application/json"
    ],
}

```

```

    "parameters": [
      {
        "name": "_s",
        "description": "Query Specification",
        "required": false,
        "type": "string",
        "paramType": "query",
        "allowMultiple": false
      },
      {
        "name": "fields",
        "description": "Field Specification",
        "required": false,
        "type": "string",
        "paramType": "query",
        "allowMultiple": false
      }
    ],
    "responseMessages": [
      {
        "code": 204,
        "message": "No Data"
      },
      {
        "code": 400,
        "message": "Invalid Filter"
      },
      {
        "code": 404,
        "message": "Not found"
      },
      {
        "code": 500,
        "message": "Internal Server Error"
      }
    ]
  },
  {
    "path": "/api/policyElement",
    "operations": [
      {
        "method": "POST",
        "summary": "Create a Link",
        "notes": "Creates a link given the passed in representation.",
        "type": "PolicyElement",
        "nickname": "create",
        "produces": [
          "application/json"
        ],
        "consumes": [
          "application/json"
        ],
        "parameters": [
          {
            "name": "body",
            "description": "link to be created.",
            "required": true,
            "type": "PolicyElement",

```

```

        "paramType": "body",
        "allowMultiple": false
    },
    ],
    "responseMessages": [
        {
            "code": 201,
            "message": "Created"
        },
        {
            "code": 500,
            "message": "Internal Server Error"
        }
    ]
},
{
    "method": "HEAD",
    "summary": "Retrieve HTTP Header",
    "notes": "Retrieve the HTTP header that would have been returned
by the corresponding GET.",
    "type": "void",
    "nickname": "getHTTPHeadersForList",
    "produces": [
        "application/json"
    ],
    "parameters": [
        {
            "name": "_s",
            "description": "Query Specification",
            "required": false,
            "type": "string",
            "paramType": "query",
            "allowMultiple": false
        },
        {
            "name": "fields",
            "description": "Field Specification",
            "required": false,
            "type": "string",
            "paramType": "query",
            "allowMultiple": false
        }
    ],
    "responseMessages": [
        {
            "code": 204,
            "message": "No Data"
        },
        {
            "code": 400,
            "message": "Invalid Filter"
        },
        {
            "code": 500,
            "message": "Internal Server Error"
        }
    ]
},
{
    "method": "GET",

```

```

    "summary": "Retrieve Policy Elements",
    "notes": "Retrieve policy elements with option filter.",
    "type": "PolicyElement",
    "nickname": "findByCriteriaWithFields",
    "produces": [
      "application/json"
    ],
    "parameters": [
      {
        "name": "_s",
        "description": "Query Specification",
        "required": false,
        "type": "string",
        "paramType": "query",
        "allowMultiple": false
      },
      {
        "name": "fields",
        "description": "Field Specification",
        "required": false,
        "type": "string",
        "paramType": "query",
        "allowMultiple": false
      }
    ],
    "responseMessages": [
      {
        "code": 200,
        "message": "OK"
      },
      {
        "code": 400,
        "message": "Invalid Filter"
      },
      {
        "code": 500,
        "message": "Internal Server Error"
      }
    ]
  }
],
},
{
  "path": "/api/policyElement/{id}/graph",
  "operations": [
    {
      "method": "POST",
      "summary": "Creates a request for a graph based on the identified
link.",
      "notes": "Causes the creation of the specified graph.",
      "type": "Graph",
      "nickname": "createGraph",
      "produces": [
        "application/json"
      ],
      "consumes": [
        "application/json"
      ],
      "parameters": [
        {

```

```

        "name": "id",
        "description": "Link ID",
        "required": true,
        "type": "string",
        "paramType": "path",
        "allowMultiple": false
    },
    {
        "name": "body",
        "description": "The definition of the graph to be
created.",
        "required": true,
        "type": "GraphTask",
        "paramType": "body",
        "allowMultiple": false
    }
],
"responseMessages": [
    {
        "code": 201,
        "message": "Created"
    },
    {
        "code": 500,
        "message": "Internal Server Error"
    }
]
}
]
}
],
"models": {
    "ExplicitNameValuePair": {
        "id": "ExplicitNameValuePair",
        "properties": {
            "name": {
                "type": "string"
            },
            "value": {
                "type": "string"
            }
        }
    },
    "Href": {
        "id": "Href",
        "properties": {
            "href": {
                "type": "string"
            },
            "role": {
                "type": "string"
            }
        }
    },
    "PolicyElement": {
        "id": "PolicyElement",
        "properties": {
            "flowSpecElement": {
                "type": "array",
                "items": {

```



```

        "$ref": "FlowSpecElement"
    },
    "requestedQoS": {
        "$ref": "RequestedQoS"
    },
    "grantedQoS": {
        "$ref": "GrantedQoS"
    },
    "realm": {
        "type": "string"
    },
    "name": {
        "type": "string"
    },
    "id": {
        "type": "string"
    },
    "type": {
        "type": "string"
    },
    "alias": {
        "type": "array",
        "items": {
            "$ref": "ExplicitNameValuePair"
        }
    },
    "association": {
        "type": "array",
        "items": {
            "$ref": "Assoication"
        }
    },
    "href": {
        "type": "string"
    },
    "resourceCharacteristic": {
        "type": "array",
        "items": {
            "$ref": "ExplicitNameValuePair"
        }
    },
    "resourceSpecification": {
        "type": "array",
        "items": {
            "$ref": "Href"
        }
    }
},
"Graph": {
    "id": "Graph",
    "properties": {
        "association": {
            "$ref": "Array[Assoication]"
        },
    },
    "data": {
        "type": "array",
        "items": {
            "$ref": "Object"
        }
    }
}

```

```

    }
  }
},
"AssociationAndRole": {
  "id": "AssociationAndRole",
  "properties": {
    "attribute": {
      "type": "string"
    },
    "role": {
      "type": "array",
      "items": {
        "type": "string"
      }
    }
  }
},
"Association": {
  "id": "Association",
  "properties": {
    "name": {
      "type": "string"
    },
    "role": {
      "type": "string"
    },
    "href": {
      "type": "string"
    },
    "validFor": {
      "type": "string"
    }
  }
},
"RequestedQoS": {
  "id": "RequestedQoS",
  "properties": {
    "applicationClass": {
      "type": "string"
    },
    "averageBandwidth": {
      "type": "integer",
      "format": "int32"
    },
    "maxBandwidth": {
      "type": "integer",
      "format": "int32"
    },
    "minBandwidth": {
      "type": "integer",
      "format": "int32"
    }
  }
},
"FlowSpecElement": {
  "id": "FlowSpecElement",
  "properties": {
    "someFlowSpecAttribute": {
      "type": "string"
    }
  }
}

```

```

    }
  },
  "GrantedQoS": {
    "id": "GrantedQoS",
    "properties": {
      "actualClass": {
        "type": "string"
      },
      "dscp": {
        "type": "integer",
        "format": "int32"
      },
      "actualBandwidth": {
        "type": "integer",
        "format": "int32"
      }
    }
  },
  "GraphTask": {
    "id": "GraphTask",
    "properties": {
      "associationAndRole": {
        "type": "array",
        "items": {
          "$ref": "AssoicationAndRole"
        }
      },
      "field": {
        "type": "array",
        "items": {
          "type": "string"
        }
      },
      "zendhref": {
        "type": "string"
      },
      "depth": {
        "type": "integer",
        "format": "int32"
      }
    }
  }
}

```

## SESSIONELEMENT

```

{
  "apiVersion": "1.0.0",
  "swaggerVersion": "1.2",
  "basePath": "http://localhost:8080/ONFNBIUIC/webresources",
  "resourcePath": "/api/sessionElement",
  "apis": [
    {
      "path": "/api/sessionElement/{id}",
      "operations": [
        {

```

```

"method": "DELETE",
"summary": "Delete the specified session element",
"notes": "Delete the specified session element",
"type": "void",
"nickname": "remove",
"parameters": [
  {
    "name": "id",
    "description": "Node ID",
    "required": true,
    "type": "string",
    "paramType": "path",
    "allowMultiple": false
  }
],
"responseMessages": [
  {
    "code": 204,
    "message": "No Content"
  },
  {
    "code": 404,
    "message": "Not found"
  },
  {
    "code": 500,
    "message": "Internal Server Error"
  }
]
},
{
"method": "GET",
"summary": "Retrieve a session element by ID",
"notes": "Retrieves a specific session element by ID.",
"type": "SessionElement",
"nickname": "findWithFields",
"produces": [
  "application/json"
],
"parameters": [
  {
    "name": "id",
    "description": "Node ID",
    "required": true,
    "type": "string",
    "paramType": "path",
    "allowMultiple": false
  },
  {
    "name": "fields",
    "description": "Field Specification",
    "required": false,
    "type": "string",
    "paramType": "query",
    "allowMultiple": false
  }
],
"responseMessages": [
  {
    "code": 200,

```

```

        "message": "OK"
      },
      {
        "code": 404,
        "message": "Not found"
      },
      {
        "code": 500,
        "message": "Internal Server Error"
      }
    ]
  },
  {
    "method": "PATCH",
    "summary": "Modify a session element",
    "notes": "Modify the specified session element.",
    "type": "SessionElement",
    "nickname": "edit",
    "produces": [
      "application/json"
    ],
    "consumes": [
      "application/json+patch"
    ],
    "parameters": [
      {
        "name": "id",
        "description": "Node ID",
        "required": true,
        "type": "string",
        "paramType": "path",
        "allowMultiple": false
      },
      {
        "name": "body",
        "required": false,
        "type": "org.tmf.dsmapi.common.model.JsonPatch",
        "paramType": "body",
        "allowMultiple": false
      }
    ],
    "responseMessages": [
      {
        "code": 200,
        "message": "OK"
      },
      {
        "code": 500,
        "message": "Internal Server Error"
      }
    ]
  },
  {
    "method": "HEAD",
    "summary": "Retrieve the HTTP Header",
    "notes": "Retrieve the HTTP header that would have been returned by the corresponding GET.",
    "type": "void",
    "nickname": "getHTTPHeadersForInstance",
    "produces": [

```

```

        "application/json"
    ],
    "parameters": [
        {
            "name": "_s",
            "description": "Query Specification",
            "required": false,
            "type": "string",
            "paramType": "query",
            "allowMultiple": false
        },
        {
            "name": "fields",
            "description": "Field Specification",
            "required": false,
            "type": "string",
            "paramType": "query",
            "allowMultiple": false
        }
    ],
    "responseMessages": [
        {
            "code": 204,
            "message": "No Data"
        },
        {
            "code": 400,
            "message": "Invalid Filter"
        },
        {
            "code": 404,
            "message": "Not found"
        },
        {
            "code": 500,
            "message": "Internal Server Error"
        }
    ]
}
]
},
{
    "path": "/api/sessionElement",
    "operations": [
        {
            "method": "POST",
            "summary": "Create a session element",
            "notes": "Creates a session element given the passed in
representation.",
            "type": "SessionElement",
            "nickname": "create",
            "produces": [
                "application/json"
            ],
            "consumes": [
                "application/json"
            ],
            "parameters": [
                {
                    "name": "body",

```

```

        "description": "Node to be created.",
        "required": true,
        "type": "SessionElement",
        "paramType": "body",
        "allowMultiple": false
    },
],
"responseMessages": [
    {
        "code": 201,
        "message": "Created"
    },
    {
        "code": 500,
        "message": "Internal Server Error"
    }
]
},
{
    "method": "HEAD",
    "summary": "Retrieve the HTTP Header",
    "notes": "Retrieve the HTTP header that would have been returned
by the coresponding GET.",
    "type": "void",
    "nickname": "getHTTPHeadersForList",
    "produces": [
        "application/json"
    ],
    "parameters": [
        {
            "name": "_s",
            "description": "Query Specification",
            "required": false,
            "type": "string",
            "paramType": "query",
            "allowMultiple": false
        },
        {
            "name": "fields",
            "description": "Field Specification",
            "required": false,
            "type": "string",
            "paramType": "query",
            "allowMultiple": false
        }
    ],
    "responseMessages": [
        {
            "code": 204,
            "message": "No Data"
        },
        {
            "code": 400,
            "message": "Invalid Filter"
        },
        {
            "code": 500,
            "message": "Internal Server Error"
        }
    ]
}
]

```

```

    },
    {
      "method": "GET",
      "summary": "Retrieve session elements",
      "notes": "Retrieve session elements using option filter.",
      "type": "SessionElement",
      "nickname": "findByCriteriaWithFields",
      "produces": [
        "application/json"
      ],
      "parameters": [
        {
          "name": "_s",
          "description": "Query Specification",
          "required": false,
          "type": "string",
          "paramType": "query",
          "allowMultiple": false
        },
        {
          "name": "fields",
          "description": "Field Specification",
          "required": false,
          "type": "string",
          "paramType": "query",
          "allowMultiple": false
        }
      ],
      "responseMessages": [
        {
          "code": 200,
          "message": "OK"
        },
        {
          "code": 400,
          "message": "Invalid Filter"
        },
        {
          "code": 500,
          "message": "Internal Server Error"
        }
      ]
    }
  ],
},
{
  "path": "/api/sessionElement/{id}/graph",
  "operations": [
    {
      "method": "POST",
      "summary": "Creates a request for a graph based on the identified
session element.",
      "notes": "Causes the creation of the specified graph.",
      "type": "Graph",
      "nickname": "createGraph",
      "produces": [
        "application/json"
      ],
      "consumes": [
        "application/json"
      ]
    }
  ]
}

```



```

    ],
    "parameters": [
      {
        "name": "id",
        "description": "Node ID",
        "required": true,
        "type": "string",
        "paramType": "path",
        "allowMultiple": false
      },
      {
        "name": "body",
        "description": "The definition of the graph to be
created.",
        "required": true,
        "type": "GraphTask",
        "paramType": "body",
        "allowMultiple": false
      }
    ],
    "responseMessages": [
      {
        "code": 201,
        "message": "Created"
      },
      {
        "code": 500,
        "message": "Internal Server Error"
      }
    ]
  }
]
}
],
"models": {
  "UserElement": {
    "id": "UserElement",
    "properties": {
      "name": {
        "type": "string"
      },
      "id": {
        "type": "string"
      },
      "realm": {
        "type": "string"
      }
    }
  },
  "ExplicitNameValuePair": {
    "id": "ExplicitNameValuePair",
    "properties": {
      "name": {
        "type": "string"
      },
      "value": {
        "type": "string"
      }
    }
  }
},
},

```

```

"Href": {
  "id": "Href",
  "properties": {
    "href": {
      "type": "string"
    },
    "role": {
      "type": "string"
    }
  }
},
"FlowElement": {
  "id": "FlowElement",
  "properties": {
    "ipAddressType": {
      "type": "string"
    },
    "sourceIpAddress": {
      "type": "string"
    },
    "sourceIpPort": {
      "type": "string"
    },
    "destinationIpAddress": {
      "type": "string"
    },
    "destinationIpPort": {
      "type": "string"
    },
    "transportType": {
      "type": "string"
    },
    "flowId": {
      "type": "string"
    }
  }
},
"Graph": {
  "id": "Graph",
  "properties": {
    "assoication": {
      "$ref": "Array[Assoication]"
    },
    "data": {
      "type": "array",
      "items": {
        "$ref": "Object"
      }
    }
  }
},
"AssoicationAndRole": {
  "id": "AssoicationAndRole",
  "properties": {
    "attribute": {
      "type": "string"
    },
    "role": {
      "type": "array",
      "items": {

```

```

        "type": "string"
      }
    }
  },
  "MediaElement": {
    "id": "MediaElement",
    "properties": {
      "flowElement": {
        "$ref": "FlowElement"
      },
      "grantedQoS": {
        "$ref": "GrantedQoS"
      },
      "requestedQoS": {
        "$ref": "RequestedQoS"
      },
      "ageOutTimer": {
        "type": "integer",
        "format": "int32"
      }
    }
  },
  "Association": {
    "id": "Association",
    "properties": {
      "name": {
        "type": "string"
      },
      "role": {
        "type": "string"
      },
      "href": {
        "type": "string"
      },
      "validFor": {
        "type": "string"
      }
    }
  },
  "RequestedQoS": {
    "id": "RequestedQoS",
    "properties": {
      "applicationClass": {
        "type": "string"
      },
      "averageBandwidth": {
        "type": "integer",
        "format": "int32"
      },
      "maxBandwidth": {
        "type": "integer",
        "format": "int32"
      },
      "minBandwidth": {
        "type": "integer",
        "format": "int32"
      }
    }
  }
},

```

```

"SessionElement": {
  "id": "SessionElement",
  "properties": {
    "mediaelement": {
      "type": "array",
      "items": {
        "$ref": "MediaElement"
      }
    },
    "startTime": {
      "type": "string"
    },
    "userElement": {
      "type": "array",
      "items": {
        "$ref": "UserElement"
      }
    },
    "groupId": {
      "type": "string"
    },
    "name": {
      "type": "string"
    },
    "id": {
      "type": "string"
    },
    "type": {
      "type": "string"
    },
    "alias": {
      "type": "array",
      "items": {
        "$ref": "ExplicitNameValuePair"
      }
    },
    "association": {
      "type": "array",
      "items": {
        "$ref": "Assoication"
      }
    },
    "href": {
      "type": "string"
    },
    "resourceCharacteristic": {
      "type": "array",
      "items": {
        "$ref": "ExplicitNameValuePair"
      }
    },
    "resourceSpecification": {
      "type": "array",
      "items": {
        "$ref": "Href"
      }
    }
  }
},
"GrantedQoS": {

```

```

    "id": "GrantedQoS",
    "properties": {
      "actualClass": {
        "type": "string"
      },
      "dscp": {
        "type": "integer",
        "format": "int32"
      },
      "actualBandwidth": {
        "type": "integer",
        "format": "int32"
      }
    }
  },
  "GraphTask": {
    "id": "GraphTask",
    "properties": {
      "associationAndRole": {
        "type": "array",
        "items": {
          "$ref": "AssoicationAndRole"
        }
      },
      "field": {
        "type": "array",
        "items": {
          "type": "string"
        }
      },
      "zendhref": {
        "type": "string"
      },
      "depth": {
        "type": "integer",
        "format": "int32"
      }
    }
  }
}

```

## OPEN ITEMS

Item Number	Description	Resolution

## AUTHORS AND CONTRIBUTORS

Name	Company	Email-Address
John Morey Author	Ciena	John Morey jmorey@mmintl.com
Nigel Davis Contributor	Ciena	Davis, Nigel ndavis@ciena.com
Pascal Menezes Vice-Chair and Contributor	Microsoft	Menezes, Pascal pmeneze@microsoft.com

## RELEASE HISTORY

Release Number	Date	Release led by:	Description
Version 1	March 2015	Pascal Menezes	Real Time Media NBI REST