VTN Coordinator
Web API Reference

Contents

Chapter 1

Web API reference

This chapter lists the following details of Web API functions:

- Request method
- Request URI
- Sample request and response body of Web API function
- Parameter descriptions

1.1. API Version function

This section lists the API Version function.

1.1.1. Show API Version

This operation is used to view the API version information.

Processing request

Method	GET

Request URI

- XML format

  /api_version.xml
- JSON format

  /api_version.json

Request body None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
  standalone="yes"?>
<api_version version="{version}" />
```

- JSON format

```
{
    "api_version": {
        "version": "{version}"
    }
}
```

Table 1: Description of response elements

Element	Description
version	API version.

7

Element	Description
	Valid value: Vn.n where *n* is a positive integer.

1.2. Coordinator Version functions

This section lists the coordinator version function.

1.2.1. Show Coordinator Version

This operation is used to view the coordinator version information.

Processing request

Method	GET
Request URI	• XML format /coordinator_version.xml • JSON format /coordinator_version.json
Request body	None

Processing result

Response body

• XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<coordinator_version version="{version}">
    <patches>
        <patch patch_no="{patch_no}" />
    </patches>
</version>
```

• JSON format

```
{
    "coordinator_version": {
        "version": "{version}",
        "patches": [
            {
                "patch_no": "{patch_no}"
            }
        ]
    }
}
```

Table 2: Description of response elements

Element	Description
version	Displays the version of coordinator in the "Va.b.c.d" format, where a - major, b - minor, c - revision, d - patch level.

8

Element	Description
	Valid range for a,b,c,d: 0 - 65535
patch_no	Lists the patch numbers applied to the coordinator. Valid range: 1 - 255.

Set the configuration options:

- Set foo to bar
- Set your blink rate
- Do some other stuff
- Do a special thing for Linux

1.3. Flow List functions

This section lists the Flow List functions.

1.3.1. Create Flow List

This operation is used to create Flow List.

Processing request

Method   POST

Request URI
- XML format

  /flowlists.xml
- JSON format

  /flowlists.json

Request body
- XML format

```
<flowlist fl_name="{fl_name}"
  ip_version="{ip_version}" />
```

- JSON format

```
{
    "flowlist": {
        "fl_name": "{fl_name}",
        "ip_version": "{ip_version}"
    }
}
```

Table 3: Description of request elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

9

Element	Description	Required
	Note: Flow List name is case sensitive.	
ip_version	IP version. Valid value: IP (default), IPv6 Note: IP version is case insensitive.	No

Processing result

Response body   None

1.3.2. Delete Flow List

This operation is used to delete Flow List.

Processing request

Method   DELETE

Request URI
- XML format

    /flowlists/{fl_name}.xml
- JSON format

    /flowlists/{fl_name}.json

Table 4: Description of request URI elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember

Flow List that is set in *fl_name* of Flow Filter cannot be deleted.

Processing result

Response body   None

1.3.3. List Flow Lists

This operation is used to list Flow List information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

/flowlists.xml

/flowlists/detail.xml

/flowlists/count.xml

- JSON format

/flowlists.json

/flowlists/detail.json

/flowlists/count.json

- Query string

?index={fl_name}&max_repetition={max_repetition}

&ip_version={ip_version}

Table 5: Description of request URI elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: Flow List name is case sensitive.	No
max_repetition	Number of the resources that are returned. Valid value: Decimal integer Valid range: 1 to MAX of UINT32. Default is 10000.	No
ip_version	IP version. Valid value: IP (default), IPv6 Note: IP version is case insensitive.	No

Request body

None

Processing result

Response body

- XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowlists>
    <flowlist fl_name="{fl_name}" />
</flowlists>
```

11

If detail is specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowlists>
    <flowlist fl_name="{fl_name}"
 ip_version="{ip_version}" />
</flowlists>
```

If count is specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowlists count="{count}" />
```

- JSON format

    If count is not specified in URI

```json
{
    "flowlists": [
        {
            "fl_name": "{fl_name}"
        }
    ]
}
```

    If detail is specified in URI

```json
{
    "flowlists": [
        {
            "fl_name": "{fl_name}",
            "ip_version": "{ip_version}"
        }
    ]
}
```

    If count is specified in URI

```json
{
    "flowlists": {
        "count": "{count}"
    }
}
```

Table 6: Description of response elements

Element	Description
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: Flow List name is case sensitive.
ip_version	IP version.

12

Element	Description
	Valid value: IP (default), IPv6
	Note: IP version is case insensitive.
count	Number of Flow List.
	Valid value: Decimal integer

1.3.4. Show Flow List

This operation is used to view a specific Flow List information.

Processing request

Method GET

Request URI
- XML format

  /flowlists/{fl_name}.xml
- JSON format

  /flowlists/{fl_name}.json
- Query string

  ?ip_version={ip_version}

Table 7: Description of request URI elements

Element	Description	Required
fl_name	Flow List name.	Yes
	Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).	
	Note: Flow List name is case sensitive.	
ip_version	IP version.	No
	Valid value: IP (default), IPv6	
	Note: IP version is case insensitive.	

Request body None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowlist fl_name="{fl_name}"
 ip_version="{ip_version}" />
```

- JSON format

```
{
    "flowlist": {
        "fl_name": "{fl_name}",
        "ip_version": "{ip_version}"
    }
}
```

Table 8: Description of response elements

Element	Description
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: Flow List name is case sensitive.
ip_version	IP version. Valid value: IP (default), IPv6 Note: IP version is case insensitive.

1.4. Flow List Entry functions

This section lists the Flow List Entry functions.

1.4.1. Create Flow List Entries

This operation is used to create a Flow List entry.

Processing request

Method	POST
Request URI	

- XML format

    /flowlists/{fl_name}/flowlistentries.xml
- JSON format

    /flowlists/{fl_name}/flowlistentries.json

Table 9: Description of request URI elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<flowlistentry seqnum="{seqnum}"
 macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}"
 macethertype="{macethertype}"
 macvlanpriority="{macvlanpriority}"
 ipdstaddr="{ipdstaddr}"
 ipdstaddrprefix="{ipdstaddrprefix}"
 ipsrcaddr="{ipsrcaddr}"
 ipsrcaddrprefix="{ipsrcaddrprefix}"
 ipv6dstaddr="{ipv6dstaddr}"
 ipv6dstaddrprefix="{ipv6dstaddrprefix}"
 ipv6srcaddr="{ipv6srcaddr}"
 ipv6srcaddrprefix="{ipv6srcaddrprefix}"
 ipproto="{ipproto}" ipdscp="{ipdscp}"
 l4dstport="{l4dstport}"
 l4dstendport="{l4dstendport}"
 l4srcport="{l4srcport}"
 l4srcendport="{l4srcendport}"
 icmptypenum="{icmptypenum}"
 icmpcodenum="{icmpcodenum}"
 ipv6icmptypenum="{ipv6icmptypenum}"
 ipv6icmpcodenum="{ipv6icmpcodenum}" />
```

- JSON format

```
{
    "flowlistentry": {
        "seqnum": "{seqnum}",
        "macdstaddr": "{macdstaddr}",
        "macsrcaddr": "{macsrcaddr}",
        "macethertype": "{macethertype}",
        "macvlanpriority": "{macvlanpriority}",
        "ipdstaddr": "{ipdstaddr}",
        "ipdstaddrprefix": "{ipdstaddrprefix}",
        "ipsrcaddr": "{ipsrcaddr}",
        "ipsrcaddrprefix": "{ipsrcaddrprefix}",
        "ipv6dstaddr": "{ipv6dstaddr}",
        "ipv6dstaddrprefix":
 "{ipv6dstaddrprefix}",
        "ipv6srcaddr": "{ipv6srcaddr}",
        "ipv6srcaddrprefix":
 "{ipv6srcaddrprefix}",
        "ipproto": "{ipproto}",
        "ipdscp": "{ipdscp}",
        "l4dstport": "{l4dstport}",
        "l4dstendport": "{l4dstendport}",
        "l4srcport": "{l4srcport}",
        "l4srcendport": "{l4srcendport}",
        "icmptypenum": "{icmptypenum}",
        "icmpcodenum": "{icmpcodenum}",
        "ipv6icmptypenum": "{ipv6icmptypenum}",
        "ipv6icmpcodenum": "{ipv6icmpcodenum}"
    }
}
```

Table 10: Description of request elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes
macdstaddr	The MAC destination address. Valid value: Three groups of four hexadecimal digits separated by dots (.). Example: 0123.4567.89ab Note: MAC address is case insensitive.	No
macsrcaddr	The transmission source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.). Example: 0123.4567.89ab Note: MAC address is case insensitive.	No
macethertype	The Ether type of the Ethernet frame Valid value: A hexadecimal number. Valid range: 0x0000 - 0xffff	No
macvlanpriority	The VLAN priority number tag. Valid value: A decimal integer. Valid range: 0 - 7	No
ipdstaddr	The destination IP address. Valid value: IPv4 dot-separated format Example: 192.168.1.1 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IP.	No
ipdstaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 32 Note: This parameter is required if *ipdstaddr* is specified.	No
ipsrcaddr	The transmission source IP address.	No

Element	Description	Required
	Valid value: IPv4 dot-separated format. Example: 192.168.1.1 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IP.	
ipsrcaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 32 Note: This parameter is required if *ipsrcaddr* is specified.	No
ipv6dstaddr	The destination IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2: 288a:1fc0:0001:10ee Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IPv6.	No
ipv6dstaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 128 Note: This parameter is required if *ipv6dstaddr* is specified.	No
ipv6srcaddr	The source IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2: 288a:1fc0:0001:10ee Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IPv6.	No
ipv6srcaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 128 Note: This parameter is required if *ipv6srcaddr* is specified.	No

Element	Description	Required
ipproto	The IP protocol number. Valid value: A decimal integer. Valid range: 1 - 255	No
ipdscp	The DSCP value. Valid value: A decimal integer. Valid range: 0 - 63	No
l4dstport	The TCP or UDP destination port number. If a range is specified, this is viewed as start port. Valid value: A decimal integer. Valid range: 0 - 65535	No
l4dstendport	The end point TCP or UDP port number. Valid value: A decimal integer. Valid range: 1 - 65535	No
l4srcport	The number of the TCP or UDP first source port. Valid value: A decimal integer. Valid range: 0 - 65535	No
l4srcendport	The end point TCP or UDP port number. Valid value: A decimal integer. Valid range: 1 - 65535	No
icmptypenum	The ICMP type value. Valid value: A decimal integer. Valid range: 0 - 255 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IP.	No
icmpcodenum	The ICMP code value. Valid value: A decimal integer. Valid range: 0 - 255 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IP.	No
ipv6icmptypenum	The ICMP type value.	No

18

Element	Description	Required
	Valid value: A decimal integer. Valid range: 0 - 255 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IPv6.	
ipv6icmpcodenum	The ICMP code value. Valid value: A decimal integer. Valid range: 0 - 255 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IPv6.	No

Processing result

Response body   None

1.4.2. Delete Flow List Entry

This operation is used to delete Flow List Entry.

Processing request

Method   DELETE

Request URI
- XML format

  /flowlists/{fl_name}/flowlistentries/{seqnum}.xml
- JSON format

  /flowlists/{fl_name}/flowlistentries/{seqnum}.json

Table 11: Description of request URI elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
seqnum	Sequence number. Valid value: An positive integer. Valid range: 1 - 65535	Yes

Request body   None

Remember

Flow List Entry under Flow List that is set in *fl_name* of Flow Filter cannot be deleted.

Processing result

Response body	None

1.4.3. Update Flow List Entry

This operation is used to update a Flow List entry.

Processing request

Method	PUT
Request URI	• XML format /flowlists/{fl_name}/flowlistentries/{seqnum}.xml • JSON format /flowlists/{fl_name}/flowlistentries/{seqnum}.json

Table 12: Description of request URI elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Request body   • XML format

```
<flowlistentry macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}"
 macethertype="{macethertype}"
 macvlanpriority="{macvlanpriority}"
 ipdstaddr="{ipdstaddr}"
 ipdstaddrprefix="{ipdstaddrprefix}"
 ipsrcaddr="{ipsrcaddr}"
 ipsrcaddrprefix="{ipsrcaddrprefix}"
 ipv6dstaddr="{ipv6dstaddr}"
 ipv6dstaddrprefix="{ipv6dstaddrprefix}"
 ipv6srcaddr="{ipv6srcaddr}"
 ipv6srcaddrprefix="{ipv6srcaddrprefix}"
 ipproto="{ipproto}" ipdscp="{ipdscp}"
 l4dstport="{l4dstport}"
 l4dstendport="{l4dstendport}"
 l4srcport="{l4srcport}"
 l4srcendport="{l4srcendport}"
 icmptypenum="{icmptypenum}"
 icmpcodenum="{icmpcodenum}"
 ipv6icmptypenum="{ipv6icmptypenum}"
 ipv6icmpcodenum="{ipv6icmpcodenum}" />
```

- JSON format

```
{
    "flowlistentry": {
        "macdstaddr": "{macdstaddr}",
        "macsrcaddr": "{macsrcaddr}",
        "macethertype": "{macethertype}",
        "macvlanpriority": "{macvlanpriority}",
        "ipdstaddr": "{ipdstaddr}",
        "ipdstaddrprefix": "{ipdstaddrprefix}",
        "ipsrcaddr": "{ipsrcaddr}",
        "ipsrcaddrprefix": "{ipsrcaddrprefix}",
        "ipv6dstaddr": "{ipv6dstaddr}",
        "ipv6dstaddrprefix":
 "{ipv6dstaddrprefix}",
        "ipv6srcaddr": "{ipv6srcaddr}",
        "ipv6srcaddrprefix":
 "{ipv6srcaddrprefix}",
        "ipproto": "{ipproto}",
        "ipdscp": "{ipdscp}",
        "l4dstport": "{l4dstport}",
        "l4dstendport": "{l4dstendport}",
        "l4srcport": "{l4srcport}",
        "l4srcendport": "{l4srcendport}",
        "icmptypenum": "{icmptypenum}",
        "icmpcodenum": "{icmpcodenum}",
        "ipv6icmptypenum": "{ipv6icmptypenum}",
        "ipv6icmpcodenum": "{ipv6icmpcodenum}"
    }
}
```

Table 13: Description of request elements

Element	Description	Required
macdstaddr	The MAC destination address. Valid value: Three groups of four hexadecimal digits separated by dots (.). Example: 0123.4567.89ab Note: MAC address is case insensitive.	No
macsrcaddr	The transmission source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.). Example: 0123.4567.89ab Note: MAC address is case insensitive.	No
macethertype	The Ether type of the Ethernet frame Valid value: A hexadecimal number.	No

Element	Description	Required
	Valid range: 0x0000 - 0xffff	
macvlanpriority	The VLAN priority number tag. Valid value: A decimal integer Valid range: 0 - 7	No
ipdstaddr	The destination IP address. Valid value: IPv4 dot-separated format Example: 192.168.1.1 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IP.	No
ipdstaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 32 Note: This parameter is required if *ipdstaddr* is specified.	No
ipsrcaddr	The transmission source IP address. Valid value: IPv4 dot-separated format Example: 192.168.1.1 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IP.	No
ipsrcaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 32 Note: This parameter is required if *ipsrcaddr* is specified.	No
ipv6dstaddr	The destination IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2: 288a:1fc0:0001:10ee Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IPv6.	No
ipv6dstaddrprefix	The prefix length. Valid value: A decimal integer.	No

22

Element	Description	Required
	Valid range: 1 - 128 Note: This parameter is required if *ipv6dstaddr* is specified.	
ipv6srcaddr	The source IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2: 288a:1fc0:0001:10ee Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IPv6.	No
ipv6srcaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 128 Note: This parameter is required if *ipv6srcaddr* is specified.	No
ipproto	The IP protocol number. Valid value: A decimal integer. Valid range: 1 - 255	No
ipdscp	The DSCP value. Valid value: A decimal integer. Valid range: 0 - 63	No
l4dstport	The TCP or UDP destination port number. If a range is specified, this is viewed as start port. Valid value: A decimal integer. Valid range: 0 - 65535	No
l4dstendport	The end point TCP or UDP port number. Valid value: A decimal integer. Valid range: 1 - 65535	No
l4srcport	The TCP or UDP source port number. Valid value: A decimal integer. Valid range: 0 - 65535	No

Element	Description	Required
l4srcendport	The end point TCP or UDP port number. Valid value: A decimal integer. Valid range: 1 - 65535	No
icmptypenum	The ICMP type value. Valid value: A decimal integer. Valid range: 0 - 255 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IP.	No
icmpcodenum	The ICMP code value. Valid value: A decimal integer. Valid range: 0 - 255 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IP.	No
ipv6icmptypenum	The ICMP type value. Valid value: A decimal integer. Valid range: 0 - 255 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IPv6.	No
ipv6icmpcodenum	The ICMP code value. Valid value: A decimal integer. Valid range: 0 - 255 Note: This parameter can be specified only when *ip_version* of Flow List specified in *fl_name* is IPv6.	No

Processing result

Response body   None

1.4.4. List Flow List Entries

This operation is used to list Flow List Entries information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

/flowlists/{fl_name}/flowlistentries.xml

/flowlists/{fl_name}/flowlistentries/detail.xml

/flowlists/{fl_name}/flowlistentries/count.xml

- JSON format

/flowlists/{fl_name}/flowlistentries.json

/flowlists/{fl_name}/flowlistentries/detail.json

/flowlists/{fl_name}/flowlistentries/count.json

- Query string

?index={seqnum}&max_repetition={max_repetition}

Table 14: Description of request URI elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 15: Description of query string elements

Element	Description	Required
seqnum	Sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowlistentries>
    <flowlistentry seqnum="{seqnum}" />
</flowlistentries>
```

If detail is specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowlistentries>
    <flowlistentry seqnum="{seqnum}"
 macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}"
 macethertype="{macethertype}"
 macvlanpriority="{macvlanpriority}"
 ipdstaddr="{ipdstaddr}"
 ipdstaddrprefix="{ipdstaddrprefix}"
 ipsrcaddr="{ipsrcaddr}"
 ipsrcaddrprefix="{ipsrcaddrprefix}"
 ipv6dstaddr="{ipv6dstaddr}"
 ipv6dstaddrprefix="{ipv6dstaddrprefix}"
 ipv6srcaddr="{ipv6srcaddr}"
 ipv6srcaddrprefix="{ipv6srcaddrprefix}"
 ipproto="{ipproto}" ipdscp="{ipdscp}"
 l4dstport="{l4dstport}"
 l4dstendport="{l4dstendport}"
 l4srcport="{l4srcport}"
 l4srcendport="{l4srcendport}"
 icmptypenum="{icmptypenum}"
 icmpcodenum="{icmpcodenum}"
 ipv6icmptypenum="{ipv6icmptypenum}"
 ipv6icmpcodenum="{ipv6icmpcodenum}" />
</flowlistentries>
```

If count is specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowlistentries count="{count}" />
```

- JSON format

    If count is not specified in URI

```json
{
    "flowlistentries": [
        {
            "seqnum": "{seqnum}"
        }
    ]
}
```

    If detail is specified in URI

```json
{
    "flowlistentries": [
        {
            "seqnum": "{seqnum}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}",
            "macethertype": "{macethertype}",
            "macvlanpriority":
 "{macvlanpriority}",
            "ipdstaddr": "{ipdstaddr}",
            "ipdstaddrprefix":
 "{ipdstaddrprefix}",
```

26

```
              "ipsrcaddr": "{ipsrcaddr}",
              "ipsrcaddrprefix":
"{ipsrcaddrprefix}",
              "ipv6dstaddr": "{ipv6dstaddr}",
              "ipv6dstaddrprefix":
"{ipv6dstaddrprefix}",
              "ipv6srcaddr": "{ipv6srcaddr}",
              "ipv6srcaddrprefix":
"{ipv6srcaddrprefix}",
              "ipproto": "{ipproto}",
              "ipdscp": "{ipdscp}",
              "l4dstport": "{l4dstport}",
              "l4dstendport": "{l4dstendport}",
              "l4srcport": "{l4srcport}",
              "l4srcendport": "{l4srcendport}",
              "icmptypenum": "{icmptypenum}",
              "icmpcodenum": "{icmpcodenum}",
              "ipv6icmptypenum":
"{ipv6icmptypenum}",
              "ipv6icmpcodenum":
"{ipv6icmpcodenum}"
          }
      ]
}
```

If count is specified in URI

```
{
    "flowlistentries": {
        "count": "{count}"
    }
}
```

Table 16: Description of response elements

Element	Description
seqnum	The sequence number.
	Valid value: A positive integer.
	Valid range: 1 - 65535
macdstaddr	The MAC destination address.
	Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab).
	Note: MAC address is case insensitive.
macsrcaddr	The transmission source MAC address.
	Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab).
	Note: MAC address is case insensitive.
macethertype	The Ether type of the Ethernet frame
	Valid value: A hexadecimal number.

Element	Description
	Valid range: 0x0000 - 0xffff
macvlanpriority	The VLAN priority number tag. Valid value: A decimal integer Valid range: 0 - 7
ipdstaddr	The destination IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
ipdstaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 32
ipsrcaddr	The transmission source IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
ipsrcaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 32
ipv6dstaddr	The destination IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2:288a:1fc0:0001:10ee
ipv6dstaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 128
ipv6srcaddr	The source IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2:288a:1fc0:0001:10ee
ipv6srcaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 128
ipproto	The IP protocol number. Valid value: A decimal integer.

28

Element	Description
	Valid range: 1 - 255
ipdscp	The DSCP value.
	Valid value: A decimal integer.
	Valid range: 0 - 63
l4dstport	The TCP or UDP destination port number.
	If a range is specified, this is viewed as start port.
	Valid value: A decimal integer.
	Valid range: 0 - 65535
l4dstendport	The end point TCP or UDP port number.
	Valid value: A decimal integer.
	Valid range: 1 - 65535
l4srcport	The TCP or UDP source port number.
	If a range is specified, this is viewed as l4SrcPortStart.
	Valid value: A decimal integer.
	Valid range: 0 - 65535
l4srcendport	The end point TCP or UDP port number.
	Valid value: A decimal integer.
	Valid range: 1 - 65535
icmptypenum	The ICMP type value.
	Valid value: A decimal integer.
	Valid range: 0 - 255
icmpcodenum	The ICMP code value.
	Valid value: A decimal integer.
	Valid range: 0 - 255
ipv6icmptypenum	The ICMP type value.
	Valid value: A decimal integer.
	Valid range: 0 - 255
ipv6icmpcodenum	The ICMP code value.
	Valid value: A decimal integer.
	Valid range: 0 - 255
count	Number of Flow List entry.
	Valid value: A decimal integer.

1.4.5. Show Flow List Entry

This operation is used to view a specific Flow List Entry information.

Processing request

Method	GET
Request URI	• XML format /flowlists/{fl_name}/flowlistentries/{seqnum}.xml • JSON format /flowlists/{fl_name}/flowlistentries/{seqnum}.json

Table 17: Description of request URI elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: Flow List name is case sensitive.	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Request body	None

Processing result

Response body	• XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowlistentry seqnum="{seqnum}"
 macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}"
 macethertype="{macethertype}"
 macvlanpriority="{macvlanpriority}"
 ipdstaddr="{ipdstaddr}"
 ipdstaddrprefix="{ipdstaddrprefix}"
 ipsrcaddr="{ipsrcaddr}"
 ipsrcaddrprefix="{ipsrcaddrprefix}"
 ipv6dstaddr="{ipv6dstaddr}"
 ipv6dstaddrprefix="{ipv6dstaddrprefix}"
 ipv6srcaddr="{ipv6srcaddr}"
 ipv6srcaddrprefix="{ipv6srcaddrprefix}"
 ipproto="{ipproto}" ipdscp="{ipdscp}"
 l4dstport="{l4dstport}"
 l4dstendport="{l4dstendport}"
 l4srcport="{l4srcport}"
 l4srcendport="{l4srcendport}"
 icmptypenum="{icmptypenum}"
```

```
icmpcodenum="{icmpcodenum}"
ipv6icmptypenum="{ipv6icmptypenum}"
ipv6icmpcodenum="{ipv6icmpcodenum}" />
```

- JSON format

```
{
    "flowlistentry": {
        "seqnum": "{seqnum}",
        "macdstaddr": "{macdstaddr}",
        "macsrcaddr": "{macsrcaddr}",
        "macethertype": "{macethertype}",
        "macvlanpriority": "{macvlanpriority}",
        "ipdstaddr": "{ipdstaddr}",
        "ipdstaddrprefix": "{ipdstaddrprefix}",
        "ipsrcaddr": "{ipsrcaddr}",
        "ipsrcaddrprefix": "{ipsrcaddrprefix}",
        "ipv6dstaddr": "{ipv6dstaddr}",
        "ipv6dstaddrprefix":
 "{ipv6dstaddrprefix}",
        "ipv6srcaddr": "{ipv6srcaddr}",
        "ipv6srcaddrprefix":
 "{ipv6srcaddrprefix}",
        "ipproto": "{ipproto}",
        "ipdscp": "{ipdscp}",
        "l4dstport": "{l4dstport}",
        "l4dstendport": "{l4dstendport}",
        "l4srcport": "{l4srcport}",
        "l4srcendport": "{l4srcendport}",
        "icmptypenum": "{icmptypenum}",
        "icmpcodenum": "{icmpcodenum}",
        "ipv6icmptypenum": "{ipv6icmptypenum}",
        "ipv6icmpcodenum": "{ipv6icmpcodenum}"
    }
}
```

Table 18: Description of response elements

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
macdstaddr	The MAC destination address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
macsrcaddr	The transmission source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.

Element	Description
macethertype	The Ether type of the Ethernet frame Valid value: A hexadecimal number. Valid range: 0x0000 - 0xffff
macvlanpriority	The VLAN priority number tag. Valid value: A decimal integer Valid range: 0 - 7
ipdstaddr	The destination IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
ipdstaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 32
ipsrcaddr	The transmission source IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
ipsrcaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 32
ipv6dstaddr	The destination IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2:288a:1fc0:0001:10ee
ipv6dstaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 128
ipv6srcaddr	The source IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2:288a:1fc0:0001:10ee
ipv6srcaddrprefix	The prefix length. Valid value: A decimal integer. Valid range: 1 - 128

Element	Description
ipproto	The IP protocol number. Valid value: A decimal integer. Valid range: 1 - 255
ipdscp	The DSCP value. Valid value: A decimal integer. Valid range: 0 - 63
l4dstport	The TCP or UDP destination port number. If a range is specified, this is viewed as start port. Valid value: A decimal integer. Valid range: 0 - 65535
l4dstendport	The end point TCP or UDP port number. Valid value: A decimal integer. Valid range: 1 - 65535
l4srcport	The TCP or UDP source port number. If a range is specified, this is viewed as l4SrcPortStart. Valid value: A decimal integer. Valid range: 0 - 65535
l4srcendport	The end point TCP or UDP port number. Valid value: A decimal integer. Valid range: 1 - 65535
icmptypenum	The ICMP type value. Valid value: A decimal integer. Valid range: 0 - 255
icmpcodenum	The ICMP code value. Valid value: A decimal integer. Valid range: 0 - 255
ipv6icmptypenum	The ICMP type value. Valid value: A decimal integer. Valid range: 0 - 255
ipv6icmpcodenum	The ICMP code value. Valid value: A decimal integer. Valid range: 0 - 255

1.5. VTN Station functions

This section lists the VTN Station functions.

1.5.1. Show VTN Stations

This operation is used to view a specific VTN Station information.

Processing request

Method	GET
Request URI	

- XML format

  /vtnstations.xml

  /vtnstations/detail.xml

  /vtnstations/count.xml

- JSON format

  /vtnstations.json

  /vtnstations/detail.json

  /vtnstations/count.json

- Query string

  ?controller_id={controller_id}&macaddr={macaddr}&ipaddr={ipaddr}

  &ipv6addr={ipv6addr}&switch_id={switch_id}&port_name={port_name}

  &vlan_id={vlan_id}&vtn_name={vtn_name}&domain_id={domain_id}

  &vbr_name={vbr_name}& if_name={if_name}

Table 19: Description of query string elements

Element	Description	Required
controller_id	The Controller identifier. Valid values: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *controller_id* is case sensitive.	Yes
macaddr	The MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)	No
ipv6addr	IPv6 address.	No

Element	Description	Required
	Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:) Example: 2001:0db8:bd05:01d2: 288a:1fc0:0001:10ee	
switch_id	Switch identifier. Valid value: A string of up to 255 characters.	No
port_name	Physical port name. Valid value: A string of up to 31 characters.	No
vlan_id	VLAN identifier. Valid value: A positive integer. Valid range: 1 - 4095	No
no_vlan_id	No VLAN ID. Valid value: Always true.	No
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.	No
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No

Request body   None

Processing result

Response body

- XML format

  If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtnstations>
    <vtnstation station_id="{station_id}"
 createdtime="{createdtime}"
 macaddr="{macaddr}" ipaddrs="{ipaddr}"
 ipv6addrs="{ipv6addr}" maptype="{maptype}"
 mapstatus="{mapstatus}"
 vtn_name="{vtn_name}" domain_id="{domain_id}"
 vbr_name="{vbr_name}" switch_id="{switch_id}"
 port_name="{port_name}" vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}" />
    <interface if_name="{if_name}"
 operstatus="{operstatus}" />
</vtnstations>
```

If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtnstations>
    <vtnstation station_id="{station_id}"
 createdtime="{createdtime}"
 macaddr="{macaddr}" ipaddrs="{ipaddr}"
 ipv6addrs="{ipv6addr}" maptype="{maptype}"
 mapstatus="{mapstatus}"
 vtn_name="{vtn_name}" domain_id="{domain_id}"
 vbr_name="{vbr_name}" switch_id="{switch_id}"
 port_name="{port_name}" vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}">
        <interface if_name="{if_name}"
 operstatus="{operstatus}" />
        <statistics>
            <openflow_controller>
                <all_rx packets="{packets}"
 octets="{octets}" />
                <all_tx packets="{packets}"
 octets="{octets}" />
            </openflow_controller>
            <openflow_nw>
                <all_rx packets="{packets}"
 octets="{octets}" />
                <all_tx packets="{packets}"
 octets="{octets}" />
                <existing_rx
 packets="{packets}" octets="{octets}" />
                <existing_tx
 packets="{packets}" octets="{octets}" />
                <expired_rx packets="{packets}"
 octets="{octets}" />
                <expired_tx packets="{packets}"
 octets="{octets}" />
                <all_drop_rx
 packets="{packets}" octets="{octets}" />
                <existing_drop_rx
 packets="{packets}" octets="{octets}" />
```

```
                    <expired_drop_rx
 packets="{packets}" octets="{octets}" />
                </openflow_nw>
            </statistics>
        </vtnstation>
</vtnstations>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtnstations count="{count}" />
```

- JSON format

  If count is not specified in URI

```
{
    "vtnstations": [
        {
            "station_id": "{station_id}",
            "createdtime": "{createdtime}",
            "macaddr": "{macaddr}",
            "ipaddrs": [
                "{ipaddr}"
            ],
            "ipv6addrs": [
                "{ipv6addr}"
            ],
            "maptype": "{maptype}",
            "mapstatus": "{mapstatus}",
            "vtn_name": "{vtn_name}",
            "domain_id": "{domain_id}",
            "vbr_name": "{vbr_name}",
            "interface": {
                "if_name": "{if_name}",
                "operstatus": "{operstatus}"
            },
            "switch_id": "{switch_id}",
            "port_name": "{port_name}",
            "vlan_id": "{vlan_id}",
            "no_vlan_id": "{no_vlan_id}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "vtnstations": [
        {
            "station_id": "{station_id}",
            "createdtime": "{createdtime}",
            "macaddr": "{macaddr}",
            "ipaddrs": [
                "{ipaddr}"
            ],
            "ipv6addrs": [
                "{ipv6addr}"
            ],
            "maptype": "{maptype}",
            "mapstatus": "{mapstatus}",
```

37

```
                "vtn_name": "{vtn_name}",
                "domain_id": "{domain_id}",
                "vbr_name": "{vbr_name}",
                "interface": {
                    "if_name": "{if_name}",
                    "operstatus": "{operstatus}"
                },
                "switch_id": "{switch_id}",
                "port_name": "{port_name}",
                "vlan_id": "{vlan_id}",
                "no_vlan_id": "{no_vlan_id}",
                "statistics": {
                    "openflow_controller": {
                        "all_rx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "all_tx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        }
                    },
                    "openflow_nw": {
                        "all_rx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "all_tx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "existing_rx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "existing_tx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "expired_rx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "expired_tx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "all_drop_rx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "existing_drop_rx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        },
                        "expired_drop_rx": {
                            "packets": "{packets}",
                            "octets": "{octets}"
                        }
                    }
                }
            }
```

```
        ]
}
```

If count is specified in URI

```
{
    "vtnstations": {
        "count": "{count}"
    }
}
```

Table 20: Description of response elements (VTN Stations)

Element	Description
station_id	VTN Station identifier. Valid value: A positive integer. Valid range: 1 - 524287
createdtime	Date and time when the VTN was created. Valid value: Date and time from 1970-01-01 00:00:00 to current date and time.
macaddr	MAC address of VTN Station. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
macaddr	The MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
ipaddr	IPv4 address of the VTN Station. Valid value: IPv4 dot-separated format (Example: 192.168.1.1) Note: If the VTN Station has multiple IPv4 addresses, they are displayed together as comma separated values.
ipv6addr	IPv6 address of the VTN Station. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). Example: 2001:0db8:bd05:01d2:288a:1fc0:0001:10ee

Element	Description
	Note: If the VTN Station has multiple IPv6 addresses, they are displayed together as comma separated values.
maptype	The vlan_id of port_name with datapath_id to which the VTN Station is connected and the type of mapping to VTN Interface. Valid value: ofs-map, vlan-map.
mapstatus	Map status. Valid value: valid, invalid.
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
interface	vBridge interface information.
switch_id	Switch identifier. Valid value: A string of up to 255 characters.
port_name	OFS port name. Valid value: A string of up to 31 characters.
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095
no_vlan_id	Indicates that no *vlan_id* is used. Valid value: true. Note: Either *vlan_id* or *no_vlan_id* can be specified.
statistics	Statistics information.

Element	Description
count	Number of displayed flows. Valid value: A positive integer.

Table 21: Description of response elements (interface)

Element	Description
if_name	vBridge interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
operstatus	Operational status. Valid value: up, down, unknown

Table 22: Description of response elements (statistics)

Element	Description
openflow_controller	The statistical information for the VTN Station in VTN Interface associated with the VTN Station.
openflow_nw	The statistical information for the VTN Station in vlan_id of port_name with datapath_id to which the VTN Station is connected.

**Table 23: Description of response
elements (openflow_controller)**

Element	Description
all_rx	The statistical information for all the frames received from the VTN Station.
all_tx	The statistical information for all the frames sent to the VTN Station.

Table 24: Description of response elements (openflow_nw)

Element	Description
all_rx	The statistical information for all the Ingress flow entries associated with the VTN Station. This is the sum of Existing RX and Expired RX. All Drop RX is included.
all_tx	The statistical information for all the Egress flow entries associated with the

Element	Description
	VTN Station. This is the sum of Existing RX and Expired RX.
existing_rx	The statistical information for all the currently existing Ingress flow entries associated with the VTN Station. Existing Drop RX is included.
existing_tx	The statistical information for all the currently existing Egress flow entries associated with the VTN Station.
expired_rx	The statistical information for all the already erased Ingress flow entries associated with the VTN Station. Expired Drop RX is included.
expired_tx	The statistical information for all the already erased Egress flow entries associated with the VTN Station.
all_drop_rx	The statistical information for the Ingress flow entries associated with the VTN Station whose action is drop. This is the sum of Existing Drop RX and Expired Drop RX.
existing_drop_rx	The statistical information for the currently existing Ingress flow entries associated with the VTN Station whose action is drop.
expired_drop_rx	The statistical information for the already erased Ingress flow entries associated with the VTN Station whose action is drop.

Table 25: Description of response elements (common)

Element	Description
packets	Number of frames. Valid value: A positive integer. Valid range: 0 - 18446744073709551615
octets	Number of octets in the frames. Valid value: A positive integer. Valid range: 0 - 18446744073709551615

1.6. VTN functions

This section lists the VTN functions.

1.6.1. Create VTN

This operation is used to create a VTN.

Processing request

Method   POST

Request URI
- XML format

  /vtns.xml
- JSON format

  /vtns.json

Request body
- XML format

```
<vtn vtn_name="{vtn_name}"
 description="{description}" />
```

- JSON format

```
{
    "vtn": {
        "vtn_name": "{vtn_name}",
        "description": "{description}"
    }
}
```

Table 26: Description of request elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	A small description text for the VTN. Valid value: A string of up to 127 characters.	No

Processing result

Response body   None

1.6.2. Delete VTN

This operation is used to delete a VTN.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}.xml

- JSON format

  /vtns/{vtn_name}.json

Table 27: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember

VTN that is set in *vtn_name* of vTunnel cannot be deleted.

Processing result

Response body   None

1.6.3. Update VTN

This operation is used to update a VTN.

Processing request

Method   PUT

Request URI
- XML format

  /vtns/{vtn_name}.xml
- JSON format

  /vtns/{vtn_name}.json

Table 28: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body
- XML format

```
<vtn description="{description}" />
```
- JSON format

```
{
    "vtn": {
        "description": "{description}"
    }
```

44

```
}
```

Table 29: Description of request elements

Element	Description	Required
description	VTN information. Valid value: A string of up to 127 characters.	No

Processing result

Response body   None

1.6.4. List VTNs

This operation is used to list VTN information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

  /vtns.xml

  /vtns/detail.xml

  /vtns/count.xml
- JSON format

  /vtns.json

  /vtns/detail.json

  /vtns/count.json
- Query string

  ?index={vtn_name}&max_repetition={max_repetition}

Table 30: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body   None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vtns>
      <vtn vtn_name="{vtn_name}" />
  </vtns>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vtns>
      <vtn vtn_name="{vtn_name}"
   description="{description}"
   operstatus="{operstatus}"
   createdtime="{createdtime}"
   lastcommittedtime="{lastcommittedtime}"
   alarmstatus="{alarmstatus}" />
  </vtns>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vtns count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "vtns": [
          {
              "vtn_name": "{vtn_name}"
          }
      ]
  }
  ```

  If detail is specified in URI

  ```
  {
      "vtns": [
          {
              "vtn_name": "{vtn_name}",
              "description": "{description}",
              "operstatus": "{operstatus}",
              "createdtime": "{createdtime}",
              "lastcommittedtime":
   "{lastcommittedtime}",
              "alarmstatus": "{alarmstatus}"
          }
      ]
  }
  ```

If count is specified in URI

```
{
    "vtns": {
        "count": "{count}"
    }
}
```

Table 31: Description of response elements

Element	Description
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	A small description for the VTN. Valid value: A string of up to 127 characters.
operstatus	Operational status. Valid value: up, down, unknown
createdtime	Date and time when the VTN was created. Valid value: Date and time from 1970-01-01 00:00:00 to current date and time.
lastcommittedtime	Last time the VTN related settings were updated. Date and time from 1970-01-01 00:00:00 to current date and time.
alarmstatus	Alarm status. Valid value: clear, raise
count	Number of VTN. Valid value: A positive integer.

1.6.5. Show VTN

This operation is used to view a specific VTN information.

Processing request

Method   GET

Request URI

- XML format

  /vtns/{vtn_name}.xml
- JSON format

  /vtns/{vtn_name}.json

Table 32: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtn vtn_name="{vtn_name}"
 description="{description}"
 operstatus="{operstatus}"
 createdtime="{createdtime}"
 lastcommittedtime="{lastcommittedtime}"
 alarmstatus="{alarmstatus}" />
```

- JSON format

```
{
    "vtn": {
        "vtn_name": "{vtn_name}",
        "description": "{description}",
        "operstatus": "{operstatus}",
        "createdtime": "{createdtime}",
        "lastcommittedtime":
 "{lastcommittedtime}",
        "alarmstatus": "{alarmstatus}"
    }
}
```

Table 33: Description of response elements

Element	Description
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
operstatus	Operational status. Valid value: up, down, unknown
createdtime	Date and time when the VTN was created.

Element	Description
	Valid value: Date and time from 1970-01-01 00:00:00 to current date and time.
lastcommittedtime	Last time the VTN related settings were updated. Date and time from 1970-01-01 00:00:00 to current date and time.
alarmstatus	Alarm status. Valid value: clear, raise

1.7. VTN Flow Filter functions

This section lists the VTN Flow Filter functions.

1.7.1. Create VTN Flow Filter

This operation is used to create a VTN Flow Filter.

Processing request

Method   POST

Request URI
- XML format

  /vtns/{vtn_name}/flowfilters.xml
- JSON format

  /vtns/{vtn_name}/flowfilters.json

Table 34: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No

Request body
- XML format

```
<flowfilter ff_type = "{ff_type}" />
```
- JSON format

```
{
    "flowfilter": {
        "ff_type": "{ff_type}"
    }
}
```

Table 35: Description of parameters

Element	Description	Required
ff_type	Input and output filters. Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Processing result

Response body   None

1.7.2. Delete VTN Flow Filter

This operation is used to delete a VTN Flow Filter.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/flowfilters/{ff_type}.xml
- JSON format

  /vtns/{vtn_name}/flowfilters/{ff_type}.json

Table 36: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
ff_type	Input and output filters. Valid value: in, out Note: The value of *ff_type* is case insensitive.	No

Request body   None

Processing result

Response body   None

1.7.3. Show VTN Flow Filter

This operation is used to view a specific VTN Flow Filter information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/flowfilters/{ff_type}.xml
• JSON format

/vtns/{vtn_name}/flowfilters/{ff_type}.json

Table 37: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
ff_type	Input and output filters. Valid value: in, out Note: The value of *ff_type* is case insensitive.	No

Request body   None

Processing result

Response body   • XML format

```
<flowfilter ff_type = "{ff_type}" />
```
• JSON format

```
{
    "flowfilter": {
        "ff_type": "{ff_type}"
    }
}
```

Table 38: Description of request elements

Element	Description	Required
ff_type	Input and output filters. Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

1.8. VTN Flow Filter Entry functions

This section lists the VTN Flow Filter Entry functions.

1.8.1. Create VTN Flow Filter Entry

This operation is used to create a VTN Flow Filter Entry.

Processing request

Method POST

Request URI
- XML format

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries.xml
- JSON format

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries.json

Table 39: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
ff_type	The input and output filters. Valid value: in, out	No

Request body
- XML format

```
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}" />
```

- JSON format

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}"
    }
}
```

Table 40: Description of request URI elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes
fl_name	Flow List name.	No

Element	Description	Required
	Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.	
action_type	Action type. Valid values: • pass: Passes the frame Note: The value of *action_type* is case insensitive.	No
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.	No
priority	The packet transfer priority order value. Valid value: A positive whole number. Valid range: 0 - 7	No
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63	No

Processing result

Response body   None

1.8.2. Delete VTN Flow Filter Entry

This operation is used to delete a VTN Flow Filter Entry.

Processing request

Method   DELETE

Request URI

• XML format

/vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.xml

• JSON format

/vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.json

Table 41: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	The input and output filters. Valid value: in, out	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Request body　　　None

Processing result

Response body　　　None

1.8.3. Update VTN Flow Filter Entry

This operation is used to update a VTN Flow Filter Entry.

Processing request

Method　　　PUT

Request URI
- XML format

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.xml
- JSON format

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.json

Table 42: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	The input and output filters. Valid value: in, out	No
seqnum	The sequence number. Valid value: A positive integer.	Yes

Element	Description	Required
	Valid range: 1 - 65535	

Request body

- XML format

```
<flowfilterentry fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}" />
```

- JSON format

```
{
    "flowfilterentry": {
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}"
    }
}
```

Table 43: Description of request elements

Element	Description
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action type. Valid values: • pass: Passes the frame Note: The value of *action_type* is case insensitive.
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.
priority	The packet transfer priority order value. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value.

Element	Description
	Valid value: A positive whole number. Valid range: 0 - 63

Processing result

Response body   None

1.8.4. List VTN Flow Filter Entries

This operation is used to list VTN Flow Filter Entry information.

Processing request

Method   GET

Request URI

- XML format

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries.xml

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/detail.xml

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/count.xml
- JSON format

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries.json

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/detail.json

  /vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/count.json
- Query string

  ?index={seqnum}&max_repetition={max_repetition}

Table 44: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	The input and output filters. Valid value: in, out	Yes

Table 45: Description of query string elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	No
max_repetition	The number of the resources that are returned.	No

Element	Description	Required
	Valid value: A positive integer.	
	Valid range: 1 to MAX of UINT32. Default is 10000.	

Request body None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```xml
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries>
      <flowfilterentry seqnum="{seqnum}" />
  </flowfilterentries>
  ```

  If detail is specified in URI

  ```xml
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries>
      <flowfilterentry
   seqnum="{seqnum}" fl_name="{fl_name}"
   action_type="{action_type}"
   nmg_name="{nmg_name}" priority="{priority}"
   dscp="{dscp}" />
  </flowfilterentries>
  ```

  If count is specified in URI

  ```xml
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries count="{count}"/>
  ```

- JSON format

  If count is not specified in URI

  ```json
  {
      "flowfilterentries": [
          {
              "seqnum": "{seqnum}"
          }
      ]
  }
  ```

  If detail is specified in URI

  ```json
  {
      "flowfilterentries": [
          {
              "seqnum": "{seqnum}",
              "fl_name": "{fl_name}",
              "action_type": "{action_type}",
              "nmg_name": "{nmg_name}",
  ```

57

```
                "priority": "{priority}",
                "dscp": "{dscp}"
            }
        ]
}
```

If count is specified in URI

```
{
    "flowfilterentries": {
        "count": "{count}"
    }
}
```

Table 46: Description of response elements

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action type. Valid values: • pass: Passes the frame
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
priority	The packet transfer priority order value. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63
count	The number of Flow Filter Entry. Valid value: A positive integer.

1.8.5. Show VTN Flow Filter Entry

This operation is used to view a specific VTN Flow Filter Entry information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.xml

/vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}/detail.xml

• JSON format

/vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.json

/vtns/{vtn_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}/detail.json

• Query string

?controller_id={controller_id}&domain_id={domain_id}

Table 47: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
ff_type	The input and output filters. Valid value: in, out	No
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	No

Table 48: Description of query string elements

Element	Description	Required
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT).	Yes

Element	Description	Required
	Note: The value of *domain_id* is case sensitive.	

Request body   None

Processing result

Response body
- XML format

  If detail is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}" />
```

  If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}" nmg_status="{nmg_status}">
    <statistics>
        <software packets="{packets}"
octets="{octets}" />
        <expiredflow packets="packets"
octets="{octets}" />
        <existingflow packets="{packets}"
octets="{octets}" />
        <total packets="{packets}"
octets="{octets}" />
    </statistics>
    <flowlist>
        <flowlistentries>
            <flowlistentry seqnum="{seqnum}">
                <statistics>
                    <software
packets="{packets}" octets="{octets}" />
                    <existingflow
packets="{packets}" octets="{octets}" />
                    <expiredflow
packets="{packets}" octets="{octets}" />
                    <total packets="{packets}"
octets="{octets}" />
                </statistics>
            </flowlistentry>
        </flowlistentries>
    </flowlist>
</flowfilterentry>
```

- JSON format

If detail is not specified in URI

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}"
    }
}
```

If detail is specified in URI

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name ": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "nmg_status": "{nmg_status}",
        "statistics": {
            "software": {
                "packets": "{packets}",
                "octets": "{octets}"
            },
            "existingflow": {
                "packets": "{packets}",
                "octets": "{octets}"
            },
            "expiredflow": {
                "packets": "{packets}",
                "octets": "{octets}"
            },
            "total": {
                "packets": "{packets}",
                "octets": "{octets}"
            }
        },
        "flowlist": {
            "flowlistentries": [
                {
                    "seqnum": "{seqnum}",
                    "statistics": {
                        "software": {
                            "packets":
 "{packets}",
                            "octets":
 "{octets}"
                        },
                        "existingflow": {
                            "packets":
 "{packets}",
                            "octets":
 "{octets}"
                        },
                        "expiredflow": {
```

```
                                            "packets":
"{packets}",
                                            "octets":
"{octets}"
                                    },
                                    "total": {
                                            "packets":
"{packets}",
                                            "octets":
"{octets}"
                                    }
                            }
                        }
                    ]
                }
            }
}
```

Table 49: Description of response elements (flowfilterentry)

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action type. Valid values: • pass: Passes the frame Note: The value of *action_type* is case insensitive.
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.
priority	The packet transfer priority order value. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value. Valid value: A positive whole number.

Element	Description
	Valid range: 0 - 63
nmg_status	Status of monitored host. Valid value: 0 - 256. Value 1 is faulty state and other than 1 are not defined.
statistics	Statistical information.

Table 50: Description of response elements (statistics)

Element	Description
software	The number of packets or bytes that passed through the VTN among the flows that match the flow filter entry.
existingflow	The number of packets or bytes that are hard-transferred by the flow entry currently set to the OFS among the flows that match the flow filter entry.
expiredflow	The number of packets or bytes that are hard-transferred by the flow entry previously set to the OFS among the flows that match the flow filter entry.
total	Total number of packets or bytes of the flow that matches the flow filter entry. This is the sum of Software, and ExistingFlow, and ExpiredFlow.

Table 51: Description of response elements (flowlistentry)

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
statistics	Statistics information.

Table 52: Description of response elements (statistics)

Element	Description
packets	Number of frames. Valid value: A positive integer. Valid range: 0 - 18446744073709551615
octets	Number of octets in the frames. Valid value: A positive integer.

Element	Description
	Valid range: 0 - 18446744073709551615

1.9. vBridge functions

This section lists the vBridge functions.

1.9.1. Create vBridge

This operation is used to create a vBridge.

Processing request

Method   POST

Request URI
- XML format

  /vtns/{vtn_name}/vbridges.xml
- JSON format

  /vtns/{vtn_name}/vbridges.json

Table 53: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body
- XML format

```
<vbridge vbr_name="{vbr_name}"
 controller_id="{controller_id}"
 description="{description}"
 domain_id="{domain_id}" />
```

- JSON format

```
{
    "vbridge": {
        "vbr_name": "{vbr_name}",
        "controller_id": "{controller_id}",
        "description": "{description}",
        "domain_id": "{domain_id}"
    }
}
```

Table 54: Description of request elements

Element	Description	Required
vbr_name	vBridge name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.	Yes

Processing result

Response body   None

1.9.2. Delete vBridge

This operation is used to delete a vBridge.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}.json

Table 55: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	

Request body None

Remember

vBridge whose Interface is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Processing result

Response body None

1.9.3. Update vBridge

This operation is used to update a vBridge.

Processing request

Method PUT

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}.json

Table 56: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body
- XML format

```
<vbridge controller_id="{controller_id}"
  description="{description}"
  domain_id="{domain_id}" />
```
- JSON format

```
{
    "vbridge": {
```

```
            "controller_id": "{controller_id}",
            "description": "{description}",
            "domain_id": "{domain_id}"
        }
    }
```

Table 57: Description of request elements

Element	Description	Required
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
description	Additional information. Valid value: A string of up to 127 characters.	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.	No

Processing result

Response body   None

1.9.4. List vBridges

This operation is used to list vBridge information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

    /vtns/{vtn_name}/vbridges.xml

    /vtns/{vtn_name}/vbridges/detail.xml

    /vtns/{vtn_name}/vbridges/count.xml
- JSON format

    /vtns/{vtn_name}/vbridges.json

    /vtns/{vtn_name}/vbridges/detail.json

    /vtns/{vtn_name}/vbridges/count.json
- Query string

    ?index={vbr_name}&max_repetition={max_repetition}

Table 58: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 59: Description of query string elements

Element	Description	Required
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

  If detail/count are not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vbridges>
      <vbridge vbr_name="{vbr_name}" />
  </vbridges>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vbridges>
      <vbridge vbr_name="{vbr_name}"
   controller_id="{controller_id}"
   description="{description}"
   domain_id="{domain_id}" status="{status}"></
  vbridge>
  </vbridges>
  ```

68

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vbridges count="{count}" />
```

• JSON format

If count is not specified in URI

```
{
    "vbridges": [
        {
            "vbr_name": "{vbr_name}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "vbridges": [
        {
            "vbr_name": "{vbr_name}",
            "controller_id": "{controller_id}",
            "description": "{description}",
            "status": "{status}",
            "domain_id": "{domain_id}"
        }
    ]
}
```

If count is specified in URI

```
{
    "vbridges": {
        "count": "{count}"
    }
}
```

Table 60: Description of response elements

Element	Description
count	The number of vBridge.
	Valid value: A positive integer.
vbr_name	vBridge name.
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
controller_id	Controller identifier.
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

69

Element	Description
description	Additional information. Valid value: A string of up to 127 characters.
status	vBridge status. Valid value: up, down, unknown
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.

1.9.5. Show vBridge

This operation is used to view a specific vBridge information.

Processing request

Method GET

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}.json

Table 61: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vbridge vbr_name="{vbr_name}"
 controller_id="{controller_id}"
 description="{description}" status="{status}"
 domain_id="{domain_id}"></vbridge>
```

- JSON format

```
{
    "vbridge": {
        "vbr_name": "{vbr_name}",
        "controller_id": "{controller_id}",
        "description": "{description}",
        "status": "{status}",
        "domain_id": "{domain_id}"
    }
}
```

Table 62: Description of response elements

Element	Description
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
status	vBridge status. Valid value: up, down, unknown
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.

1.10. Host Address functions

This section lists the Host Address functions.

1.10.1. Delete Host Address

This operation is used to delete a Host Address.

Processing request

Method	DELETE
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/ipaddress.xml
* JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/ipaddress.json

Table 63: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body   None

1.10.2. Set Host Address

This operation is used to set a Host Address.

Processing request

Method	PUT
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/ipaddress.xml
* JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/ipaddress.json

Table 64: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<ipaddress ipaddr="{ipaddr}"
 prefix="{prefix}" />
```

- JSON format

```
{
    "ipaddress": {
        "ipaddr": "{ipaddr}",
        "prefix": "{prefix}"
    }
}
```

Table 65: Description of request elements

Element	Description	Required
ipaddr	IP address. Valid value: IPv4 dot-separated format Example: 192.168.1.1 Note: The value of this parameter must be unique within VTN.	Yes
prefix	Prefix length. Valid value: A positive integer between 1 and 30.	Yes

Processing result

Response body   None

1.10.3. Show Host Address

This operation is used to view a specific Host Address information.

Processing request

Method	GET

Request URI

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/ipaddress.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/ipaddress.json

Table 66: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body	None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<ipaddress ipaddr="ipaddr" prefix="prefix" />
```

- JSON format

```
{
    "ipaddress":{
        "ipaddr": "ipaddr",
        "prefix": "prefix"
    }
}
```

Table 67: Description of response elements

Element	Description
ipaddr	IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
prefix	Prefix length.

74

Element	Description
	Valid value: A positive integer between 1 and 30.

1.11. L2 Domain function

This section describes the L2 Domain function.

1.11.1. Show L2 Domains

This operation is used to view a specific L2 Domain information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/l2domains.xml

/vtns/{vtn_name}/vbridges/{vbr_name}/l2domains/count.xml

• JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/l2domains.json

/vtns/{vtn_name}/vbridges/{vbr_name}/l2domains/count.json

Table 68: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body None

Processing result

Response body • XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<l2domains>
    <l2domain l2domain_id="{l2domain_id}">
        <l2domain_members>
```

```
            <l2domain_member
 switch_id="{switch_id}" vlan_id="{vlan_id}" />
        </l2domain_members>
    </l2domain>
</l2domains>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<l2domains count="{count}" />
```

- JSON format

    If count is not specified in URI

```
{
    "l2domains": [
        {
            "l2domain_id": "{l2domain_id}",
            "l2domain_members": [
                {
                    "switch_id": "{switch_id}",
                    "vlan_id": "{vlan_id}"
                }
            ]
        }
    ]
}
```

    If count is specified in URI

```
{
    "l2domains": {
        "count": "{count}"
    }
}
```

Table 69: Description of response elements

Element	Description
l2domain_id	L2 domain identifier. Valid value: A positive integer of length 7.
l2domain_members	L2 domain member.
count	Number of L2 domain in system. Valid value: A positive integer of length 5.
switch_id	Switch identifier. Valid value: A string of up to 255 characters.
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer.

Element	Description
	Valid range: 1 - 4095

1.12. MAC Entry function

This section describes the MAC Entry function.

1.12.1. Show MAC Entries

This operation is used to view a specific MAC Entry information.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/macentries.xml

  /vtns/{vtn_name}/vbridges/{vbr_name}/macentries/count.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/macentries.json

  /vtns/{vtn_name}/vbridges/{vbr_name}/macentries/count.json

Table 70: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body
- XML format

  If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<macentries>
[<macentry macaddr="{macaddr}"
type="{type}"
port_name="{port_name}"
```

```
if_kind="{if_kind}"/>]
</macentries>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<macentries count="{count}"/>
```

- JSON format

  If count is not specified in URI

```
{
    "macentries": [
        {
            "macaddr": "{macaddr}",
            "type": "{type}",
            "port_name": "{port_name}",
            "if_kind": "{if_kind}"
        }
    ]
}
```

  If count is specified in URI

```
{
    "macentries": {
        "count": "{count}"
    }
}
```

Table 71: Description of response elements

Element	Description
macaddr	The MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
type	Type of MAC address. Valid value: static, dynamic
port_name	The interface name or trunk interface name. Valid value: A string of up to 31 characters.
if_kind	If 0 from UPLL then blank string and if 1 from UPLL then trunk needs to be set. Valid value: trunk, a blank string.
count	Total number of MAC entries. Valid value: A positive integer.

Element	Description
	Valid range: 1 - 65535

1.13. VLAN Map functions

This section lists the VLAN Map functions.

1.13.1. Create VLAN Map

This operation is used to create a VLAN Map.

Processing request

Method	POST
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps.xml

• JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps.json

Table 72: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

• XML format

```
<vlanmap logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}" />
```

• JSON format

```
{
    "vlanmap": {
        "logical_port_id": "{logical_port_id}",
        "vlan_id": "{vlan_id}",
        "no_vlan_id": "{no_vlan_id}"
    }
}
```

Table 73: Description of request elements

Element	Description	Required
logical_port_id	VTN name. Valid value: A string of up to 319 characters.	No
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095	No
no_vlan_id	Indicates that no *vlan_id* is used. Valid value: true. Note: Either *vlan_id* or *no_vlan_id* can be specified.	No

Remember

Combination of *logical_port_id* and *vlan_id* must be unique within vBridge.

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vlanmap vlanmap_id="{vlanmap_id}" />
```

- JSON format

```
{
    "vlanmap": {
        "vlanmap_id": "{vlanmap_id}"
    }
}
```

Table 74: Description of response elements

Element	Description
vlanmap_id	VLAN Map identifier. If logical_port_id is specified at creation time, *vlanmap_id* is "lpid-{logical_port_id}". Otherwise, it is "no_lpid ".

1.13.2. Delete VLAN Map

This operation is used to delete a VLAN Map.

Processing request

Method   DELETE

Request URI

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/{vlanmap_id}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/{vlanmap_id}.json

Table 75: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vlanmap_id	VLAN Map identifier. If *logical_port_id* is specified at creation time, *vlanmap_id* is "lpid-{logical_port_id}". Otherwise, it is "no_lpid".	Yes

Request body   None

Processing result

Response body   None

1.13.3. Update VLAN Map

This operation is used to update a VLAN Map.

Processing request

Method   PUT

Request URI

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/{vlanmap_id}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/{vlanmap_id}.json

Table 76: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vlanmap_id	VLAN Map identifier. If logical_port_id is specified at creation time, *vlanmap_id* is "lpid-{logical_port_id}". Otherwise, it is "no_lpid".	Yes

Request body

- XML format

```
<vlanmap vlan_id ="{vlan_id}" no_vlan_id
 ="{no_vlan_id}" />
```

- JSON format

```
{
    "vlanmap": {
        "vlan_id": "{vlan_id}",
        "no_vlan_id": "{no_vlan_id}"
    }
}
```

Table 77: Description of request elements

Element	Description	Required
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095	No
no_vlan_id	Indicates that no *vlan_id* is used. Valid value: true. Note: Either *vlan_id* or *no_vlan_id* can be specified.	No

Remember

Combination of *logical_port_id* and *vlan_id* must be unique within vBridge.

Processing result

Response body   None

1.13.4. List VLAN Maps

This operation is used to list VLAN Map information based on specified conditions.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps.xml

/vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/detail.xml

/vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/count.xml

• JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps.json

/vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/detail.json

/vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/count.json

• Query string

?index={vlanmap_id}&max_repetition={max_repetition}

Table 78: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 79: Description of query string elements

Element	Description	Required
vlanmap_id	VLAN Map identifier. If *logical_port_id* is specified at creation time, *vlanmap_id* is "lpid-{logical_port_id}". Otherwise, *vlanmap_id* is "no_lpid".	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body   None

Processing result

Response body   • XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vlanmaps>
    <vlanmap vlanmap_id="{vlanmap_id}" />
</vlanmaps>
```

If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vlanmaps>
    <vlanmap vlanmap_id="{vlanmap_id}"
 logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}" />
</vlanmaps>
```

If count is specified in URI

```
<vlanmaps count="{count}"/>
```

• JSON format

If count is not specified in URI

```
{
    "vlanmaps": [
        {
            "vlanmap_id": "{vlanmap_id}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "vlanmaps": [
        {
            "vlanmap_id": "{vlanmap_id}",
            "logical_port_id":
 "{logical_port_id}",
            "vlan_id": "{vlan_id}",
            "no_vlan_id": "{no_vlan_id}"
        }
    ]
}
```

If count is specified in URI

```
{
    "vlanmaps": {
        "count": "{count}"
```

84

```
      }
    }
```

Table 80: Description of response elements

Element	Description
vlanmap_id	VLAN Map identifier. If *logical_port_id* is specified at creation time, *vlanmap_id* is "lpid-{logical_port_id}". Otherwise, *vlanmap_id* is "no_lpid".
logical_port_id	Logical port identifier. Valid value: A string of up to 319 characters.
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095
no_vlan_id	Indicates that no *vlan_id* is used. Valid value: true. Note: Either *vlan_id* or *no_vlan_id* can be specified.
count	Number of VLAN Maps. Valid value: A positive integer.

1.13.5. Show VLAN Map

This operation is used to view a specific VLAN Map information.

Processing request

Method	GET
Request URI	• XML format /vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/{vlanmap_id}.xml • JSON format /vtns/{vtn_name}/vbridges/{vbr_name}/vlanmaps/{vlanmap_id}.json

Table 81: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Element	Description	Required
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vlanmap_id	VLAN Map identifier. If *logical_port_id* is specified at creation time, *vlanmap_id* is "lpid-{logical_port_id}". Otherwise, it is "no_lpid".	Yes

Request body

None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vlanmap vlanmap_id="{vlanmap_id}"
 logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}" />
```

- JSON format

```
{
    "vlanmap": {
        "vlanmap_id": "{vlanmap_id}",
        "logical_port_id": "{logical_port_id}",
        "vlan_id": "{vlan_id}",
        "no_vlan_id": "{no_vlan_id}"
    }
}
```

Table 82: Description of response elements

Element	Description
vlanmap_id	VLAN Map identifier. If *logical_port_id* is specified at creation time, *vlanmap_id* is "lpid-{logical_port_id}". Otherwise, *vlanmap_id* is "no_lpid".
logical_port_id	Logical port identifier. Valid value: A string of up to 319 characters.
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer.

86

Element	Description
	Valid range: 1 - 4095
no_vlan_id	Indicates that no *vlan_id* is used. Valid value: true. Note: Either *vlan_id* or *no_vlan_id* can be specified.

1.14. vBridge Flow Filter functions

This section lists the vBridge Flow Filter functions.

1.14.1. Create vBridge Flow Filter

This operation is used to create a vBridge Flow Filter.

Processing request

Method POST

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters.json

Table 83: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body
- XML format

  ```
  <flowfilter ff_type="{ff_type}" />
  ```
- JSON format

  ```
  {
      "flowfilter": {
          "ff_type": "{ff_type}"
      }
  }
  ```

87

Table 84: Description of request elements

Element	Description	Required
ff_type	Direction to which the Flow Filter is applied Valid value: in Note: The value of *ff_type* is case insensitive.	Yes

Processing result

Response body   None

1.14.2. Delete vBridge Flow Filter

This operation is used to delete a vBridge Flow Filter.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}.json

Table 85: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in Note: The value of *ff_type* is case insensitive.	Yes

Request body   None

Processing result

Response body   None

1.14.3. Show vBridge Flow Filter

This operation is used to view a specific vBridge Flow Filter information.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}.json

Table 86: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in Note: The value of *ff_type* is case insensitive.	Yes

Request body   None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilter ff_type="{ff_type}" />
```
- JSON format

```
{
    "flowfilter": {
        "ff_type": "{ff_type}"
    }
```

```
}
```

Table 87: Description of response elements

Element	Description
ff_type	Direction to which the Flow Filter is applied Valid value: in Note: The value of *ff_type* is case insensitive.

1.15. vBridge Flow Filter Entry functions

This section lists the vBridge Flow Filter Entry functions.

1.15.1. Create vBridge Flow Filter Entry

This operation is used to create a vBridge Flow Filter Entry.

Processing request

Method	POST
Request URI	• XML format /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/ flowfilterentries.xml • JSON format /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/ flowfilterentries.json

Table 88: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in	Yes

Element	Description	Required
	Note: The value of *ff_type* is case insensitive.	

Request body

- XML format

```
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
    <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}" />
</flowfilterentry>
```

- JSON format

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        }
    }
}
```

Table 89: Description of request elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.	No
action_type	Action that is registered in the Flow Filter entry. Valid values:	No

91

Element	Description	Required
	• pass: Passes the frame. • drop: Discards the frame. • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified. Note: The value of *action_type* is case insensitive.	
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.	No
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7	No
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63	No
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No
macsrcaddr	Source MAC address.	No

92

Element	Description	Required
	Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	

Processing result

Response body   None

1.15.2. Delete vBridge Flow Filter Entry

This operation is used to delete a vBridge Flow Filter Entry.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/
  flowfilterentries/{seqnum}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/
  flowfilterentries/{seqnum}.json

Table 90: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in Note: The value of *ff_type* is case insensitive.	Yes
seqnum	The sequence number. Valid value: A positive integer.	Yes

Element	Description	Required
	Valid range: 1 - 65535	

Request body   None

Processing result

Response body   None

1.15.3. Update vBridge Flow Filter Entry

This operation is used to update a vBridge Flow Filter Entry.

Processing request

Method   PUT

Request URI

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/ flowfilterentries/{seqnum}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/ flowfilterentries/{seqnum}.json

Table 91: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in Note: The value of *ff_type* is case insensitive.	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Request body

- XML format

```
<flowfilterentry fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
     <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}" />
</flowfilterentry>
```

- JSON format

```
{
    "flowfilterentry": {
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        }
    }
}
```

Table 92: Description of request elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.	No
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified. Note: The value of *action_type* is case insensitive.	No
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers,	No

Element	Description	Required
	and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.	
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7	No
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63	No
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No

Processing result

Response body   None

1.15.4. List vBridge Flow Filter Entries

This operation is used to list vBridge Flow Filter Entry information.

Processing request

Method	GET
Request URI	

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/flowfilterentries.xml

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/flowfilterentries/detail.xml

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/flowfilterentries/count.xml

- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/flowfilterentries.json

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/flowfilterentries/detail.json

  /vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/flowfilterentries/count.json

- Query string

  ?index={seqnum}&max_repetition={max_repetition}

Table 93: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in Note: The value of *ff_type* is case insensitive.	Yes

Table 94: Description of query string elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer.	No

97

Element	Description	Required
	Valid range: 1 - 65535	
max_repetition	The number of the resources that are returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries>
      <flowfilterentry seqnum="{seqnum}" />
  </flowfilterentries>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries>
      <flowfilterentry
   seqnum="{seqnum}" fl_name="{fl_name}"
   action_type="{action_type}"
   nmg_name="{nmg_name}" priority="{priority}"
   dscp="{dscp}">
          <redirectdst vnode_name="{vnode_name}"
   if_name="{if_name}" macdstaddr="{macdstaddr}"
   macsrcaddr="{macsrcaddr}" />
      </flowfilterentry>
  </flowfilterentries>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentry count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "flowfilterentries": [
          {
              "seqnum": "{seqnum}"
          }
      ]
  }
  ```

98

If detail is specified in URI

```
{
    "flowfilterentries": [
        {
            "seqnum": "{seqnum}",
            "fl_name": "{fl_name}",
            "action_type": "{action_type}",
            "nmg_name": "{nmg_name}",
            "priority": "{priority}",
            "dscp": "{dscp}",
            "redirectdst": {
                "vnode_name": "{vnode_name}",
                "if_name": "{if_name}",
                "macdstaddr": "{macdstaddr}",
                "macsrcaddr": "{macsrcaddr}"
            }
        }
    ]
}
```

If count is specified in URI

```
{
    "flowfilterentries": {
        "count": "{count}"
    }
}
```

Table 95: Description of response elements

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified.
nmg_name	Network monitor group name.

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63
count	The number of Flow Filter Entry. Valid value: A positive integer.
redirectdst	Redirect information.

Table 96: Description of response elements (redirectdst)

Element	Description
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.

1.15.5. Show vBridge Flow Filter Entry

This operation is used to view a specific vBridge Flow Filter Entry information.

Processing request

Method	GET

Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/
flowfilterentries/{seqnum}.xml

/vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/
flowfilterentries/{seqnum}/detail.xml

• JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/
flowfilterentries/{seqnum}.json

/vtns/{vtn_name}/vbridges/{vbr_name}/flowfilters/{ff_type}/
flowfilterentries/{seqnum}/detail.json

• Query string

?controller_id={controller_id}

Table 97: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in Note: The value of *ff_type* is case insensitive.	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Table 98: Description of query string elements

Element	Description	Required
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	

Request body None

Processing result

Response body
- XML format

  If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
    <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}" />
</flowfilterentry>
```

  If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}" nmg_status="{nmg_status}">
    <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macsrcaddr}"
 macsrcaddr="{macsrcaddr}" />
    <statistics>
        <software packets="{packets}"
 octets="{octets}" />
        <existingflow packets="{packets}"
 octets="{octets}" />
        <expiredflow packets="{packets}"
 octets="{octets}" />
        <total packets="{packets}"
 octets="{octets}" />
    </statistics>
    <flowlist>
        <flowlistentries>
            <flowlistentry seqnum="{seqnum}">
                <statistics>
                    <software
 packets="{packets}" octets="{octets}" />
                    <existingflow
 packets="{packets}" octets="{octets}" />
                    <expiredflow
 packets="{packets}" octets="{octets}" />
                    <total packets="{packets}"
 octets="{octets}" />
                </statistics>
```

102

```
            </flowlistentry>
          </flowlistentries>
      </flowlist>
</flowfilterentry>
```

- JSON format

  If count is not specified in URI

```
{
     "flowfilterentry": {
          "seqnum": "{seqnum}",
          "fl_name": "{fl_name}",
          "action_type": "{action_type}",
          "nmg_name": "{nmg_name}",
          "priority": "{priority}",
          "dscp": "{dscp}",
          "redirectdst": {
               "vnode_name": "{vnode_name}",
               "if_name": "{if_name}",
               "macdstaddr": "{macdstaddr}",
               "macsrcaddr": "{macsrcaddr}"
          }
     }
}
```

  If detail is specified in URI

```
{
     "flowfilterentry": {
          "seqnum": "{seqnum}",
          "fl_name": "{fl_name}",
          "action_type": "{action_type}",
          "nmg_name ": "{nmg_name}",
          "priority": "{priority}",
          "dscp": "{dscp}",
          "nmg_status": "{nmg_status}",
          "redirectdst": {
               "vnode_name": "{vnode_name}",
               "if_name": "{if_name}",
               "macdstaddr": "{macdstaddr}",
               "macsrcaddr": "{macsrcaddr}"
          },
          "statistics": {
               "software": {
                    "packets": "{packets}",
                    "octets": "{octets}"
               },
               "existingflow": {
                    "packets": "{packets}",
                    "octets": "{octets}"
               },
               "expiredflow": {
                    "packets": "{packets}",
                    "octets": "{octets}"
               },
               "total": {
                    "packets": "{packets}",
                    "octets": "{octets}"
               }
          },
          "flowlist": {
```

103

```
            "flowlistentries": [
                {
                    "seqnum": "{seqnum}",
                    "statistics": {
                        "software": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "existingflow": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "expiredflow": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "total": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        }
                    }
                }
            ]
        }
    }
}
```

Table 99: Description of response elements (flowfilterentry)

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame

Element	Description
	• redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified.
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.
priority	Priority value registered to the flow filter entry. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63
nmg_status	Status of monitored host. Valid value: 0 - 256. Value "1" is faulty state and other than "1" are not defined.
redirectdst	Redirect information.
statistics	Statistical information.
flowlist	Flow List information.

Table 100: Description of response elements (redirectdst)

Element	Description
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab).

Element	Description
	Note: MAC address is case insensitive.
macsrcaddr	Source MAC address.
	Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab).
	Note: MAC address is case insensitive.

Table 101: Description of response elements (statistics)

Element	Description
software	The number of packets or bytes that passed through the VTN among the flows that match the flow filter entry.
existingflow	The number of packets or bytes that are hard-transferred by the flow entry currently set to the OFS among the flows that match the flow filter entry.
expiredflow	The number of packets or bytes that are hard-transferred by the flow entry previously set to the OFS among the flows that match the flow filter entry.
total	Total number of packets or bytes of the flow that matches the flow filter entry. This is the sum of Software, and ExistingFlow, and ExpiredFlow.

Table 102: Description of response elements (flowlist)

Element	Description
flowlistentries	Flow List entry list.

Table 103: Description of response elements (flowlistentry)

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
statistics	Statistics information.

Table 104: Description of response elements (common)

Element	Description
packets	Number of frames. Valid value: A positive integer.

Element	Description
	Valid range: 0 - 18446744073709551615
octets	Number of octets in the frames. Valid value: A positive integer. Valid range: 0 - 18446744073709551615

1.16. vBridge Interface functions

This section lists the vBridge Interface functions.

1.16.1. Create vBridge Interface

This operation is used to create a vBridge Interface.

Processing request

Method   POST

Request URI

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces.json

Table 105: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}" />
```

- JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}"
```

107

```
        }
    }
```

Table 106: Description of request elements

Element	Description	Required
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.16.2. Delete vBridge Interface

This operation is used to delete a vBridge Interface.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}.json

Table 107: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Element	Description	Required
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember
vBridge Interface that is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Processing result

Response body   None

1.16.3. List vBridge Interfaces

This operation is used to list vBridge Interface information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces.xml

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/detail.xml

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/count.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces.json

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/detail.json

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/count.json
- Query string

  ?index={if_name}&max_repetition={max_repetition}

Table 108: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

109

Table 109: Description of query string elements

Element	Description	Required
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <interfaces>
      <interface if_name="{if_name}" />
  </interfaces>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <interfaces>
      <interface if_name="{if_name}"
   description="{description}"
   adminstatus="{adminstatus}"
   operstatus="{operstatus}">
          <neighbor vnode_name="{vnode_name}"
   if_name="{if_name}" vlk_name="{vlk_name}" />
      </interface>
  </interfaces>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <interfaces count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "interfaces": [
          {
  ```

110

```
                    "if_name": "{if_name}"
            }
        ]
}
```

If detail is specified in URI

```
{
    "interfaces": [
        {
            "if_name": "{if_name}",
            "description": "{description}",
            "adminstatus": "{adminstatus}",
            "operstatus": "{operstatus}",
            "neighbor": {
                "vnode_name": "{vnode_name}",
                "if_name": "{if_name}",
                "vlk_name": "{vlk_name}"
            }
        }
    ]
}
```

If count is specified in URI

```
{
    "interfaces": {
        "count": "{count}"
    }
}
```

Table 110: Description of response elements (Interface)

Element	Description
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
operstatus	Operational status. Valid value: up, down, unknown
neighbor	Information about the neighbor.
count	The number of vBridge Interface. Valid value: A positive integer.

Table 111: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: A positive integer.
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.16.4. Show vBridge Interface

This operation is used to view a specific vBridge Interface information.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}.json

Table 112: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}"
 operstatus="{operstatus}">
    <neighbor vnode_name="{vnode_name}"
 if_name="{if_name}" vlk_name="{vlk_name}" />
</interface>
```

- JSON format

```
{
    "interfaces": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "operstatus": "{operstatus}",
        "neighbor": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "vlk_name": "{vlk_name}"
        }
    }
}
```

Table 113: Description of response elements (Interface)

Element	Description
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: Up to 127 characters that can include alphabets, numbers, and underscore (except at the beginning).
adminstatus	Admin status. Valid value: enable, disable.
operstatus	Operational status. Valid value: up, down, unknown
neighbor	Information about the neighbor.

113

Table 114: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: A positive integer.
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.16.5. Update vBridge Interface

This operation is used to update a vBridge Interface.

Processing request

Method   PUT

Request URI

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}.json

Table 115: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<interface description="{description}"
 adminstatus="{adminstatus}" />
```

- JSON format

```
{
    "interface": {
        "description": "{description}",
        "adminstatus": "{adminstatus}"
    }
}
```

Table 116: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.17. vBridge Interface Port Map functions

This section lists the vBridge Interface Port Map functions.

1.17.1. Delete vBridge Interface Port Map

This operation is used to delete a vBridge Interface Port Map.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/
  portmap.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/
  portmap.json

Table 117: Description of request URI elements

Element	Description	Required
vtn_name	VTN name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body   None

1.17.2. Set vBridge Interface Port Map

This operation is used to update a vBridge Interface Port Map for specific settings.

Processing request

Method   PUT

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/portmap.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/portmap.json

Table 118: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<portmap logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}" tagged="{tagged}" />
```

- JSON format

```
{
    "portmap": {
        "logical_port_id": "{logical_port_id}",
        "vlan_id": "{vlan_id}",
        "tagged": "{tagged}"
    }
}
```

Table 119: Description of request elements

Element	Description	Required
logical_port_id	Logical port identifier. Valid value: A string of up to 319 characters.	Yes
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095 Note: This parameter is required if *tagged* is specified.	No
tagged	Displays whether VLAN tags are sent and received in the physical network. Valid value: - true: Send and receive packets with VLANtags. - false: Send and receive packets without VLAN tags.	No

Processing result

Response body   None

117

1.17.3. Show vBridge Interface Port Map

This operation is used to view a specific vBridge Interface Port Map information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/
portmap.xml

• JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/
portmap.json

Table 120: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body	None

Processing result

Response body	• XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<portmap logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}" tagged="{tagged}" />
```

• JSON format

```
{
    "portmap": {
        "logical_port_id": "{logical_port_id}",
        "vlan_id": "{vlan_id}",
        "tagged": "{tagged}"
    }
```

```
}
```

Table 121: Description of response elements

Element	Description
logical_port_id	Logical port identifier. Valid value: A string of up to 319 characters.
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095
tagged	Displays whether VLAN tags are sent and received in the physical network . Valid value: • true: Send and receive packets with VLANtags. • false: Send and receive packets without VLAN tags.

1.18. vBridge Interface Flow Filter functions

This section lists the vBridge Interface Flow Filter functions.

1.18.1. Create vBridge Interface Flow Filter

This operation is used to create a vBridge Interface Flow Filter.

Processing request

Method	POST
Request URI	• XML format /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters.xml • JSON format /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters.json

Table 122: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<flowfilter ff_type="{ff_type}" />
```

- JSON format

```
{
    "flowfilter": {
        "ff_type": "{ff_type}"
    }
}
```

Table 123: Description of request elements

Element	Description	Required
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Processing result

Response body   None

1.18.2. Delete vBridge Interface Flow Filter

This operation is used to delete a vBridge Interface Flow Filter.

Processing request

Method   DELETE

Request URI

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}.xml

- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}.json

Table 124: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Request body   None

Processing result

Response body   None

1.18.3. Show vBridge Interface Flow Filter

This operation is used to view a specific vBridge Interface Flow Filter information.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}.xml
- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}.json

Table 125: Description of request URI elements

Element	Description	Required
vtn_name	VTN name.	Yes

121

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Request body None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilter ff_type="{ff_type}" />
```

- JSON format

```
{
    "flowfilter": {
        "ff_type": "{ff_type}"
    }
}
```

Table 126: Description of response elements

Element	Description
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.

1.19. vBridge Interface Flow Filter Entry functions

This section lists the vBridge Interface Flow Filter Entry functions.

1.19.1. Create vBridge Interface Flow Filter Entry

This operation is used to create a vBridge Interface Flow Filter Entry.

Processing request

Method	POST
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries.xml
• JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries.json

Table 127: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Request body
• XML format

```
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
```

```
      <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}" />
</flowfilterentry>
```

- JSON format

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        }
    }
}
```

Table 128: Description of request URI elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.	No
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame. • drop: Discards the frame. • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified. Note: The value of *action_type* is case insensitive.	No
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers,	No

Element	Description	Required
	and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.	
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7	No
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63	No
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No

Processing result

Response body None

1.19.2. Delete vBridge Interface Flow Filter Entry

This operation is used to delete a vBridge Interface Flow Filter Entry.

Processing request

Method	DELETE
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.xml
• JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.json

Table 129: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Request body	None

Processing result

Response body	None

126

1.19.3. Update vBridge Interface Flow Filter Entry

This operation is used to update a vBridge Interface Flow Filter Entry.

Processing request

Method	PUT
Request URI	• XML format

/vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/
{ff_type}/flowfilterentries/{seqnum}.xml

• JSON format

/vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/
{ff_type}/flowfilterentries/{seqnum}.json

Table 130: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Request body	• XML format

```
<flowfilterentry fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
```

```
     <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}" />
</flowfilterentry>
```

- JSON format

```
{
    "flowfilterentry": {
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        }
    }
}
```

Table 131: Description of request elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.	No
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified. Note: The value of *action_type* is case insensitive.	No
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.	No

Element	Description	Required
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7	No
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63	No
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No

Processing result

  Response body   None

1.19.4. List vBridge Interface Flow Filter Entries

This operation is used to list vBridge Interface Flow Filter Entry information.

Processing request

Method	GET
Request URI	

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries.xml

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/detail.xml

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/count.xml

- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries.json

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/detail.json

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/count.json

- Query string

  ?index={seqnum}&max_repetition={max_repetition}

Table 132: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Table 133: Description of query string elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	No
max_repetition	The number of the resources that hope to be returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries>
      <flowfilterentry seqnum="{seqnum}" />
  </flowfilterentries>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries>
      <flowfilterentry
   seqnum="{seqnum}" fl_name="{fl_name}"
   action_type="{action_type}"
   nmg_name="{nmg_name}" priority="{priority}"
   dscp="{dscp}">
          <redirectdst vnode_name="{vnode_name}"
   if_name="{if_name}" macdstaddr="{macdstaddr}"
   macsrcaddr="{macsrcaddr}" />
      </flowfilterentry>
  </flowfilterentries>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentry count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "flowfilterentries": [
          {
  ```

131

```
                "seqnum": "{seqnum}"
            }
        ]
}
```

If detail is specified in URI

```
{
    "flowfilterentries": [
        {
            "seqnum": "{seqnum}",
            "fl_name": "{fl_name}",
            "action_type": "{action_type}",
            "nmg_name": "{nmg_name}",
            "priority": "{priority}",
            "dscp": "{dscp}",
            "redirectdst": {
                "vnode_name": "{vnode_name}",
                "if_name": "{if_name}",
                "macdstaddr": "{macdstaddr}",
                "macsrcaddr": "{macsrcaddr}"
            }
        }
    ]
}
```

If count is specified in URI

```
{
    "flowfilterentries": {
        "count": "{count}"
    }
}
```

Table 134: Description of response elements

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame

Element	Description
	• redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified.
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63
count	The number of Flow Filter Entry. Valid value: A positive integer.
redirectdst	Redirect information.

Table 135: Description of response elements (redirectdst)

Element	Description
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab).

133

Element	Description
	Note: MAC address is case insensitive.

1.19.5. Show vBridge Interface Flow Filter Entry

This operation is used to view a specific vBridge Interface Flow Filter Entry information.

Processing request

Method GET

Request URI

- XML format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.xml

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}/detail.xml

- JSON format

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.json

  /vtns/{vtn_name}/vbridges/{vbr_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}/detail.json

- Query string

  ?controller_id={controller_id}

Table 136: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbr_name	vBridge name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Table 137: Description of query string elements

Element	Description	Required
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body None

Processing result

Response body
- XML format

    If count is not specified in URI

    ```
    <?xml version="1.0" encoding="UTF-8"
     standalone="yes"?>
    <flowfilterentry seqnum="{seqnum}"
     fl_name="{fl_name}"
     action_type="{action_type}"
     nmg_name="{nmg_name}" priority="{priority}"
     dscp="{dscp}">
        <redirectdst vnode_name="{vnode_name}"
     if_name="{if_name}" macdstaddr="{macdstaddr}"
     macsrcaddr="{macsrcaddr}" />
    </flowfilterentry>
    ```

    If detail is specified in URI

    ```
    <?xml version="1.0" encoding="UTF-8"
     standalone="yes"?>
    <flowfilterentry seqnum="{seqnum}"
     fl_name="{fl_name}"
     action_type="{action_type}"
     nmg_name="{nmg_name}" priority="{priority}"
     dscp="{dscp}" nmg_status="{nmg_status}">
        <redirectdst vnode_name="{vnode_name}"
     if_name="{if_name}" macdstaddr="{macsrcaddr}"
     macsrcaddr="{macsrcaddr}" />
        <statistics>
            <software packets="{packets}"
    octets="{octets}" />
            <existingflow packets="{packets}"
    octets="{octets}" />
            <expiredflow packets="{packets}"
     octets="{octets}" />
    ```

135

```
            <total packets="{packets}"
octets="{octets}" />
    </statistics>
    <flowlist>
        <flowlistentries>
            <flowlistentry seqnum="{seqnum}">
                <statistics>
                    <software
packets="{packets}" octets="{octets}" />
                    <existingflow
packets="{packets}" octets="{octets}" />
                    <expiredflow
packets="{packets}" octets="{octets}" />
                    <total packets="{packets}"
octets="{octets}" />
                </statistics>
            </flowlistentry>
        </flowlistentries>
    </flowlist>
</flowfilterentry>
```

- JSON format

  If count is not specified in URI

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        }
    }
}
```

  If detail is specified in URI

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name ": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "nmg_status": "{nmg_status}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        },
        "statistics": {
            "software": {
                "packets": "{packets}",
```

```
                "octets": "{octets}"
            },
            "existingflow": {
                "packets": "{packets}",
                "octets": "{octets}"
            },
            "expiredflow": {
                "packets": "{packets}",
                "octets": "{octets}"
            },
            "total": {
                "packets": "{packets}",
                "octets": "{octets}"
            }
        },
        "flowlist": {
            "flowlistentries": [
                {
                    "seqnum": "{seqnum}",
                    "statistics": {
                        "software": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "existingflow": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "expiredflow": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "total": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        }
                    }
                }
            ]
        }
    }
}
```

Table 138: Description of response elements (flowfilterentry)

Element	Description
seqnum	The sequence number.
	Valid value: A positive integer.
	Valid range: 1 - 65535

Element	Description
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified.
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.
priority	Priority value registered to the flow filter entry. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63
nmg_status	Status of monitored host. Valid value: 0 - 256. Value "1" is faulty state and other than "1" are not defined.
redirectdst	Redirect information.
statistics	Statistical information.
flowlist	Flow List information.

Table 139: Description of response elements (redirectdst)

Element	Description
vnode_name	Redirect destination virtual node name.

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.

Table 140: Description of response elements (statistics)

Element	Description
software	The number of packets or bytes that passed through the VTN among the flows that match the flow filter entry.
existingflow	The number of packets or bytes that are hard-transferred by the flow entry currently set to the OFS among the flows that match the flow filter entry.
expiredflow	The number of packets or bytes that are hard-transferred by the flow entry previously set to the OFS among the flows that match the flow filter entry.
total	Total number of packets or bytes of the flow that matches the flow filter entry. This is the sum of Software, and ExistingFlow, and ExpiredFlow.

Table 141: Description of response elements (flowlist)

Element	Description
flowlistentries	Flow List entry list.

Table 142: Description of response elements (flowlistentry)

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
statistics	Statistics information.

Table 143: Description of response elements (common)

Element	Description
packets	Number of frames. Valid value: A positive integer. Valid range: 0 - 18446744073709551615
octets	Number of octets in the frames. Valid value: A positive integer. Valid range: 0 - 18446744073709551615

1.20. vRouter functions

This section lists the vRouter functions.

1.20.1. Create vRouter

This operation is used to create a vRouter.

Processing request

Method	POST
Request URI	• XML format /vtns/{vtn_name}/vrouters.xml • JSON format /vtns/{vtn_name}/vrouters.json

Table 144: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<vrouter vrt_name="{vrt_name}"
 controller_id="{controller_id}"
 description="{description}"
 domain_id="{domain_id}" />
```

- JSON format

```
{
    "vrouter": {
        "vrt_name": "{vrt_name}",
        "controller_id": "{controller_id}",
        "description": "{description}",
        "domain_id": "{domain_id}"
    }
}
```

Table 145: Description of request elements

Element	Description	Required
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.	Yes

Processing result

Response body   None

1.20.2. Delete vRouter

This operation is used to delete a vRouter.

Processing request

Method	DELETE
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}.xml
• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}.json

Table 146: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember

vRouter whose Interface is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Processing result

Response body   None

1.20.3. Update vRouter

This operation is used to update a vRouter.

Processing request

Method	PUT
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}.xml
• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}.json

Table 147: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<vrouter controller_id="{controller_id}"
 description="{description}"
 new_name="{new_name}"
 domain_id="{domain_id}" />
```

- JSON format

```
{
    "vrouter": {
        "controller_id": "{controller_id}",
        "description": "{description}",
        "domain_id": "{domain_id}"
    }
}
```

Table 148: Description of request elements

Element	Description	Required
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
description	Additional information. Valid value: A string of up to 127 characters.	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.	No

Processing result

Response body   None

1.20.4. List vRouters

This operation is used to list vRouter information based on specified conditions.

Processing request

Method	GET
Request URI	• XML format

   /vtns/{vtn_name}/vrouters.xml

   /vtns/{vtn_name}/vrouters/detail.xml

   /vtns/{vtn_name}/vrouters/count.xml
   • JSON format

   /vtns/{vtn_name}/vrouters.json

   /vtns/{vtn_name}/vrouters/detail.json

   /vtns/{vtn_name}/vrouters/count.json
   • Query string

   ?index={vrt_name}&max_repetition={max_repetition}

Table 149: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 150: Description of query string elements

Element	Description	Required
vtn_name	VTN name. The next instances of the specified key will be returned. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer	No

Element	Description	Required
	Valid range: 1 to MAX of UINT32. Default is 10000.	

Request body   None

Processing result

Response body
- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vrouters>
      <vrouter vrt_name="{vrt_name}" />
  </vrouters>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vrouters>
      <vrouter vrt_name="{vrt_name}"
   controller_id="{controller_id}"
   description="{description}" status="{status}"
   domain_id="{domain_id}" />
  </vrouters>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vrouters count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "vrouters": [
          {
              "vrt_name": "{vrt_name}"
          }
      ]
  }
  ```

  If detail is specified in URI

  ```
  {
      "vrouters": [
          {
              "vrt_name": "{vrt_name}",
              "controller_id": "{controller_id}",
              "description": "{description}",
              "status": "{status}",
              "domain_id": "{domain_id}"
          }
  ```

145

```
        ]
    }
```

If count is specified in URI

```
{
    "vrouters": {
        "count": "{count}"
    }
}
```

Table 151: Description of response elements

Element	Description
vrt_name	vRouter name.
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
status	vRouter status. Valid value: up, down, unknown
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.
count	The number of vRouter. Valid value: A positive integer.

1.20.5. Show vRouter

This operation is used to view a specific vRouter information.

Processing request

Method   GET

Request URI   • XML format

              /vtns/{vtn_name}/vrouters/{vrt_name}.xml
              • JSON format

              /vtns/{vtn_name}/vrouters/{vrt_name}.json

Table 152: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

None

Processing result

Response body

- XML format

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vrouter vrt_name="{vrt_name}"
 controller_id="{controller_id}"
 description="{description}" status="{status}"
 domain_id="{domain_id}"></vrouter>
```

- JSON format

```json
{
    "vrouter": {
        "vrt_name": "{vrt_name}",
        "controller_id": "{controller_id}",
        "description": "{description}",
        "status": "{status}",
        "domain_id": "{domain_id}"
    }
}
```

Table 153: Description of response elements

Element	Description
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: Up to 127 characters that can include alphabets, numbers, and underscore (except at the beginning).

147

Element	Description
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
status	vRouter status. Valid value: up, down, unknown
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning) and the string - (DEFAULT). Note: The value of *domain_id* is case sensitive.

1.21. Static IP Route functions

This section lists the Static IP Route functions.

1.21.1. Create Static IP Route

This operation is used to create a Static IP Route.

Processing request

Method	POST
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes.xml
• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes.json

Table 154: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<static_iproute ipaddr="{ipaddr}"
 prefix="{prefix}"
 nexthopaddr="{nexthopaddr}" />
```

- JSON format

```
{
    "static_iproute": {
        "ipaddr": "{ipaddr}",
        "prefix": "{prefix}",
        "nexthopaddr": "{nexthopaddr}"
    }
}
```

Table 155: Description of request elements

Element	Description	Required
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)	Yes
prefix	Prefix length. Valid value: A positive integer between 1 and 30.	Yes
nexthopaddr	Next hop address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)	Yes

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<static_iproute
 static_iproute_id="{static_iproute_id}" />
```

- JSON format

```
{
    "static_iproute": {
        "static_iproute_id":
 "{static_iproute_id}"
    }
}
```

Table 156: Description of response elements

Element	Description
static_iproute_id	Static IP Route identifier.

149

Element	Description
	Valid value: {ipaddr}-{nexthopaddr}-{prefix}

1.21.2. Delete Static IP Route

This operation is used to delete a Static IP Route.

Processing request

Method	DELETE
Request URI	• XML format /vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes/{static_iproute_id}.xml • JSON format /vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes/{static_iproute_id}.json

Table 157: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
static_iproute_id	Static IP Route identifier. Valid value: {ipaddr}-{nexthopaddr}-{prefix}	Yes

Request body	None

Processing result

Response body	None

1.21.3. List Static IP Routes

This operation is used to list Static IP Route information based on specified conditions.

Processing request

Method	GET
Request URI	• XML format

150

/vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes.xml

/vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes/detail.xml

/vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes/count.xml
- JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes.json

/vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes/detail.json

/vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes/count.json
- Query string

?index={static_iproute_id}&max_repetition={max_repetition}

Table 158: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 159: Description of query string elements

Element	Description	Required
static_iproute_id	Static IP Route identifier. Valid value: {ipaddr}-{nexthopaddr}-{prefix}	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
  standalone="yes"?>
<static_iproutes>
```

151

```
      <static_iproute
 static_iproute_id="{static_iproute_id}" />
</static_iproutes>
```

If detail is specified in URI

```
<static_iproutes>
     <static_iproute
 static_iproute_id="{static_iproute_id}"
 ipaddr="{ipaddr}" prefix="{prefix}"
 nexthopaddr="{nexthopaddr}" />
</static_iproutes>
```

If count is specified in URI

```
<static_iproutes count="{count}" />
```

- JSON format

    If count is not specified in URI

```
{
    "static_iproutes": [
        {
            "static_iproute_id":
 "{static_iproute_id}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "static_iproutes": [
        {
            "static_iproute_id":
 "{static_iproute_id}",
            "ipaddr": "{ipaddr}",
            "prefix": "{prefix}",
            "nexthopaddr": "{nexthopaddr}"
        }
    ]
}
```

If count is specified in URI

```
{
    "static_iproutes": {
        "count": "{count}"
    }
}
```

Table 160: Description of response elements

Element	Description
static_iproute_id	Static IP Route identifier. Valid value: {ipaddr}-{nexthopaddr}-{prefix}

Element	Description
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
prefix	Prefix length. Valid value: A positive integer between 1 and 30.
count	The number of vRouter. Valid value: A positive integer.

1.21.4. Show Static IP Route

This operation is used to view a specific Static IP Route information.

Processing request

Method GET

Request URI

- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes/{static_iproute_id}.xml

- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/static_iproutes/{static_iproute_id}.json

Table 161: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
static_iproute_id	Static IP Route identifier. Valid value: {ipaddr}-{nexthopaddr}-{prefix}	Yes

Request body None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<static_iproute
 static_iproute_id="{static_iproute_id}"
 ipaddr="{ipaddr}" prefix="{prefix}"
 nexthopaddr="{nexthopaddr}" />
```

- JSON format

```
{
    "static_iproute": {
        "static_iproute_id":
 "{static_iproute_id}",
        "ipaddr": "{ipaddr}",
        "prefix": "{prefix}",
        "nexthopaddr": "{nexthopaddr}"
    }
}
```

Table 162: Description of response elements

Element	Description
static_iproute_id	Static IP Route identifier. Valid value: {ipaddr}-{nexthopaddr}-{prefix}
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
prefix	Prefix length. Valid value: A positive integer between 1 and 30.
nexthopaddr	Next hop address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)

1.22. IP Routes function

This section lists the IP Routes function.

1.22.1. Show IP Routes

This operation is used to view a specific IP Routes information.

Processing request

Method   GET

Request URI
- XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/iproutes.xml

/vtns/{vtn_name}/vrouters/{vrt_name}/iproutes/count.xml

- JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/iproutes.json

/vtns/{vtn_name}/vrouters/{vrt_name}/iproutes/count.json

- Query string

?type={type}

Table 163: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body
- XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<iproutes>
   [
   <iproute dstaddr="{dstaddr}"
 gateway="{gateway}" prefix="{prefix}"
 flags="{flags}" metric="{metric}" use="{use}"
 if_name="{if_name}" nmg_name="{nmg_name}"
 groupmetric="{groupmetric}" />
   ]
</iproutes>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<iproutes count="{count}" />
```

- JSON format

If count is not specified in URI

```
{
```

155

```
      "iproutes": [
            {
                  "dstaddr": "{dstaddr}",
                  "gateway": "{gateway}",
                  "prefix": "{prefix}",
                  "flags": "{flags}",
                  "metric": "{metric}",
                  "use": "{use}",
                  "if_name": "{if_name}",
                  "nmg_name": "{nmg_name}",
                  "groupmetric": "{groupmetric}"
            }
      ]
}
```

If count is specified in URI

```
{
      "iproutes": {
            "count": "{count}"
      }
}
```

Table 164: Description of response elements (Interface)

Element	Description
dstaddr	Destination IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
gateway	Gateway IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
prefix	Prefix length of the destination IP address. Valid value: A positive integer. Valid range: 1 - 30
flags	Route status. Valid value: hexadecimal integer Valid range: 0x0000-0xffff
metric	Priority order for multiple paths. Valid value: A positive integer. Valid range: 0 - 65535
use	Number of lookups on the route. Valid value: A positive integer. Valid range: 0 - UINT32_MAX .
if_name	Interface that receives packets through this route.

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
groupmetric	Priorities for multiple paths. Valid value: A positive integer. Valid range: 0 - 65535
count	Number of IP route. Valid value: A positive integer. Valid range: 1 - 65535

1.23. ARP Entry functions

This section lists the ARP Entry functions.

1.23.1. Show ARP Entries

This operation is used to view a specific ARP Entry information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/arpentries.xml

/vtns/{vtn_name}/vrouters/{vrt_name}/arpentries/count.xml

• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/arpentries.json

/vtns/{vtn_name}/vrouters/{vrt_name}/arpentries/count.json

• Query string

?type={type}

Table 165: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
type	Displays the learning type of the ARP entry. Valid values: • dynamic: The learning type is dynamic. • static: The learning type is static.	Yes

Request body

None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <arpentries>
      [
      <arpentry ipaddr="{ipaddr}"
   macaddr="{macaddr}" type="{type}"
   if_name="{if_name}" />
      ]
  </arpentries>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <arpentries count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "arpentries ": [
          {
              "ipaddr": "{ipaddr}",
              "macaddr": "{macaddr}",
              "type": "{type}",
              "if_name": "{if_name}"
          }
      ]
  }
  ```

  If count is specified in URI

  ```
  {
      "arpentries": {
          "count": "{count}"
  ```

158

```
        }
    }
```

Table 166: Description of response elements

Element	Description
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
macaddr	The MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
type	Displays the learning type of the ARP entry. Valid values: • dynamic: The learning type is dynamic. • static: The learning type is static.
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
count	Number of entries. Valid value: 1 - 65535

1.24. DHCP Relay Status functions

This section lists the DHCP Relay Status functions.

1.24.1. Show DHCP Relay Status

This operation is used to view the DHCP Relay Status.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay.xml
  • JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay.json

Table 167: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<dhcprelay
 dhcp_relay_status="{dhcp_relay_status}" />
```

- JSON format

```
{
    "dhcprelay": {
        "dhcp_relay_status":
 "{dhcp_relay_status}"
    }
}
```

Table 168: Description of response elements

Element	Description
dhcp_relay_status	DHCP Relay status. Valid value: enable, disable

1.24.2. Enable or Disable DHCP Relay Status

This operation is used to enable or disable the DHCP Relay Status.

Processing request

Method   PUT

Request URI

- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay.xml
- JSON format

160

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay.json

Table 169: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
  <dhcprelay
 dhcp_relay_status="{dhcp_relay_status}" />
```

- JSON format

```
{
    "dhcprelay": {
        "dhcp_relay_status":
 "{dhcp_relay_status}"
    }
}
```

Table 170: Description of request elements

Element	Description
dhcp_relay_status	DHCP Relay status. Valid value: enable, disable

Processing result

Response body   None

1.25. DHCP Relay Interface functions

This section lists the DHCP Relay Interface functions.

1.25.1. Create DHCP Relay Interface

This operation is used to create a DHCP Relay Interface.

Processing request

Method	POST
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces.xml
• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces.json

Table 171: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   • XML format

```
<interface if_name="{if_name}"/>
```
• JSON format

```
{
    "interface": {
        "if_name": "{if_name}"
    }
}
```

Table 172: Description of request URI elements

Element	Description	Required
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Processing result

Response body	None

1.25.2. Delete DHCP Relay Interface

This operation is used to delete a DHCP Relay Interface.

Processing request

Method	DELETE
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces/{if_name}.xml

• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces/{if_name}.json

Table 173: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body	None

Processing result

Response body	None

1.25.3. List DHCP Relay Interfaces

This operation is used to list DHCP Relay Interface information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces.xml

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces/count.xml

• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces.json

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces/count.json

- Query string

  ?index={if_name}&max_repetition={max_repetition}

Table 174: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 175: Description of query string elements

Element	Description	Required
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
max_repetition	The number of the resources that are returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body　　　None

Processing result

Response body

- XML format

  If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interfaces>
    <interface if_name="{if_name}"
 status="{status}" />
</interfaces>
```

  If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
```

```
<interfaces count="{count}" />
```

- JSON format

If count is not specified in URI

```
{
    "interfaces": [
        {
            "if_name": "{if_name}",
            "status": "{status}"
        }
    ]
}
```

If count is specified in URI

```
{
    "interfaces": {
        "count": "{count}"
    }
}
```

Table 176: Description of response elements

Element	Description
interface	Interface
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
status	Status. Valid value: inactive, active, starting, waiting, error.
count	Number of Interfaces. Valid value: A positive integer.

1.25.4. Show DHCP Relay Interface

This operation is used to view a specific DHCP Relay Interface information.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces/
  {if_name}.xml
- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/interfaces/
  {if_name}.json

165

Table 177: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interface if_name="{if_name}"
 status="{status}" />
```

- JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "status": "{status}"
    }
}
```

Table 178: Description of response elements

Element	Description
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
status	Status.

Element	Description
	Valid value: inactive, active, starting, waiting, error.

1.26. DHCP Relay Server functions

This section lists the DHCP Relay Server functions.

1.26.1. Create DHCP Relay Server

This operation is used to create a DHCP Relay Server.

Processing request

Method	POST
Request URI	• XML format /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers.xml • JSON format /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers.json

Table 179: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body	• XML format

```
<server ipaddr="{ipaddr}"/>
```

• JSON format

```
{
    "server": {
        "ipaddr": "{ipaddr}"
    }
}
```

Table 180: Description of request URI elements

Element	Description	Required
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)	Yes

Processing result

Response body   None

1.26.2. Delete DHCP Relay Server

This operation is used to delete a DHCP Relay Server.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers/{ipaddr}.xml
- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers/{ipaddr}.json

Table 181: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)	Yes

Request body   None

Processing result

Response body   None

1.26.3. List DHCP Relay Servers

This operation is used to list DHCP Relay Server information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers.xml

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers/count.xml

• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers.json

/vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers/count.json

• Query string

?index={ipaddr}&max_repetition={max_repetition}

Table 182: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 183: Description of query string elements

Element	Description	Required
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)	Yes
max_repetition	The number of the resources that are returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body	None

Processing result

Response body	• XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<servers>
    <server ipaddr="{ipaddr}" />
</servers>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<servers count="{count}" />
```

- JSON format

If count is not specified in URI

```
{
    "servers": [
        {
            "ipaddr": "{ipaddr}"
        }
    ]
}
```

If count is specified in URI

```
{
    "servers": {
        "count": "{count}"
    }
}
```

Table 184: Description of response elements

Element	Description
server	Server
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
count	Number of Interfaces. Valid value: A positive integer.

1.26.4. Show DHCP Relay Server

This operation is used to view a specific DHCP Relay Server information.

Processing request

Method　　　　　GET

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers/{ipaddr}.xml

170

- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/dhcprelay/servers/{ipaddr}.json

Table 185: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)	Yes

Request body

None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<server ipaddr="{ipaddr}" />
```

- JSON format

```
{
    "server": {
        "ipaddr": "{ipaddr}"
    }
}
```

Table 186: Description of response elements

Element	Description
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)

1.27. vRouter Interface functions

This section lists the vRouter Interface functions.

1.27.1. Create vRouter Interface

This operation is used to create a vRouter Interface.

Processing request

Method	POST
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces.xml
• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces.json

Table 187: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

• XML format

```
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}" ipaddr="{ipaddr}"
 prefix="{prefix}" macaddr="{macaddr}" />
```

• JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "ipaddr": "{ipaddr}",
        "prefix": "{prefix}",
        "macaddr": "{macaddr}"
    }
}
```

Table 188: Description of request elements

Element	Description	Required
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format. Example: 192.168.1.1 Note: The value of this parameter must be unique within vRouter.	No
prefix	Prefix length. Valid value: A positive integer. Valid range: 1 - 30. Note: This parameter is required if *ipaddr* is specified.	No
macaddr	The MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.). Example: 0123.4567.89ab Note: MAC address is case insensitive.	No

Processing result

Response body   None

1.27.2. Delete vRouter Interface

This operation is used to delete a vRouter Interface.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}.json

Table 189: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember
vRouter Interface that is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Remember
vRouter Interface that is set in *if_name* of DHCP Relay Interface cannot be deleted.

Processing result

Response body   None

1.27.3. List vRouter Interfaces

This operation is used to list vRouter Interface information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces.xml

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/detail.xml

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/count.xml
- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces.json

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/detail.json

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/count.json
- Query string

  ?index={if_name}& max_repetition={max_repetition}

Table 190: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 191: Description of query string elements

Element	Description	Required
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body   None

Processing result

Response body
- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <interfaces>
      <interface if_name="{if_name}" />
  </interfaces>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <interfaces>
      <interface if_name="{if_name}"
   description="{description}"
   adminstatus="{adminstatus}"
  ```

175

```
  operstatus="{operstatus}" ipaddr="{ipaddr}"
 prefix="{prefix}" macaddr="{macaddr}">
        <neighbor vnode_name="{vnode_name}"
 if_name="{if_name}" vlk_name="{vlk_name}" />
    </interface>
</interfaces>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interfaces count="{count}" />
```

- JSON format

    If count is not specified in URI

```
{
    "interfaces": [
        {
            "if_name": "{if_name}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "interfaces": [
        {
            "if_name": "{if_name}",
            "description": "{description}",
            "adminstatus": "{adminstatus}",
            "operstatus": "{operstatus}",
            "ipaddr": "{ipaddr}",
            "prefix": "{prefix}",
            "macaddr": "{macaddr}",
            "neighbor": {
                "vnode_name": "{vnode_name}",
                "if_name": "{if_name}",
                "vlk_name": "{vlk_name}"
            }
        }
    ]
}
```

If count is specified in URI

```
{
    "interfaces": {
        "count": "{count}"
    }
}
```

Table 192: Description of response elements (Interface)

Element	Description
if_name	vRouter Interface name.

176

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
operstatus	Operational status. Valid value: up, down, unknown
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
prefix	Prefix length. Valid value: A positive integer. Valid range: 1 - 30.
macaddr	The MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
neighbor	Information about the neighbor.
count	The number of Interface. Valid value: A positive integer.

Table 193: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name.

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.27.4. Show vRouter Interface

This operation is used to view a specific vRouter Interface information.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interface/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interface/{if_name}.json

Table 194: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}"
 operstatus="{operstatus}" ipaddr="{ipaddr}"
 prefix="{prefix}" macaddr="{macaddr}">
```

178

```
      <neighbor vnode_name="{vnode_name}"
 if_name="{if_name}" vlk_name="{vlk_name}" />
</interface>
```

- JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "operstatus": "{operstatus}",
        "ipaddr": "{ipaddr}",
        "prefix": "{prefix}",
        "macaddr": "{macaddr}",
        "neighbor": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "vlk_name": "{vlk_name}"
        }
    }
}
```

Table 195: Description of response elements (Interface)

Element	Description
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
adminstatus	Admin status. Valid value: enable, disable.
operstatus	Operational status. Valid value: up, down, unknown
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
prefix	Prefix length. Valid value: A positive integer. Valid range: 1 - 30.
macaddr	The MAC address.

Element	Description
	Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
neighbor	Information about the neighbor.

Table 196: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.27.5. Update vRouter Interface

This operation is used to update a vRouter Interface.

Processing request

Method   PUT

Request URI   • XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}.xml
               • JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}.json

Table 197: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
if_name	vBridge Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<interface description="{description}"
  adminstatus="{adminstatus}" ipaddr="{ipaddr}"
  prefix="{prefix}" macaddr="{macaddr}" />
```

- JSON format

```
{
    "interface": {
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "ipaddr": "{ipaddr}",
        "prefix": "{prefix}",
        "macaddr": "{macaddr}"
    }
}
```

Table 198: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No
ipaddr	IPv4 address. Valid value: IPv4 dot-separated format. Example: 192.168.1.1 Note: The value of this parameter must be unique within vRouter.	No
prefix	Prefix length. Valid value: A positive integer. Valid range: 1 - 30. Note: This parameter is required if *ipaddr* is specified.	No
macaddr	The MAC address.	No

Element	Description	Required
	Valid value: Three groups of four hexadecimal digits separated by dots (.). Example: 0123.4567.89ab Note: MAC address is case insensitive.	
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.28. vRouter Interface Flow Filter functions

This section lists the vRouter Interface Flow Filter functions.

1.28.1. Create vRouter Interface Flow Filter

This operation is used to create a vRouter Interface Flow Filter.

Processing request

Method   POST

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters.xml
- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters.json

Table 199: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	

Request body

- XML format

```
<flowfilter ff_type="{ff_type}" />
```

- JSON format

```
{
    "flowfilter": {
        "ff_type": "{ff_type}"
    }
}
```

Table 200: Description of request elements

Element	Description	Required
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Processing result

Response body   None

1.28.2. Delete vRouter Interface Flow Filter

This operation is used to delete a vRouter Interface Flow Filter.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/ {ff_type}.xml
- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/ {ff_type}.json

Table 201: Description of request URI elements

Element	Description	Required
vtn_name	VTN name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied. Valid value: in, out. Note: The value of *ff_type* is case insensitive.	Yes

Request body None

Processing result

Response body None

1.28.3. Show vRouter Interface Flow Filter

This operation is used to view a specific vRouter Interface Flow Filter information.

Processing request

Method GET

Request URI
- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/{ff_type}.xml
- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/{ff_type}.json

Table 202: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Request body   None

Processing result

Response body   • XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilter ff_type="{ff_type}" />
```

• JSON format

```
{
    "flowfilter": {
        "ff_type": "{ff_type}"
    }
}
```

Table 203: Description of response elements

Element	Description
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.

185

1.29. vRouter Interface Flow Filter Entry functions

This section lists the vRouter Interface Flow Filter Entry functions.

1.29.1. Create vRouter Interface Flow Filter Entry

This operation is used to create a vRouter Interface Flow Filter Entry.

Processing request

Method	POST
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries.xml

• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries.json

Table 204: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Request body	• XML format

```
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
```

```
      <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}" />
</flowfilterentry>
```

- JSON format

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        }
    }
}
```

Table 205: Description of request URI elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.	No
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame. • drop: Discards the frame. • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified. Note: The value of *action_type* is case insensitive.	No
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers,	No

Element	Description	Required
	and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.	
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7	No
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63	No
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No

Processing result

Response body   None

1.29.2. Delete vRouter Interface Flow Filter Entry

This operation is used to delete a vRouter Interface Flow Filter Entry.

Processing request

Method	DELETE
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/ {ff_type}/flowfilterentries/{seqnum}.xml
• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/ {ff_type}/flowfilterentries/{seqnum}.json

Table 206: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Request body	None

Processing result

Response body	None

1.29.3. Update vRouter Interface Flow Filter Entry

This operation is used to update a vRouter Interface Flow Filter Entry.

Processing request

Method	PUT
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/
{ff_type}/flowfilterentries/{seqnum}.xml
* JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/
{ff_type}/flowfilterentries/{seqnum}.json

Table 207: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Request body	• XML format

```
<flowfilterentry fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
```

```
    <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}" />
</flowfilterentry>
```

• JSON format

```
{
    "flowfilterentry": {
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        }
    }
}
```

Table 208: Description of request elements

Element	Description	Required
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.	No
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified. Note: The value of *action_type* is case insensitive.	No
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.	No

191

Element	Description	Required
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7	No
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63	No
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.	No

Processing result

Response body   None

1.29.4. List vRouter Interface Flow Filter Entries

This operation is used to list vRouter Interface Flow Filter Entry information.

Processing request

Method	GET
Request URI	

- XML format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/
  {ff_type}/flowfilterentries.xml

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/
  {ff_type}/flowfilterentries/detail.xml

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/
  {ff_type}/flowfilterentries/count.xml

- JSON format

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/
  {ff_type}/flowfilterentries.json

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/
  {ff_type}/flowfilterentries/detail.json

  /vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/
  {ff_type}/flowfilterentries/count.json

- Query string

  ?index={seqnum}&max_repetition={max_repetition}

Table 209: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

Table 210: Description of query string elements

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	No
max_repetition	The number of the resources that hope to be returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries>
      <flowfilterentry seqnum="{seqnum}" />
  </flowfilterentries>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentries>
      <flowfilterentry
   seqnum="{seqnum}" fl_name="{fl_name}"
   action_type="{action_type}"
   nmg_name="{nmg_name}" priority="{priority}"
   dscp="{dscp}">
          <redirectdst vnode_name="{vnode_name}"
   if_name="{if_name}" macdstaddr="{macdstaddr}"
   macsrcaddr="{macsrcaddr}" />
      </flowfilterentry>
  </flowfilterentries>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <flowfilterentry count= "{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "flowfilterentries": [
          {
  ```

194

```
                    "seqnum": "{seqnum}"
            }
        ]
}
```

If detail is specified in URI

```
{
    "flowfilterentries": [
        {
            "seqnum": "{seqnum}",
            "fl_name": "{fl_name}",
            "action_type": "{action_type}",
            "nmg_name": "{nmg_name}",
            "priority": "{priority}",
            "dscp": "{dscp}",
            "redirectdst": {
                "vnode_name": "{vnode_name}",
                "if_name": "{if_name}",
                "macdstaddr": "{macdstaddr}",
                "macsrcaddr": "{macsrcaddr}"
            }
        }
    ]
}
```

If count is specified in URI

```
{
    "flowfilterentries": {
        "count": "{count}"
    }
}
```

Table 211: Description of response elements

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
fl_name	Flow List name. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame

Element	Description
	• redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified.
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
priority	Priority value registered to the Flow Filter entry. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63
count	The number of Flow Filter Entry. Valid value: A positive integer.
redirectdst	Redirect information.

Table 212: Description of response elements (redirectdst)

Element	Description
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	A virtual interface of a redirect destination virtual node. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
macdstaddr	Destination MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
macsrcaddr	Source MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab).

Element	Description
	Note: MAC address is case insensitive.

1.29.5. Show vRouter Interface Flow Filter Entry

This operation is used to view a specific vRouter Interface Flow Filter Entry information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.xml

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}/detail.xml

• JSON format

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}.json

/vtns/{vtn_name}/vrouters/{vrt_name}/interfaces/{if_name}/flowfilters/{ff_type}/flowfilterentries/{seqnum}/detail.json

• Query string

?controller_id={controller_id}

Table 213: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vrt_name	vRouter name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vRouter Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
ff_type	Direction to which the Flow Filter is applied Valid value: in, out Note: The value of *ff_type* is case insensitive.	Yes

197

Element	Description	Required
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535	Yes

Table 214: Description of query string elements

Element	Description	Required
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

None

Processing result

Response body

- XML format

  If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
    <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macdstaddr}"
 macsrcaddr="{macsrcaddr}" />
</flowfilterentry>
```

  If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<flowfilterentry seqnum="{seqnum}"
 fl_name="{fl_name}"
 action_type="{action_type}"
 nmg_name="{nmg_name}" priority="{priority}"
 dscp="{dscp}">
    <redirectdst vnode_name="{vnode_name}"
 if_name="{if_name}" macdstaddr="{macsrcaddr}"
 macsrcaddr="{macsrcaddr}" />
    <statistics>
        <software packets="{packets}"
octets="{octets}" />
        <existingflow packets="{packets}"
octets="{octets}" />
        <expiredflow packets="{packets}"
 octets="{octets}" />
```

198

```
                    <total packets="{packets}"
octets="{octets}" />
    </statistics>
    <flowlist>
        <flowlistentries>
            <flowlistentry seqnum="{seqnum}">
                <statistics>
                    <software
packets="{packets}" octets="{octets}" />
                    <existingflow
packets="{packets}" octets="{octets}" />
                    <expiredflow
packets="{packets}" octets="{octets}" />
                    <total packets="{packets}"
octets="{octets}" />
                </statistics>
            </flowlistentry>
        </flowlistentries>
    </flowlist>
</flowfilterentry>
```

- JSON format

  If count is not specified in URI

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        }
    }
}
```

  If detail is specified in URI

```
{
    "flowfilterentry": {
        "seqnum": "{seqnum}",
        "fl_name": "{fl_name}",
        "action_type": "{action_type}",
        "nmg_name ": "{nmg_name}",
        "priority": "{priority}",
        "dscp": "{dscp}",
        "redirectdst": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "macdstaddr": "{macdstaddr}",
            "macsrcaddr": "{macsrcaddr}"
        },
        "statistics": {
            "software": {
                "packets": "{packets}",
                "octets": "{octets}"
```

```
            },
            "existingflow": {
                "packets": "{packets}",
                "octets": "{octets}"
            },
            "expiredflow": {
                "packets": "{packets}",
                "octets": "{octets}"
            },
            "total": {
                "packets": "{packets}",
                "octets": "{octets}"
            }
        },
        "flowlist": {
            "flowlistentries": [
                {
                    "seqnum": "{seqnum}",
                    "statistics": {
                        "software": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "existingflow": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "expiredflow": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        },
                        "total": {
                            "packets":
"{packets}",
                            "octets":
"{octets}"
                        }
                    }
                }
            ]
        }
    }
}
```

Table 215: Description of response elements (flowfilterentry)

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535
fl_name	Flow List name.

Element	Description
	Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *fl_name* is case sensitive.
action_type	Action that is registered in the Flow Filter entry. Valid values: • pass: Passes the frame • drop: Discards the frame • redirect: Transfers a frame to the virtual interface of the virtual node in which the frame is specified.
nmg_name	Network monitor group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *nmg_name* is case sensitive.
priority	Priority value registered to the flow filter entry. Valid value: A positive whole number. Valid range: 0 - 7
dscp	The DSCP value. Valid value: A positive whole number. Valid range: 0 - 63
redirectdst	Redirect information.
statistics	Statistical information.
flowlist	Flow List information.

Table 216: Description of response elements (redirectdst)

Element	Description
vnode_name	Redirect destination virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	A virtual interface of a redirect destination virtual node.

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
macdstaddr	Destination MAC address.
	Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab).
	Note: MAC address is case insensitive.
macsrcaddr	Source MAC address.
	Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab).
	Note: MAC address is case insensitive.

Table 217: Description of response elements (statistics)

Element	Description
software	The number of packets or bytes that passed through the VTN among the flows that match the flow filter entry.
existingflow	The number of packets or bytes that are hard-transferred by the flow entry currently set to the OFS among the flows that match the flow filter entry.
expiredflow	The number of packets or bytes that are hard-transferred by the flow entry previously set to the OFS among the flows that match the flow filter entry.
total	Total number of packets or bytes of the flow that matches the flow filter entry. This is the sum of Software, and ExistingFlow, and ExpiredFlow.

Table 218: Description of response elements (flowlist)

Element	Description
flowlistentries	Flow List entry list.

Table 219: Description of response elements (flowlistentry)

Element	Description
seqnum	The sequence number. Valid value: A positive integer. Valid range: 1 - 65535

Element	Description
statistics	Statistics information.

Table 220: Description of response elements (common)

Element	Description
packets	Number of frames.
	Valid value: A positive integer.
	Valid range: 0 - 18446744073709551615
octets	Number of octets in the frames.
	Valid value: A positive integer.
	Valid range: 0 - 18446744073709551615

1.30. vBypass functions

This section lists the vBypass functions.

1.30.1. Create vBypass

This operation is used to create a vBypass.

Processing request

Method   POST

Request URI
- XML format

  /vtns/{vtn_name}/vbypasses.xml
- JSON format

  /vtns/{vtn_name}/vbypasses.json

Table 221: Description of request URI element

Element	Description	Required
vtn_name	VTN name.	Yes
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	

Request body
- XML format

```
<vbypass vbypass_name="{vbypass_name}"
  description="{description}" type="{type}"
  controller_id="{controller_id}"
  domain_id="{domain_id}"/>
```
- JSON format

```
{
```

```
     "vbypass": {
         "vbypass_name": "{vbypass_name}",
         "description": "{description}",
         "type": "{type}",
         "controller_id": "{controller_id}",
         "domain_id": "{domain_id}"
     }
}
```

Table 222: Description of request elements

Element	Description	Required
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
type	Type of vBypass node. Valid value: bridge, router.	No
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.	Yes

Processing result

Response body   None

1.30.2. Delete vBypass

This operation is used to delete a vBypass.

Processing request

Method   DELETE

Request URI
- XML format

/vtns/{vtn_name}/vbypasses/{vbypass_name}.xml
- JSON format

/vtns/{vtn_name}/vbypasses/{vbypass_name}.json

Table 223: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body None

Processing result

Response body None

1.30.3. Update vBypass

This operation is used to update a vBypass.

Processing request

Method PUT

Request URI
- XML format

/vtns/{vtn_name}/vbypasses/{vbypass_name}.xml
- JSON format

/vtns/{vtn_name}/vbypasses/{vbypass_name}.json

Table 224: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	

Request body

- XML format

```
<vbypass description="{description}"
 type="{type}" controller_id="{controller_id}"
 domain_id="{domain_id}"/>
```

- JSON format

```
{
    "vbypass": {
        "description": "{description}",
        "type": "{type}",
        "controller_id": "{controller_id}",
        "domain_id": "{domain_id}"
    }
}
```

Table 225: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No
type	Type of vBypass node. Valid value: bridge, router	No
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.	No

Processing result

Response body   None

206

1.30.4. List vBypass

This operation is used to list vBypass information based on specified conditions.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vbypasses.xml

/vtns/{vtn_name}/vbypasses/detail.xml

/vtns/{vtn_name}/vbypasses/count.xml
• JSON format

/vtns/{vtn_name}/vbypasses.json

/vtns/{vtn_name}/vbypasses/detail.json

/vtns/{vtn_name}/vbypasses/count.json
• Query string

?index={vbypass_name}&max_repetition={max_repetition}

Table 226: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 227: Description of query string elements

Element	Description	Required
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body	None

Processing result

Response body	• XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vbypasses>
[
<vbypass vbypass_name="{vbypass_name}"/>
]
</vbypasses>
```

If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vbypasses>
[
<vbypass vbypass_name="{vbypass_name}"
 description="{description}" type="{type}"
 controller_id="{controller_id}"
 domain_id="{domain_id}"/>
]
</vbypasses>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vbypasses count="{count}"/>
```

- JSON format

  If count is not specified in URI

```
{
    "vbypasses": [
        {
            "vbypass_name": "{vbypass_name}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "vbypasses": [
        {
            "vbypass_name": "{vbypass_name}",
            "description": "{description}",
            "type": "{type}",
            "controller_id": "{controller_id}",
            "domain_id": "{domain_id}"
        }
    ]
}
```

If count is specified in URI

```
{
    "vbypasses": {
        "count": "{count}"
```

208

```
      }
   }
}
```

Table 228: Description of response elements

Element	Description
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
type	Type of vBypass node. Valid value: bridge, router.
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.
count	The number of vBypass. Valid value: A positive integer.

1.30.5. Show vBypass

This operation is used to view a specific vBypass information.

Processing request

Method	GET
Request URI	• XML format /vtns/{vtn_name}/vbypasses/{vbypass_name}.xml • JSON format /vtns/{vtn_name}/vbypasses/{vbypass_name}.json

Table 229: Description of request URI elements

Element	Description	Required
vtn_name	VTN name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body   • XML format

```
<?xml version="1.0" encoding="UTF-8"
  standalone="yes"?>
<vbypass vbypass_name="{vbypass_name}"
  description="{description}" type="{type}"
  controller_id="{controller_id}"
  domain_id="{domain_id}"/>
```

• JSON format

```
{
    "vbypass": {
        "   vbypass_name": "{vbypass_name}",
        "description": "{description}",
        "type": "{type}",
        "controller_id": "{controller_id}",
        "domain_id": "{domain_id}"
    }
}
```

Table 230: Description of response elements

Element	Description
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
type	Type of vBypass node. Valid value: bridge, router
controller_id	Controller identifier.

210

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.

1.31. vBypass Interface functions

This section lists the vBypass Interface functions.

1.31.1. Create vBypass Interface

This operation is used to create a vBypass Interface.

Processing request

Method　　　　　POST

Request URI
- XML format

  /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces.xml
- JSON format

  /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces.json

Table 231: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body
- XML format

```
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}"/>
```

211

- JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}"
    }
}
```

Table 232: Description of request elements

Element	Description	Required
if_name	vBypass Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.31.2. Delete vBypass Interface

This operation is used to delete a vBypass Interface.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/{if_name}.json

Table 233: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Element	Description	Required
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBypass Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember

vBypass Interface that is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Processing result

Response body   None

1.31.3. Update vBypass Interface

This operation is used to update a vBypass Interface.

Processing request

Method   PUT

Request URI
- XML format

  /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/{if_name}.json

Table 234: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBypass Interface name.	Yes

213

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	

Request body

- XML format

```
<interface description="{description}"
  adminstatus="{adminstatus}"/>
```

- JSON format

```
{
    "interface": {
        "description": "{description}",
        "adminstatus": "{adminstatus}"
    }
}
```

Table 235: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.31.4. List vBypass Interfaces

This operation is used to list vBypass Interface information based on specified conditions.

Processing request

Method   GET

Request URI

- XML format

   /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces.xml

   /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/detail.xml

   /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/count.xml

- JSON format

   /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces.json

   vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/detail.json

   /vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/count.json

- Query string

?index={if_name}&max_repetition={max_repetition}

Table 236: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 237: Description of query string elements

Element	Description	Required
if_name	vBypass Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interfaces>
[
<interface if_name="{if_name}"/>
]
</interfaces>
```

If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interfaces>
[
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}"><neighbor
 vnode_name="{vnode_name}" if_name="{if_name}"
 vlk_name="{vlk_name}"/></interface>
]
</interfaces>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interfaces count="{count}"/>
```

- JSON format

    If count is not specified in URI

```
{
    "interfaces": [
        {
            "if_name": "{if_name}"
        }
    ]
}
```

    If detail is specified in URI

```
{
    "interfaces": [
        {
            "if_name": "{if_name}",
            "description": "{description}",
            "adminstatus": "{adminstatus}",
            "neighbor": {
                "vnode_name": "{vnode_name}",
                "if_name": "{if_name}",
                "vlk_name": "{vlk_name}"
            }
        }
    ]
}
```

    If count is specified in URI

```
{
    "interfaces": {
        "count": "{count}"
    }
}
```

Table 238: Description of response elements (Interface)

Element	Description
if_name	vBypass Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
neighbor	Information about the neighbor.
count	The number of vBypass Interface. Valid value: A positive integer.

Table 239: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	vBypass Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.31.5. Show vBypass Interface

This operation is used to view a specific vBypass Interface information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/{if_name}.xml
• JSON format

/vtns/{vtn_name}/vbypasses/{vbypass_name}/interfaces/{if_name}.json

Table 240: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vbypass_name	vBypass name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vBypass Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}">
<neighbor vnode_name="{vnode_name}"
 if_name="{if_name}" vlk_name="{vlk_name}"/>
</interface>
```

- JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "neighbor": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "vlk_name": "{vlk_name}"
        }
    }
}
```

218

Table 241: Description of response elements (Interface)

Element	Description
if_name	vBypass Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
neighbor	Information about the neighbor.

Table 242: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	vBypass Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.32. vTep functions

This section lists the vTep functions.

1.32.1. Create vTep

This operation is used to create a vTep.

Processing request

Method	POST
Request URI	• XML format /vtns/{vtn_name}/vteps.xml • JSON format /vtns/{vtn_name}/vteps.json

Table 243: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<vtep vtep_name="{vtep_name}"
 controller_id="{controller_id}"
 description="{description}"
 domain_id="{domain_id}" />
```

- JSON format

```
{
    "vtep": {
        "vtep_name": "{vtep_name}",
        "controller_id": "{controller_id}",
        "description": "{description}",
        "domain_id": "{domain_id}"
    }
}
```

Table 244: Description of request elements

Element	Description	Required
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Element	Description	Required
	Note: The value of *domain_id* is case sensitive.	

Processing result

Response body   None

1.32.2. Delete vTep

This operation is used to delete a vTep.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vteps/{vtep_name}.xml
- JSON format

  /vtns/{vtn_name}/vteps/{vtep_name}.json

Table 245: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember
vTep whose Interface is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Remember
vTep that is set in *member_vteps* of vTep Group cannot be deleted.

Processing result

Response body   None

1.32.3. Update vTep

This operation is used to update a vTep.

Processing request

Method	PUT
Request URI	• XML format

/vtns/{vtn_name}/vteps/{vtep_name}.xml
• JSON format

/vtns/{vtn_name}/vteps/{vtep_name}.json

Table 246: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   • XML format

```
<vtep controller_id="{controller_id}"
 description="{description}"
 domain_id="domain_id}"/>
```

• JSON format

```
{
    "vtep": {
        "controller_id": "{controller_id}",
        "description": "{description}",
        "domain_id": "{domain_id}"
    }
}
```

Table 247: Description of request elements

Element	Description	Required
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
description	Additional information. Valid value: A string of up to 127 characters.	No

222

Element	Description	Required
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.	No

Processing result

Response body   None

1.32.4. List vTeps

This operation is used to list vTep information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vteps.xml

  /vtns/{vtn_name}/vteps/detail.xml

  /vtns/{vtn_name}/vteps/count.xml
- JSON format

  /vtns/{vtn_name}/vteps.json

  /vtns/{vtn_name}/vteps/detail.json

  /vtns/{vtn_name}/vteps/count.json
- Query string

  ?index={vtep_name}&max_repetition={max_repetition}

Table 248: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 249: Description of query string elements

Element	Description	Required
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body   None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vteps>
      <vtep vtep_name="{vtep_name}" />
  </vteps>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vteps>
      <vtep vtep_name="{vtep_name}"
   controller_id="{controller_id}"
   description="{description}"
   operstatus="{operstatus}"
   domain_id="{domain_id}"></vtep>
  </vteps>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vteps count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "vteps": [
          {
              "vtep_name": "{vtep_name}"
          }
      ]
  }
  ```

  If detail is specified in URI

  ```
  {
  ```

224

```
    "vteps": [
        {
            "vtep_name": "{vtep_name}",
            "controller_id": "{controller_id}",
            "description": "{description}",
            "operstatus": "{operstatus}",
            "domain_id": "{domain_id}"
        }
    ]
}
```

If count is specified in URI

```
{
    "vteps": {
        "count": "{count}"
    }
}
```

Table 250: Description of response elements

Element	Description
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
operstatus	The vTep operational status. Valid value: up, down, unknown
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.
count	The number of vTep. Valid value: A positive integer.

1.32.5. Show vTep

This operation is used to view a specific vTep information.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vteps/{vtep_name}.xml
• JSON format

/vtns/{vtn_name}/vteps/{vtep_name}.json

Table 251: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body	None

Processing result

Response body	• XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtep vtep_name="{vtep_name}"
 controller_id="{controller_id}"
 description="{description}"
 operstatus="{operstatus}"
 domain_id="{domain_id}"></vtep>
```

• JSON format

```
{
    "vtep": {
        "vtep_name": "{vtep_name}",
        "controller_id": "{controller_id}",
        "description": "{description}",
        "operstatus": "{operstatus}",
        "domain_id": "{domain_id}"
    }
}
```

Table 252: Description of response elements

Element	Description
vtep_name	vTep name.

226

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
operstatus	The vTep operational status. Valid value: up, down, unknown
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.

1.33. vTep Interface functions

This section lists the vTep Interface functions.

1.33.1. Create vTep Interface

This operation is used to create a vTep Interface.

Processing request

Method	POST
Request URI	• XML format

/vtns/{vtn_name}/vteps/{vtep_name}/interfaces.xml
• JSON format

/vtns/{vtn_name}/vteps/{vtep_name}/interfaces.json

Table 253: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	

Request body

- XML format

```
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}" />
```

- JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}"
    }
}
```

Table 254: Description of request elements

Element	Description	Required
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.33.2. Delete vTep Interface

This operation is used to delete a vTep Interface.

Processing request

Method   DELETE

Request URI

- XML format

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}.xml

- JSON format

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}.json

Table 255: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember
vTep Interface that is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Processing result

Response body   None

1.33.3. Update vTep Interface
This operation is used to update a vTep Interface.

Processing request

Method   PUT

Request URI

- XML format

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}.json

Table 256: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<interface description="{description}"
adminstatus="{adminstatus}" />
```

- JSON format

```
{
    "interface": {
        "description": "{description}",
        "adminstatus": "{adminstatus}"
    }
}
```

Table 257: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.33.4. List vTep Interfaces

This operation is used to list vTep Interface information based on specified conditions.

Processing request

Method   GET

Request URI

- XML format

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces.xml

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/detail.xml

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/count.xml
- JSON format

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces.json

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/detail.json

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/count.json
- Query string

  ?index={if_name}&max_repetition={max_repetition}

Table 258: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 259: Description of query string elements

Element	Description	Required
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

If count is not specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interfaces>
    <interface if_name="{if_name}" />
</interfaces>
```

If detail is specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interfaces>
    [
    <interface if_name="{if_name}"
        description="{description}"
        adminstatus="{adminstatus}"
        operstatus="{operstatus}">
        <neighbor vnode_name="{vnode_name}"
            if_name="{if_name}"
            vlk_name="{vlk_name}" />
    </interface>
    ]
</interfaces>
```

If count is specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interfaces count="{count}" />
```

• JSON format

If count is not specified in URI

```json
{
    "interfaces": [
        {
            "if_name": "{if_name}"
        }
    ]
}
```

If detail is specified in URI

```json
{
    "interfaces": [
        {
            "if_name": "{if_name}",
            "description": "{description}",
            "adminstatus": "{adminstatus}",
            "operstatus": "{operstatus}",
            "neighbor": {
                "vnode_name": "{vnode_name}",
                "if_name": "{if_name}",
                "vlk_name": "{vlk_name}"
            }
        }
    ]
}
```

232

If count is specified in URI

```
{
    "interfaces": {
        "count": "{count}"
    }
}
```

Table 260: Description of response elements

Element	Description
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
operstatus	The link-up status of the interface. Valid value: up, down, unknown
neighbor	Information about the neighbor.
count	The number of vTep Interfaces. Valid value: A positive integer.

Table 261: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.33.5. Show vTep Interface

This operation is used to view a specific vTep information Interface.

Processing request

Method	GET
Request URI	• XML format

/vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}.xml
• JSON format

/vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}.json

Table 262: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body	None

Processing result

Response body	• XML format

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}"
 operstatus="{operstatus}">
    <neighbor vnode_name="{vnode_name}"
 if_name="{if_name}" vlk_name="{vlk_name}" />
</interface>
```

• JSON format

```json
{
    "interface": {
        "if_name": "{if_name}",
```

```
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "operstatus": "{operstatus}",
        "neighbor": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "vlk_name": "{vlk_name}"
        }
    }
}
```

Table 263: Description of response elements

Element	Description
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
operstatus	The link-up status of the interface. Valid value: up, down, unknown
neighbor	Information about the neighbor.

Table 264: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.34. vTep Interface Port Map functions

This section lists the vTep Interface Port Map functions.

1.34.1. Delete vTep Interface Port Map

This operation is used to delete a vTep Interface Port Map.

Processing request

Method	DELETE
Request URI	• XML format

/vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}/portmap.xml
• JSON format

/vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}/portmap.json

Table 265: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body	None

Processing result

Response body	None

1.34.2. Set vTep Interface Port Map

This operation is used to update a vTep Interface Port Map for specific settings.

Processing request

Method	PUT
Request URI	• XML format

/vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}/portmap.xml
- JSON format

/vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}/portmap.json

Table 266: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<?xml version="1.0"?>
<portmap logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}" tagged="{tagged}"/>
```

- JSON format

```
{
    "portmap": {
        "logical_port_id": "{logical_port_id}",
        "vlan_id": "{vlan_id}",
        "tagged": "{tagged}"
    }
}
```

Table 267: Description of request elements

Element	Description	Required
logical_port_id	Logical port identifier. Valid value: A string of up to 319 characters.	Yes
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095	No

Element	Description	Required
	Note: This parameter is required if *tagged* is specified.	
tagged	Displays whether VLAN tags are sent and received in the physical network. Valid value: • true: Send and receive packets with VLANtags. • false: Send and receive packets without VLAN tags.	No

Processing result

Response body   None

1.34.3. Show vTep Interface Port Map

This operation is used to view a specific vTep Interface Port Map information.

Processing request

Method   GET

Request URI

• XML format

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}/portmap.xml
• JSON format

  /vtns/{vtn_name}/vteps/{vtep_name}/interfaces/{if_name}/portmap.json

Table 268: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<portmap logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}" tagged="{tagged}" />
```

- JSON format

```
{
    "portmap": {
        "logical_port_id": "{logical_port_id}",
        "vlan_id": "{vlan_id}",
        "tagged": "{tagged}"
    }
}
```

Table 269: Description of response elements

Element	Description
logical_port_id	Logical switch identifier. Valid value: A string of up to 319 characters.
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095
tagged	Displays whether VLAN tags are sent and received in the physical network . Valid value: • true: Send and receive packets with VLANtags. • false: Send and receive packets without VLAN tags.

1.35. vTep Group functions

This section lists the vTep Group functions.

1.35.1. Create vTep Group

This operation is used to create a vTep Group.

Processing request

Method   POST

Request URI
- XML format

  /vtns/{vtn_name}/vtepgroups.xml
- JSON format

  /vtns/{vtn_name}/vtepgroups.json

Table 270: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body
- XML format

```
<vtepgroup vtepgroup_name="{vtep_name}"
 controller_id="{controller_id}"
 description="{description}">
    <member_vteps>
        <member_vtep vtep_name="{vtep_name}" />
    </member_vteps>
</vtepgroup>
```

- JSON format

```
{
    "vtepgroup": {
        "vtepgroup_name": "{vtep_name}",
        "controller_id": "{controller_id}",
        "description": "{description}",
        "member_vteps": [
            {
                "vtep_name": "{vtep_name}"
            }
        ]
    }
}
```

Table 271: Description of request elements

Element	Description	Required
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
description	Additional information about the vTep. Valid value: Up to 127 characters that can include alphabets, numbers, and underscore (except at the beginning).	No

Processing result

Response body   None

1.35.2. Delete vTep Group

This operation is used to delete a vTep Group.

Processing request

Method   DELETE

Request URI

- XML format

  /vtns/{vtn_name}/vtepgroups/{vtepgroup_name}.xml
- JSON format

  /vtns/{vtn_name}/vtepgroups/{vtepgroup_name}.json

Table 272: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember

vTep Group that is set in *vtepgroup_name* of vTunnel cannot be deleted.

Processing result

Response body   None

1.35.3. Update vTep Group

This operation is used to update a vTep Group.

Processing request

Method   PUT

Request URI   • XML format

/vtns/{vtn_name}/vtepgroups/{vtepgroup_name}.xml
• JSON format

/vtns/{vtn_name}/vtepgroups/{vtepgroup_name}.json

Table 273: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   • XML format

```xml
<vtepgroup>
    <member_vteps>
        <member_vtep vtep_name="{vtep_name}" />
    </member_vteps>
</vtepgroup>
```

• JSON format

```json
{
    "vtepgroup": {
        "member_vteps": [
            {
                "vtep_name": "{vtep_name}"
            }
        ]
    }
}
```

Table 274: Description of request elements

Element	Description	Required
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No

Processing result

Response body   None

1.35.4. List vTep Groups

This operation is used to list vTep Group information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vtepgroups.xml

  /vtns/{vtn_name}/vtepgroups/detail.xml

  /vtns/{vtn_name}/vtepgroups/count.xml
- JSON format

  /vtns/{vtn_name}/vtepgroups.json

  /vtns/{vtn_name}/vtepgroups/detail.json

  /vtns/{vtn_name}/vtepgroups/count.json
- Query string

  ?index={vtepgroup_name}&max_repetition={max_repetition}

Table 275: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 276: Description of query string elements

Element	Description	Required
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

  If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtepgroups>
    <vtepgroup
 vtepgroup_name="{vtepgroup_name}" />
</vtepgroups>
```

  If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtepgroups>
    <vtepgroup
 vtepgroup_name="{vtepgroup_name}"
 controller_id="{controller_id}"
 description="{description}">
        <member_vteps>
            <member_vtep
 vtep_name="{vtep_name}" />
        </member_vteps>
    </vtepgroup>
</vtepgroups>
```

  If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtepgroups count="{count}" />
```

- JSON format

  If count is not specified in URI

```
{
    "vtepgroups": [
        {
            "vtepgroup_name":
 "{vtepgroup_name}"
        }
```

244

```
        ]
    }
```

If detail is specified in URI

```
{
    "vtepgroups": [
        {
            "vtepgroup_name":
 "{vtepgroup_name}",
            "controller_id": "{controller_id}",
            "description": "{description}",
            "member_vteps": [
                {
                    "vtep_name": "{vtep_name}"
                }
            ]
        }
    ]
}
```

If count is specified in URI

```
{
    "vtepgroups": {
        "count": "{count}"
    }
}
```

Table 277: Description of response elements (vtepgroups)

Element	Description
vtepgroup	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
count	The number of vTep Group. Valid value: A positive integer.

Table 278: Description of response elements (vtepgroup)

Element	Description
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information.

Element	Description
	Valid value: A string of up to 127 characters.
member_vteps	member_vteps list. Valid value: A string of up to 127 characters.

Table 279: Description of response elements (vtepgroups)

Element	Description
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.35.5. Show vTep Group

This operation is used to view a specific vTep Group information.

Processing request

Method   GET

Request URI

- XML format

  /vtns/{vtn_name}/vtepgroups/{vtepgroup_name}.xml
- JSON format

  /vtns/{vtn_name}/vtepgroups/{vtepgroup_name}.json

Table 280: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body

- XML format

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vtepgroup_name="{vtep_name}"
 controller_id="{controller_id}"
 description="{description}">
    <member_vteps>
        <member_vtep_name="{vtep_name}" />
    </member_vteps>
</vtepgroup>
```

- JSON format

```json
{
    "vtepgroup": {
        "vtepgroup_name": "{vtepgroup_name}",
        "controller_id": "{controller_id}",
        "description": "{description}",
        "member_vteps": [
            {
                "vtep_name": "{vtep_name}"
            }
        ]
    }
}
```

Table 281: Description of response elements

Element	Description
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vtep_name	vTep name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.

1.36. vTunnel functions

This section lists the vTunnel functions.

1.36.1. Create vTunnel

This operation is used to create a vTunnel.

Processing request

Method	POST
Request URI	• XML format

/vtns/{vtn_name}/vtunnels.xml
 • JSON format

/vtns/{vtn_name}/vtunnels.json

Table 282: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body　　　　• XML format

```
<vtunnel vtunnel_name="{vtunnel_name}"
 description="{description}"
 controller_id="{controller_id}"
 vtn_name="{vtn_name}"
 vtepgroup_name="{vtepgroup_name}"
 label="{label}" domain_id="{domain_id}" />
```

 • JSON format

```
{
    "vtunnel": {
        "vtunnel_name": "{vtunnel_name}",
        "description": "{description}",
        "controller_id": "{controller_id}",
        "vtn_name": "{vtn_name}",
        "vtepgroup_name": "{vtepgroup_name}",
        "label": "{label}",
        "domain_id": "{domain_id}"
    }
}
```

Table 283: Description of request elements

Element	Description	Required
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information.	No

248

Element	Description	Required
	Valid value: A string of up to 127 characters.	
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.	Yes
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.	No
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.	No
label	Label to identify the tunnel. Valid value: A positive integer. Valid range: 0 - 4294967295.	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.	Yes

Processing result

Response body   None

1.36.2. Delete vTunnel

This operation is used to delete a vTunnel.

Processing request

Method   DELETE

Request URI
- XML format

    /vtns/{vtn_name}/vtunnels/{vtunnel_name}.xml
- JSON format

    /vtns/{vtn_name}/vtunnels/{vtunnel_name}.json

Table 284: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember
vTunnel whose Interface is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Processing result

Response body   None

1.36.3. Update vTunnel

This operation is used to update a vTunnel.

Processing request

Method   PUT

Request URI
- XML format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}.xml
- JSON format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}.json

Table 285: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<vtunnel description="{description}"
 controller_id="{controller_id}"
 vtn_name="{vtn_name}"
 vtepgroup_name="{vtepgroup_name}"
 label="{label}" domain_id="{domain_id}" />
```

- JSON format

```
{
    "vtunnel": {
        "description": "{description}",
        "controller_id": "{controller_id}",
        "vtn_name": "{vtn_name}",
        "vtepgroup_name": "{vtepgroup_name}",
        "label": "{label}",
        "domain_id": "{domain_id}"
    }
}
```

Table 286: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
label	Label to identify the tunnel. Valid value: A positive integer. Valid range: 0 - 4294967295.	No
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers,	No

Element	Description	Required
	and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.	

Processing result

Response body   None

1.36.4. List vTunnels

This operation is used to list vTunnel information based on specified conditions.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vtunnels.xml

  /vtns/{vtn_name}/vtunnels/detail.xml

  /vtns/{vtn_name}/vtunnels/count.xml
- JSON format

  /vtns/{vtn_name}/vtunnels.json

  /vtns/{vtn_name}/vtunnels/detail.json

  /vtns/{vtn_name}/vtunnels/count.json
- Query string

  ?index={vtunnel_name}&max_repetition={max_repetition}

Table 287: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 288: Description of query string elements

Element	Description	Required
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Element	Description	Required
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body   None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vtunnels>
      <vtunnel vtunnel_name="{vtunnel_name}" />
  </vtunnels>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vtunnels>
      <vtunnel vtunnel_name="{vtunnel_name}"
   description="{description}"
   controller_id="{controller_id}"
   vtn_name="{vtn_name}"
   vtepgroup_name="{vtepgroup_name}"
   label="{label}" domain_id="{domain_id}"
   operstatus="{operstatus}" />
  </vtunnels>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <vtunnels count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "vtunnels": [
          {
              "vtunnel_name": "{vtunnel_name}"
          }
      ]
  }
  ```

  If detail is specified in URI

  ```
  {
      "vtunnels": [
  ```

```
        {
            "vtunnel_name": "{vtunnel_name}",
            "description": "{description}",
            "controller_id": "{controller_id}",
            "vtn_name": "{vtn_name}",
            "vtepgroup_name":
 "{vtepgroup_name}",
            "label": "{label}",
            "domain_id": "{domain_id}",
            "operstatus": "{operstatus}"
        }
    ]
}
```

If count is specified in URI

```
{
    "vtunnels": {
        "count": "{count}"
    }
}
```

Table 289: Description of response elements

Element	Description
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
label	Label to identify the tunnel. Valid value: A positive integer. Valid range: 0 - 4294967295.

Element	Description
domain_id	Domain identifier.
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
	Note: The value of *domain_id* is case sensitive.
operstatus	Operational status.
	Valid value: up, down, unknown
count	The number of vTunnel.
	Valid value: A positive integer.

1.36.5. Show vTunnel

This operation is used to view a specific vTunnel information.

Processing request

Method   GET

Request URI

- XML format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}.xml
- JSON format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}.json

Table 290: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
  standalone="yes"?>
```

255

```
<vtunnel vtunnel_name="{vtunnel_name}"
 description="{description}"
 controller_id="{controller_id}"
 vtn_name="{vtn_name}"
 vtepgroup_name="{vtepgroup_name}"
 label="{label}" domain_id="{domain_id}"></
vtunnel>
```

- JSON format

```
{
    "vtunnel": {
        "vtunnel_name": "{vtunnel_name}",
        "description": "{description}",
        "controller_id": "{controller_id}",
        "vtn_name": "{vtn_name}",
        "vtepgroup_name": "{vtepgroup_name}",
        "label": "{label}",
        "domain_id": "{domain_id}",
        "operstatus": "{operstatus}"
    }
}
```

Table 291: Description of response elements

Element	Description
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
controller_id	Controller identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vtepgroup_name	vTep Group name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
label	Label to identify the tunnel. Valid value: A positive integer. Valid range: 0 - 4294967295.

Element	Description
domain_id	Domain identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: The value of *domain_id* is case sensitive.
operstatus	Operational status. Valid value: up, down, unknown

1.37. vTunnel Interface functions

This section lists the vTunnel Interface functions.

1.37.1. Create vTunnel Interface

This operation is used to create a vTunnel Interface.

Processing request

Method	POST
Request URI	• XML format /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces.xml • JSON format /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces.json

Table 292: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body	• XML format

```
<interface if_name="{if_name}"
    description="{description}"
    adminstatus="{adminstatus}" />
```

- JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}"
    }
}
```

Table 293: Description of request elements

Element	Description	Required
if_name	vTunnel Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.37.2. Delete vTunnel Interface
This operation is used to delete a vTunnel Interface.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}.json

Table 294: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Element	Description	Required
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body　　　　　None

Remember
　　vTunnel Interface that is set in *if1_name* or *if2_name* of vLink cannot be deleted.

Processing result

Response body　　　　None

1.37.3. Update vTunnel Interface

This operation is used to update a vTunnel Interface.

Processing request

Method　　　　　　PUT

Request URI
- XML format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}.json

Table 295: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.	Yes
if_name	vTep Interface name.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.	

Request body

- XML format

```
<interface  description ="{description}"
adminstatus="{adminstatus}" />
```

- JSON format

```
{
    "interface": {
        "description": "{description}",
        "adminstatus": "{adminstatus}"
    }
}
```

Table 296: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No

Processing result

Response body   None

1.37.4. List vTunnel Interfaces

This operation is used to list vTunnel Interface information based on specified conditions.

Processing request

Method   GET

Request URI

- XML format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces.xml

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/detail.xml

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/count.xml

- JSON format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces.json

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/detail.json

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/count.json

- Query string

?index={if_name}&max_repetition={max_repetition}

Table 297: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.	Yes

Table 298: Description of query string elements

Element	Description	Required
if_name	vTunnel Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body   None

Processing result

Response body
- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <interfaces>
      <interface if_name="{if_name}" />
  </interfaces>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <interfaces>
      <interface if_name="{if_name}"
   description="{description}"
  ```

261

```
  adminstatus="{adminstatus}"
  operstatus="{operstatus}">
          <neighbor vnode_name="{vnode_name}"
  if_name="{if_name}" vlk_name="{vlk_name}" />
      </interface>
</interfaces>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
  standalone="yes"?>
<interfaces count="{count}" />
```

- JSON format

    If count is not specified in URI

```
{
    "interfaces": [
        {
            "if_name": "{if_name}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "interfaces": [
        {
            "if_name": "{if_name}",
            "description": "{description}",
            "adminstatus": "{adminstatus}",
            "operstatus": "{operstatus}",
            "neighbor": {
                "vnode_name": "{vnode_name}",
                "if_name": "{if_name}",
                "vlk_name": "{vlk_name}"
            }
        }
    ]
}
```

If count is specified in URI

```
{
    "interfaces": {
        "count": "{count}"
    }
}
```

Table 299: Description of response elements (interfaces)

Element	Description
if_name	vTunnel Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

Element	Description
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
operstatus	The link-up status of the interface. Valid value: up, down, unknown
neighbor	Information about the neighbor.
count	The number of vTunnel Interfaces. Valid value: A positive integer.

Table 300: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	vTunnel Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.37.5. Show vTunnel Interface

This operation is used to view a specific vTunnel information Interface.

Processing request

Method GET

Request URI
- XML format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}.xml
- JSON format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}.json

Table 301: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	vTep Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<interface if_name="{if_name}"
 description="{description}"
 adminstatus="{adminstatus}"
 operstatus="{operstatus}">
    <neighbor vnode_name="{vnode_name}"
 if_name="{if_name}" vlk_name="{vlk_name}" />
</interface>
```

- JSON format

```
{
    "interface": {
        "if_name": "{if_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "operstatus": "{operstatus}",
        "neighbor": {
            "vnode_name": "{vnode_name}",
            "if_name": "{if_name}",
            "vlk_name": "{vlk_name}"
        }
    }
}
```

264

Table 302: Description of response elements (interface)

Element	Description
if_name	vTunnel Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
operstatus	The link-up status of the interface. Valid value: up, down, unknown
neighbor	Information about the neighbor.

Table 303: Description of response elements (neighbor)

Element	Description
vnode_name	Virtual node name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if_name	vTunnel Interface anme. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

1.38. vTunnel Interface Port Map functions

This section lists the vTunnel Interface Port Map functions.

1.38.1. Delete vTunnel Interface Port Map

This operation is used to delete a vTunnel Interface Port Map.

Processing request

Method	DELETE
Request URI	• XML format

/vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}/
portmap.xml
- JSON format

/vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}/
portmap.json

Table 304: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body   None

1.38.2. Set vTunnel Interface Port Map

This operation is used to update a vTunnel Interface Port Map for specific settings.

Processing request

Method   PUT

Request URI
- XML format

/vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}/
portmap.xml
- JSON format

/vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}/
portmap.json

Table 305: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```xml
<?xml version="1.0"?>
<portmap logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}" tagged="{tagged}"/>
```

- JSON format

```json
{
    "portmap": {
        "logical_port_id": "{logical_port_id}",
        "vlan_id": "{vlan_id}",
        "tagged": "{tagged}"
    }
}
```

Table 306: Description of request elements

Element	Description	Required
logical_port_id	Logical port identifier. Valid value: A string of up to 319 characters.	Yes
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095 Note: This parameter is required if *tagged* is specified.	No

267

Element	Description	Required
tagged	Displays whether VLAN tags are sent and received in the physical network. Valid value: • true: Send and receive packets with VLANtags. • false: Send and receive packets without VLAN tags.	No

Processing result

Response body   None

1.38.3. Show vTunnel Interface Port Map

This operation is used to view a specific vTunnel Interface Port Map information.

Processing request

Method   GET

Request URI
- XML format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}/portmap.xml
- JSON format

  /vtns/{vtn_name}/vtunnels/{vtunnel_name}/interfaces/{if_name}/portmap.json

Table 307: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vtunnel_name	vTunnel name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if_name	Interface name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body　　　　None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<portmap logical_port_id="{logical_port_id}"
 vlan_id="{vlan_id}" tagged="{tagged}"/>
```

- JSON format

```
{
    "portmap": {
        "logical_port_id": "{logical_port_id}",
        "vlan_id": "{vlan_id}",
        "tagged": "{tagged}"
    }
}
```

Table 308: Description of response elements

Element	Description
logical_port_id	Logical switch identifier. Valid value: A string of up to 319 characters.
vlan_id	Identifier of the mapped VLAN. Valid value: A positive integer. Valid range: 1 - 4095
tagged	Displays whether VLAN tags are sent and received in the physical network . Valid value: • true: Send and receive packets with VLANtags. • false: Send and receive packets without VLAN tags.

1.39. vLink functions

This section lists the vLink functions.

1.39.1. Create vLink

This operation is used to create a vLink.

Processing request

Method　　　　POST

Request URI

- XML format

  /vtns/{vtn_name}/vlinks.xml
- JSON format

  /vtns/{vtn_name}/vlinks.json

Table 309: Description of request URI element

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```xml
<vlink vlk_name="{vlk_name}"
 description="{description}"
 adminstatus="{adminstatus}"
 vnode1_name="{vnode1_name}"
 if1_name="{if1_name}"
 vnode2_name="{vnode2_name}"
 if2_name="{if2_name}">
    <boundary_map boundary_id="{boundary_id}"
 vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}" />
</vlink>
```

- JSON format

```json
{
    "vlink": {
        "vlk_name": "{vlk_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "vnode1_name": "{vnode1_name}",
        "if1_name": "{if1_name}",
        "vnode2_name": "{vnode2_name}",
        "if2_name": "{if2_name}",
        "boundary_map": {
            "boundary_id": "{boundary_id}",
            "vlan_id": "{vlan_id}",
            "no_vlan_id": "{no_vlan_id}"
        }
    }
}
```

Table 310: Description of request elements (vLink)

Element	Description	Required
vlk_name	Virtual link name. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No
vnode1_name	The name of one of the two virtual nodes linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if1_name	The name of the virtual interface of VTN node1 linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: Following interface cannot be specified in this parameter. • Interface that is set in if1_name or if2_name of other vLink. • Interface for which Port Map is set.	Yes
vnode2_name	The name of the virtual node that is not VTN node 1 of the two virtual nodes linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
if2_name	The name of the virtual interface of VTN node 2 linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning). Note: Following interface cannot be specified in this parameter.	Yes

Element	Description	Required
	• Interface that is set in if1_name or if2_name of other vLink. • Interface for which Port Map is set.	
boundary_map	Boundary map.	No

Table 311: Description of request elements (boundary_map)

Element	Description	Required
boundary_id	Boundary identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vlan_id	VLAN identifier. Valid value: A positive integer. Valid range: 1 - 4095	No
no_vlan_id	No VLAN ID. Valid value: true	No

Processing result

Response body   None

1.39.2. Delete vLink

This operation is used to delete a vLink.

Processing request

Method   DELETE

Request URI
- XML format

  /vtns/{vtn_name}/vlinks/{vlk_name}.xml
- JSON format

  /vtns/{vtn_name}/vlinks/{vlk_name}.json

Table 312: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

272

Element	Description	Required
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body None

Processing result

Response body None

1.39.3. Update vLink

This operation is used to update a vLink.

Processing request

Method PUT

Request URI

- XML format

  /vtns/{vtn_name}/vlinks/{vlk_name}.xml
- JSON format

  /vtns/{vtn_name}/vlinks/{vlk_name}.json

Table 313: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<vlink description="{description}"
 adminstatus="{adminstatus}">
    <boundary_map boundary_id="{boundary_id}"
 vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}" />
</vlink>
```

- JSON format

```
{
    "vlink": {
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "boundary_map": {
            "boundary_id": "{boundary_id}",
            "vlan_id": "{vlan_id}",
            "no_vlan_id": "{no_vlan_id}"
        }
    }
}
```

Table 314: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No
adminstatus	Admin status. Valid value: enable, disable.	No
boundary_map	Boundary map.	No

Table 315: Description of request elements (boundary_map)

Element	Description	Required
boundary_id	Boundary identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vlan_id	VLAN identifier. Valid value: A positive integer. Valid range: 1 - 4095	No
no_vlan_id	No VLAN ID. Valid value: true	No

Processing result

Response body   None

1.39.4. List vLinks

This operation is used to list vLink information based on specified conditions.

Processing request

Method	GET
Request URI	- XML format /vtns/{vtn_name}/vlinks.xml /vtns/{vtn_name}/vlinks/detail.xml /vtns/{vtn_name}/vlinks/count.xml - JSON format /vtns/{vtn_name}/vlinks.json /vtns/{vtn_name}/vlinks/detail.json /vtns/{vtn_name}/vlinks/count.json - Query string ?index={vlk_name}&max_repetition={max_repetition} &vnode1_name={vnode1_name}&vnode2_name={vnode2_name}

Table 316: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 317: Description of query string elements

Element	Description	Required
vlk_name	Virtual link name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	Number of the resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No
vnode1_name	The name of one of the two virtual nodes linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
vnode2_name	The name of the virtual node that is not VTN node 1 of the two virtual	No

275

Element	Description	Required
	nodes linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	

Request body   None

Processing result

Response body

- XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vlinks>
    <vlink vlk_name="{vlk_name}" />
</vlinks>
```

If detail is specified in URI

```
<vlink vlk_name="{vlk_name}"
 adminstatus="{adminstatus}"
 operstatus="{operstatus}"
 vnode1_name="{vnode1_name}"
 if1_name="{if1_name}"
 vnode2_name="{vnode2_name}"
 if2_name="{if2_name}"
 description="{description}">
    <boundary_map boundary_id="{boundary_id}"
 vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}" />
</vlink>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<vlinks count="{count}" />
```

- JSON format

If detail and count are not specified in URI

```
{
    "vlinks": [
        {
            "vlk_name": "{vlk_name}"
        }
    ]
}
```

If detail is specified in URI

```
{
```

276

```
     "vlinks": [
          {
               "vlk_name": "{vlk_name}",

               "adminstatus": "{adminstatus}",
               "operstatus": "{operstatus}",
               "vnode1_name": "{vnode1_name}",
               "if1_name": "{if1_name}",
               "vnode2_name": "{vnode2_name}",
               "if2_name": "{if2_name}",
               "description": "{description}",
               "boundary_map": {
                    "boundary_id": "{boundary_id}",
                    "vlan_id": "{vlan_id}",
                    "no_vlan_id": "{no_vlan_id}"
               }
          }
     ]
}
```

If count is specified in URI

```
{
     "vlinks": {
          "count": "{count}"
     }
}
```

Table 318: Description of response elements (vLink)

Element	Description
vlk_name	Virtual link name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
vnode1_name	The name of one of the two virtual nodes linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if1_name	The name of the virtual interface of VTN node1 linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).

277

Element	Description
vnode2_name	The name of the virtual node that is not VTN node 1 of the two virtual nodes linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if2_name	The name of the virtual interface of VTN node 2 linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
boundary_map	Boundary map.
count	The number of vLink.

Table 319: Description of response elements (boundary_map)

Element	Description
boundary_id	Boundary identifier. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlan_id	VLAN identifier. Valid value: A positive integer. Valid range: 1 - 4095
no_vlan_id	No VLAN ID. Valid value: true

1.39.5. Show vLink

This operation is used to view a specific vLink information.

Processing request

Method GET

Request URI
- XML format

  /vtns/{vtn_name}/vlinks/{vlk_name}.xml
- JSON format

  /vtns/{vtn_name}/vlinks/{vlk_name}.json

Table 320: Description of request URI elements

Element	Description	Required
vtn_name	VTN name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
vlk_name	vLink name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Processing result

Response body
- XML format

```
<vlink vlk_name="{vlk_name}"
 adminstatus="{adminstatus}"
 operstatus="{operstatus}"
 vnode1_name="{vnode1_name}"
 if1_name="{if1_name}"
 vnode2_name="{vnode2_name}"
 if2_name="{if2_name}"
 description="{description}">
    <boundary_map boundary_id="{boundary_id}"
 vlan_id="{vlan_id}"
 no_vlan_id="{no_vlan_id}" />
</vlink>
```

- JSON format

```
{
    "vlink": {
        "vlk_name": "{vlk_name}",
                "adminstatus": "{adminstatus}",
        "operstatus": "{operstatus}",
        "vnode1_name": "{vnode1_name}",
        "if1_name": "{if1_name}",
        "vnode2_name": "{vnode2_name}",
        "if2_name": "{if2_name}",
        "description": "{description}",
        "boundary_map": {
            "boundary_id": "{boundary_id}",
            "vlan_id": "{vlan_id}",
            "no_vlan_id": "{no_vlan_id}"
        }
    }
}
```

Table 321: Description of response elements (vLink)

Element	Description
vlk_name	Virtual link name. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	Admin status. Valid value: enable, disable.
operstatus	The link-up status of the virtual link. Valid value: up, down, unknown
vnode1_name	The name of one of the two virtual nodes linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if1_name	The name of the virtual interface of VTN node1 linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vnode2_name	The name of the virtual node that is not VTN node 1 of the two virtual nodes linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
if2_name	The name of the virtual interface of VTN node 2 linked through the virtual link. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
boundary_map	Boundary map.

Table 322: Description of response elements (boundary_map)

Element	Description
boundary_id	Boundary identifier.

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
vlan_id	VLAN identifier. Valid value: A positive integer. Valid range: 1 - 4095
no_vlan_id	No VLAN ID. Valid value: true

1.40. Physical Controller functions

This section lists the physical Controller functions.

1.40.1. Create Physical Controller

This operation is used to add a physical Controller information.

Processing request

Method   POST

Request URI
- XML format

  /controllers.xml
- JSON format

  /controllers.json

Request body
- XML format

```
<controller controller_id="{controller_id}"
 description="{description}" ipaddr="{ipaddr}"
 type="{type}" auditstatus="{auditstatus}"
 username="{username}" password="{password}"
 version="{version}" />
```
- JSON format

```
{
    "controller": {
        "controller_id": "{controller_id}",
        "description": "{description}",
        "ipaddr": "{ipaddr}",
        "type": "{type}",
        "auditstatus": "{auditstatus}",
        "username": "{username}",
        "password": "{password}",
        "version": "{version}"
    }
}
```

Table 323: Description of request elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	A small description of the Controller. Valid value: A string of up to 127 characters.	No
ipaddr	IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1) Note: This parameter cannot be specified if *type* is bypass.	No
type	Controller type. Valid value: bypass, pfc, odc	Yes
auditstatus	Audit status. Valid value: enable, disable Note: This parameter cannot be specified if *type* is bypass.	No
username	The user name you want to specify. Valid value: A string of up to 31 characters. Note: This parameter cannot be specified if *type* is bypass.	No
password	The password that corresponds to the specified user name. Valid value: A string of up to 256 characters. Note: This parameter cannot be specified if *type* is bypass.	No
version	Version of Controller. Valid value: A string of up to 31 characters that can include numbers and dot (.).	Yes

Remember

Controller cannot be created when specified *type* and *ipaddr* are same as existing controller.

Processing result

Response body   None

1.40.2. Delete Physical Controller

This operation is used to delete a physical controller.

Processing request

Method   DELETE

Request URI
- XML format

  /controllers/{controller_id}.xml
- JSON format

  /controllers/{controller_id}.json

Table 324: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

> **Remember**
> Physical Controller that is set in *controller_id* of vBridge, vRouter, vTep, vTep Group, and vTunnel cannot be deleted.

> **Remember**
> Physical Controller that is set in *controller1_id* or *controller2_id* of Boundary cannot be deleted.

Processing result

Response body   None

1.40.3. Update Physical Controller

This operation is used to update a physical Controller.

Processing request

Method   PUT

Request URI
- XML format

  /controllers/{controller_id}.xml
- JSON format

  /controllers/{controller_id}.json

Table 325: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body

- XML format

```
<controller description="{description}"
  ipaddr="{ipaddr}" auditstatus="{auditstatus}"
  username="{username}" password="{password}"
  version="{version}" />
```

- JSON format

```
{
    "controller": {
        "description": "{description}",
        "ipaddr": "{ipaddr}",
        "auditstatus": "{auditstatus}",
        "username": "{username}",
        "password": "{password}",
        "version": "{version}"
    }
}
```

Table 326: Description of request elements

Element	Description	Required
description	A small description of the Controller. Valid value: A string of up to 127 characters.	No
ipaddr	IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1) Note: This parameter cannot be specified if *type* of controller specified by *controller_id* is bypass.	No
auditstatus	Audit status. Valid value: enable, disable Note: This parameter cannot be specified if *type* of controller specified by *controller_id* is bypass.	No
username	The user name you want to specify.	No

Element	Description	Required
	Valid value: A string of up to 31 characters. Note: This parameter cannot be specified if *type* of controller specified by *controller_id* is bypass.	
password	The password that corresponds to the specified user name. Valid value: A string of up to 256 characters. Note: This parameter cannot be specified if *type* of controller specified by *controller_id* is bypass.	No
version	Version of Controller. Valid value: Up to 31 characters that can include numbers and dot (.).	No

Remember

Controller cannot be updated when specified *type* and *ipaddr* are same as existing controller.

Processing result

Response body   None

1.40.4. List Physical Controllers

This operation is used to list physical Controller information.

Processing request

Method   GET

Request URI
- XML format

  /controllers.xml

  /controllers/detail.xml

  /controllers/count.xml
- JSON format

  /controllers.json

  /controllers/detail.json

  /controllers/count.json
- Query string

  ?index={controller_id}&max_repetition={max_repetition}

Table 327: Description of query string elements

Element	Description	Required
controller_id	Identifier of the Controller.	No

285

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
max_repetition	Number of the Controllers that are returned. When the count is specified as "0", then the result is the total number of the Controller. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body　　None

Processing result

Response body
- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <controllers>
      <controller
   controller_id="{controller_id}" />
  </controllers>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <controllers>
      <controller controller_id="{controller_id}"
   description="{description}" ipaddr="{ipaddr}"
   type="{type}" auditstatus="{auditstatus}"
   username="{username}"
   password="{password}" version="{version}"
   actual_version="{actual_version}"
   operstatus="{operstatus}" />
  </controllers>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <controllers count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "controllers": [
          {
  ```

286

```
                    "controller_id": "{controller_id}"
            }
        ]
}
```

If detail is specified in URI

```
{
    "controllers": [
        {
            "controller_id": "{controller_id}",
            "description": "{description}",
            "ipaddr": "{ipaddr}",
            "type": "{type}",
            "auditstatus": "{auditstatus}",
            "username": "{username}",
            "password": "{password}",
            "version": "{version}",
            "actual_version":
 "{actual_version}",
            "operstatus": "{operstatus}"
        }
    ]
}
```

If count is specified in URI

```
{
    "controllers": {
        "count": "{count}"
    }
}
```

Table 328: Description of response elements

Element	Description
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
ipaddr	IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
type	Controller type . Valid value: bypass, pfc, odc
auditstatus	Audit status. Valid value: enable, disable

Element	Description
username	The user name you want to specify. Valid value: A string of up to 31 characters.
password	The password that corresponds to the specified user name. Valid value: A string of up to 256 characters.
version	Version of Controller. Valid value: A string of up to 31 characters that can include numbers and dot (.).
actual_version	Actual version of Controller. Valid value: A string of up to 31 characters that can include numbers and dot (.).
operstatus	The operational status. Valid value: up, down, waiting_audit, auditing
count	The number of Controllers. Valid value: A positive integer.

1.40.5. Show Physical Controller

This operation is used to view a specific physical Controller information.

Processing request

Method   GET

Request URI
- XML format

  /controllers/{controller_id}.xml
- JSON format

  /controllers/{controller_id}.json

Table 329: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

288

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<controller controller_id="{controller_id}"
 description="{description}" ipaddr="{ipaddr}"
 type="{type}" auditstatus="{auditstatus}"
 username="{username}"
 password="{password}" version="{version}"
 actual_version="{actual_version}"
 operstatus="{operstatus}" />
```

- JSON format

```
{
    "controller": {
        "controller_id": "{controller_id}",
        "description": "{description}",
        "ipaddr": "{ipaddr}",
        "type": "{type}",
        "auditstatus": "{auditstatus}",
        "username": "{username}",
        "password": "{password}",
        "version": "{version}",
        "actual_version": "{actual_version}",
        "operstatus": "{operstatus}"
    }
}
```

Table 330: Description of response elements

Element	Description
controller_id	Identifier of the Controller. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	A small description of the Controller. Valid value: A string of up to 127 characters.
ipaddr	IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
type	Controller type . Valid value: bypass, pfc, odc
auditstatus	Audit status. Valid value: enable, disable
username	The user name you want to specify. Valid value: A string of up to 31 characters.

Element	Description
password	The password that corresponds to the specified user name. Valid value: A string of up to 256 characters.
version	Version of Controller. Valid value: A string of up to 31 characters that can include numbers and dot (.).
actual_version	Actual version of Controller. Valid value: A string of up to 31 characters that can include numbers and dot (.).
operstatus	The operational status. Valid value: up, down, waiting_audit, auditing

1.41. Physical Switch functions

This section lists the physical Switch functions.

1.41.1. List Physical Switches

This operation is used to list physical Switch information.

Processing request

Method GET

Request URI
- XML format

  /controllers/{controller_id}/switches.xml

  /controllers/{controller_id}/switches/detail.xml

  /controllers/{controller_id}/switches/count.xml
- JSON format

  /controllers/{controller_id}/switches.json

  /controllers/{controller_id}/switches/detail.json

  /controllers/{controller_id}/switches/count.json
- Query string

  ?index={switch_id}&max_repetition={max_repetition}

Table 331: Description of request URI element

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 32 characters that can include alphabets, numbers,	No

Element	Description	Required
	and underscore (except at the beginning).	

Table 332: Description of query string elements

Element	Description	Required
switch_id	Identifier of the Switch. Valid value: A string of up to 255 characters.	No
max_repetition	Number of resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <switches>
      <switch switch_id="{switch_id}" />
  </switches>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <switches>
      <switch switch_id="{switch_id}"
   description="{description}" model="{model}"
   adminstatus="{adminstatus}" ipaddr="{ipaddr}"
   ipv6addr="{ipv6addr}" domain_id="{domain_id}"
   operstatus="{operstatus}"
   manufacturer="{manufacturer}"
   hardware="{hardware}" software="{software}"
   alarmsstatus="{alarmsstatus}" />
  </switches>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <switches count="{count}" />
  ```

- JSON format

If count is not specified in URI

```
{
    "switches": [
        {
            "switch_id": "{switch_id}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "switches": [
        {
            "switch_id": "{switch_id}",
            "description": "{description}",
            "model": "{model}",
            "adminstatus": "{adminstatus}",
            "ipaddr": "{ipaddr}",
            "ipv6addr": "{ipv6addr}",
            "domain_id": "{domain_id}",
            "operstatus": "{operstatus}",
            "manufacturer": "{manufacturer}",
            "hardware": "{hardware}",
            "software": "{software}",
            "alarmsstatus": "{alarmsstatus}"
        }
    ]
}
```

If count is specified in URI

```
{
    "switches": {
        "count": "{count}"
    }
}
```

Table 333: Description of response elements (switches)

Element	Description
switch	Switch list.
count	The number of Switches. Valid value: A positive integer.

Table 334: Description of response elements (switch)

Element	Description
switch_id	Identifier of the Switch. Valid value: A string of up to 255 characters.

Element	Description
controller_id	Identifier of the Controller. Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	A small description of the Switch. Valid value: A string of up to 127 characters.
model	The model. Valid value: A string of up to 15 characters.
adminstatus	The admin status. Valid value: up, down
ipv6addr	IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:). For example: 2001:0db8:bd05:01d2:288a:1fc0:0001:10ee
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
ipaddr	IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
operstatus	Operational status. Valid value: up, down, unknown
manufacturer	Manufacturer information. Valid value: A string of up to 255 characters.
hardware	Hardware information. Valid value: A string of up to 255 characters.
software	Software information. Valid value: A string of up to 255 characters.
alarmsstatus	Alarm information. Valid value: A hexadecimal number.

1.41.2. Show Physical Switch

This operation is used to view a specific physical Switch information.

Processing request

Method	GET
Request URI	• XML format

/controllers/{controller_id}/switches/{switch_id}.xml
• JSON format

/controllers/{controller_id}/switches/{switch_id}.json

Table 335: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
switch_id	Identifier of the Switch. Valid value: A string of up to 255 characters.	Yes

Request body	None

Processing result

Response body   • XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<switch switch_id="{switch_id}"
 description="{description}" model="{model}"
 adminstatus="{adminstatus}" ipaddr="{ipaddr}"
 ipv6addr="{ipv6addr}" domain_id="{domain_id}"
 operstatus="{operstatus}"
 manufacturer="{manufacturer}"
 hardware="{hardware}" software="{software}"
 alarmsstatus="{alarmsstatus}" />
```

• JSON format

```
{
    "switch": {
        "switch_id": "{switch_id}",
        "description": "{description}",
        "model": "{model}",
        "adminstatus": "{adminstatus}",
        "ipaddr": "{ipaddr}",
        "ipv6addr": "{ipv6addr}",
        "domain_id": "{domain_id}",
        "operstatus": "{operstatus}",
        "manufacturer": "{manufacturer}",
```

```
        "hardware": "{hardware}",
        "software": "{software}",
        "alarmsstatus": "{alarmsstatus}"
    }
}
```

Table 336: Description of response elements

Element	Description
switch_id	Identifier of the Switch. Valid value: A string of up to 255 characters.
description	Additional information. Valid value: A string of up to 127 characters.
model	The model. Valid value: A string of up to 15 characters.
adminstatus	The admin status. Valid value: up, down
ipaddr	IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
ipv6addr	IPv6 address. Valid value: IPv6 address with eight groups of four hexadecimal digits separated by colons(:) For example: 2001:0db8:bd05:01d2:288a:1fc0:0001:10ee
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
operstatus	Operational status. Valid value: up, down, unknown
manufacturer	Manufacturer information. Valid value: A string of up to 255 characters.
hardware	Hardware information. Valid value: A string of up to 255 characters.
software	Software information.

Element	Description
	Valid value: A string of up to 255 characters.
alarmsstatus	Alarm information. Valid value: A hexadecimal number.

1.42. Physical Port functions

This section lists the physical Port functions.

1.42.1. List Physical Ports

This operation is used to list physical Port information.

Processing request

Method GET

Request URI
- XML format

  /controllers/{controller_id}/switches/{switch_id}/ports.xml

  /controllers/{controller_id}/switches/{switch_id}/ports/detail.xml

  /controllers/{controller_id}/switches/{switch_id}/ports/count.xml
- JSON format

  /controllers/{controller_id}/switches/{switch_id}/ports.json

  /controllers/{controller_id}/switches/{switch_id}/ports/detail.json

  /controllers/{controller_id}/switches/{switch_id}/ports/count.json
- Query string

  ?index={port_name}&max_repetition={max_repetition}

Table 337: Description of request URI element

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
switch_id	Identifier of the Switch. Valid value: A string of up to 255 characters	No

Table 338: Description of query string elements

Element	Description	Required
port_name	Port name.	No

296

Element	Description	Required
	A string of up to 31 characters.	
max_repetition	Number of resources that are returned. Valid value: A positive integer Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body   None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <ports>
      <port port_name="{port_name}" />
  </ports>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <ports>
      <port port_name="{port_name}"
   description="{description}"
   adminstatus="{adminstatus}"
   direction="{direction}"
   trunk_allowed_vlan="{trunk_allowed_vlan}"
   port_id="{port_id}" operstatus="{operstatus}"
   macaddr="{macaddr}"
   speed="{speed}" duplex="{duplex}"
   alarmsstatus="{alarmsstatus}"
   logical_port_id="{logical_port_id}">
          <neighbor switch_id="{switch_id}"
   port_name="{port_name}" />
      </port>
  </ports>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <ports count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "ports": [
          {
              "port_name": "{port_name}"
          }
  ```

297

```
        ]
}
```

If detail is specified in URI

```
{
    "ports": [
        {
            "port_name": "{port_name}",
            "description": "{description}",
            "adminstatus": "{adminstatus}",
            "direction": "{direction}",
            "trunk_allowed_vlan":
 "{trunk_allowed_vlan}",
            "port_id": "{port_id}",
            "operstatus": "{operstatus}",
            "macaddr": "{macaddr}",
            "speed": "{speed}",
            "duplex": "{duplex}",
            "alarmsstatus": "{alarmsstatus}",
            "logical_port_id":
 "{logical_port_id}",
            "neighbor": {
                "switch_id": "{switch_id}",
                "port_name": "{port_name}"
            }
        }
    ]
}
```

If count is specified in URI

```
{
    "ports": {
        "count": "{count}"
    }
}
```

Table 339: Description of response elements (ports)

Element	Description
port_name	Port name of a Switch. Valid value: A string of up to 31 characters.
controller_id	Identifier of the Controller. Valid value: Up to 32 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
adminstatus	The admin status. Valid value: up, down
Direction	The direction.

Element	Description
	Valid value: internal, external, unknown
trunk_allowed_vlan	Valid value: A positive integer. Valid range: 0 - 65535 (UINT16_MAX)
port_id	Identifier of the Port. Valid value: A positive integer. Valid range: 0 - 4294967295 (UINT32_MAX)
operstatus	Operational status of Port. Valid value: up, down, unknown
macaddr	The MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
speed	The communication speed of the port. Valid value: A positive integer. Valid range: 0 - 18446744073709551615(UINT64_MAX)
duplex	The communication method of the port Valid value: • full : Full-duplex communication • half : Half-duplex communication
alarmsstatus	Alarm information. Valid value: A hexadecimal number.
Logical_port_id	Identifier of the logical Port. Valid value: A string of up to 319 characters.
neighbor	Neighbor
count	The number of port. Valid value: A positive integer.

Table 340: Description of response elements (neighbor)

Element	Description
switch_id	Identifier of the switch.

299

Element	Description
	Valid value: A string of up to 255 characters.
port_name	External Port name. Valid value: A string of up to 31 characters.

1.42.2. Show Physical Port

This operation is used to view a specific physical Port information.

Processing request

Method	GET
Request URI	• XML format /controllers/{controller_id}/switches/{switch_id}/ports/{port_name}.xml • JSON format /controllers/{controller_id}/switches/{switch_id}/ports/{port_name}.json

Table 341: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
switch_id	Identifier of the Switch. Valid value: A string of up to 255 characters.	Yes
port_name	Name of Switch Port. Valid value: A string of up to 31 characters.	Yes

Request body	None

Processing result

Response body	• XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<port port_name="{port_name}"
 description="{description}"
 adminstatus="{adminstatus}"
 direction="{direction}"
 trunk_allowed_vlan="{trunk_allowed_vlan}"
```

```
port_id="{port_id}"
operstatus="{operstatus}" macaddr="{macaddr}"
speed="{speed}" duplex="{duplex}"
alarmsstatus="{alarmsstatus}"
logical_port_id="{logical_port_id}">
    <neighbor switch_id="{switch_id}"
port_name="{port_name}" />
</port>
```

- JSON format

```
{
    "port": {
        "port_name": "{port_name}",
        "description": "{description}",
        "adminstatus": "{adminstatus}",
        "direction": "{direction}",
        "trunk_allowed_vlan":
 "{trunk_allowed_vlan}",
        "port_id": "{port_id}",
        "operstatus": "{operstatus}",
        "macaddr": "{macaddr}",
        "speed": "{speed}",
        "duplex": "{duplex}",
        "alarmsstatus": "{alarmsstatus}",
        "logical_port_id": "{logical_port_id}",
        "neighbor": {
            "switch_id": "{switch_id}",
            "port_name": "{port_name}"
        }
    }
}
```

Table 342: Description of response elements (port)

Element	Description
port_name	Port name of a Switch. Valid value: A string of up to 31 characters.
description	A small description of the Switch. Valid value: A string of up to 127 characters.
adminstatus	The admin status. Valid value: up, down
direction	The direction. Valid value: internal, external, unknown
trunk_allowed_vlan	Valid value: A positive integer. Valid range: 0 - 65535 (UINT16_MAX)
port_id	Identifier of the Port. Valid value: A positive integer.

Element	Description
	Valid range: 0 - 4294967295 (UINT32_MAX)
operstatus	Operational status of Port. Valid value: up, down, unknown
macaddr	The MAC address. Valid value: Three groups of four hexadecimal digits separated by dots (.) (Example: 0123.4567.89ab). Note: MAC address is case insensitive.
speed	The communication speed of the port. Valid value: A positive integer. Valid range: 0 - 18446744073709551615(UINT64_MAX)
duplex	The communication method of the port. Valid value: • full : Full-duplex communication • half : Half-duplex communication
alarmsstatus	Alarm information. Valid value: A hexadecimal number.
Logical_port_id	Identifier of the logical Port. Valid value: A string of up to 319 characters.
neighbor	Neighbor

Table 343: Description of response elements (neighbor)

Element	Description
switch_id	Identifier of the switch. Valid value: A string of up to 255 characters.
port_name	External Port name. Valid value: A string of up to 31 characters.

1.43. Physical Link functions

This section lists the physical Link functions.

1.43.1. List Physical Links

This operation is used to list physical Link information.

Processing request

Method	GET
Request URI	• XML format

/controllers/{controller_id}/links.xml

/controllers/{controller_id}/links/detail.xml

/controllers/{controller_id}/links/count.xml

• JSON format

/controllers/{controller_id}/links.json

/controllers/{controller_id}/links/detail.json

/controllers/{controller_id}/links/count.json

• Query string

?index={link_name}&max_repetition={max_repetition}

&switch1_id={switch1_id}&switch2_id={switch2_id}

Table 344: Description of request URI element

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 345: Description of query string elements

Element	Description	Required
link_name	Link name. Valid value: A string with the format {switch1_id}-.{port1_name}-.{switch2_id}.{port2_name}.	No
max_repetition	Number of resources that are returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No
switch1_id	Returns links that have the specified parameter. Valid value: A string of up to 255 characters.	No
switch2_id	Returns links that have the specified parameter.	No

Element	Description	Required
	Valid value: A string of up to 255 characters.	

Request body   None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <links>
      <link link_name="{link_name}" />
  </links>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <links>
      <link link_name="{link_name}"
   switch1_id="{switch1_id}"
   port1_name="{port1_name}"
   switch2_id="{switch2_id}"
   port2_name="{port2_name}"
   description="{description}"
   operstatus="{operstatus}" />
  </links>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <links count="{count}" />
  ```

- JSON format

  If count is not specified in URI

  ```
  {
      "links": [
          {
              "link_name": "{link_name}"
          }
      ]
  }
  ```

  If detail is specified in URI

  ```
  {
      "links": [
          {
              "link_name": "{link_name}",
              "switch1_id": "{switch1_id}",
              "port1_name": "{port1_name}",
  ```

304

```
                "switch2_id": "{switch2_id}",
                "port2_name": "{port2_name}",
                "description": "{description}",
                "operstatus": "{operstatus}"
        }
    ]
}
```

If count is specified in URI

```
{
    "links": {
        "count": "{count}"
    }
}
```

Table 346: Description of response elements (ports)

Element	Description
link_name	Link name. Valid value: A string with the format {switch1_id}-.{port1_name}-. {switch2_id}.{port2_name}.
switch1_id	A Switch identifier. Valid value: A string of up to 255 characters.
port1_name	The physical port name. Valid value: A string of up to 31 characters.
switch2_id	Another Switch identifier. Valid value: A string of up to 255 characters.
port2_name	The physical port name. Valid value: A string of up to 31 characters.
description	A small description. Valid value: A string of up to 127 characters.
count	The number of Links. Valid value: A positive integer.
operstatus	The operational status. Valid value: up, down, unknown.

1.43.2. Show Physical Link

This operation is used to view a specific physical Link information.

Processing request

Method	GET
Request URI	• XML format /controllers/{controller_id}/links/{link_name}.xml • JSON format /controllers/{controller_id}/links/{link_name}.json

Table 347: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
link_name	Link name. Valid value: A string with the format {switch1_id}-.{port1_name}-.{switch2_id}.{port2_name}.	Yes

Request body	None

Processing result

Response body	• XML format

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<link link_name="{link_name}"
 switch1_id="{switch1_id}"
 port1_name="{port1_name}"
 switch2_id="{switch2_id}"
 port2_name="{port2_name}"
 description="{description}"
 operstatus="{operstatus}" />
```

• JSON format

```json
{
    "link": {
        "link_name": "{link_name}",
        "switch1_id": "{switch1_id}",
        "port1_name": "{port1_name}",
        "switch2_id": "{switch2_id}",
        "port2_name": "{port2_name}",
        "description": "{description}",
        "operstatus": "{operstatus}"
    }
}
```

Table 348: Description of response elements (port)

Element	Description
link_name	Link name. Valid value: A string with the format {switch1_id}-.{port1_name}-.{switch2_id}.{port2_name}.
switch1_id	A Switch identifier. Valid value: A string of up to 255 characters.
port1_name	The physical port name. Valid value: A string of up to 31 characters.
switch2_id	Another Switch identifier. Valid value: A string of up to 255 characters.
port2_name	The physical port name. Valid value: A string of up to 31 characters.
description	Additional information. Valid value: A string of up to 127 characters.
operstatus	The operational status. Valid value: up, down, unknown.

1.44. Physical Domain functions

This section lists the physical Domain functions.

1.44.1. Create Physical Domain

This operation is used to add a physical Domain information.

Processing request

Method	POST
Request URI	• XML format /controllers/{controller_id}/domains.xml • JSON format /controllers/{controller_id}/domains.json

Table 349: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	

Request body

- XML format

```
<domain domain_id="{domain_id}" type="{type}"
 description="{description}"></domain>
```

- JSON format

```
{
    "domain": {
        "domain_id": "{domain_id}",
        "type": "{type}",
        "description": "{description}"
    }
}
```

Table 350: Description of request elements

Element	Description	Required
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
type	Domain type. Valid value: normal	Yes
description	A small description of the Domain. Valid value: A string of up to 127 characters.	No

Remember

Physical Domain can be created only for the controller whose *type* is bypass.

Processing result

Response body   None

1.44.2. Delete Physical Domain

This operation is used to delete a physical Domain.

Processing request

Method   DELETE

Request URI

- XML format

/controllers/{controller_id}/domains/{domain_id}.xml
- JSON format

/controllers/{controller_id}/domains/{domain_id}.json

Table 351: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.	Yes
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore.	Yes

Request body   None

Remember
Physical Domain can be deleted only for the controller whose *type* is bypass.

Processing result

Response body   None

1.44.3. Update Physical Domain
This operation is used to update a physical Domain.

Processing request

Method   PUT

Request URI
- XML format

/controllers/{controller_id}/domains/{domain_id}.xml
- JSON format

/controllers/{controller_id}/domains/{domain_id}.json

Table 352: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers,	Yes

Element	Description	Required
	and underscore (except at the beginning).	

Request body

• XML format

```
<domain type="{type}"
 description="{description}"></domain>
```

• JSON format

```
{
    "domain": {
        "description": "{description}"

    }
}
```

Table 353: Description of request elements

Element	Description	Required
description	A small description of the domain. Valid value: A string of up to 127 characters.	No

Remember

Physical Domain can be updated only for the controller whose *type* is bypass.

Processing result

Response body   None

1.44.4. List Physical Domains

This operation is used to list physical Domain information.

Processing request

Method   GET

Request URI

• XML format

/controllers/{controller_id}/domains.xml

/controllers/{controller_id}/domains/detail.xml

/controllers/{controller_id}/domains/count.xml

• JSON format

/controllers/{controller_id}/domains.json

/controllers/{controller_id}/domains/detail.json

/controllers/{controller_id}/domains/count.json

• Query string

?index={domain_id}&max_repetition={max_repetition}

Table 354: Description of request URI element

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 355: Description of query string elements

Element	Description	Required
Domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
max_repetition	The number of resources that are returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <domains>
      <domain domain_id="{domain_id}" />
  </domains>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <domains>
      <domain domain_id="{domain_id}"
   type="{type}" description="{description}"
   operstatus="{operstatus}"></domain>
  </domains>
  ```

  If count is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  ```

311

```
<domains count="{count}" />
```

- JSON format

  If count is not specified in URI

```
{
    "domains": [
        {
            "domain_id": "{domain_id}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "domains": [
        {
            "domain_id": "{domain_id}",
            "type": "{type}",
            "description": "{description}",
            "operstatus": "{operstatus}"
        }
    ]
}
```

If count is specified in URI

```
{
    "domains": {
        "count": "{count}"
    }
}
```

Table 356: Description of response elements

Element	Description
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
type	Domain type . Valid value: default, normal
description	A small description of the Domain. Valid value: A string of up to 127 characters.
operstatus	The operational status. Valid value: up, down, unknown
count	The number of Domains.

Element	Description
	Valid value: A positive integer.

1.44.5. Show Physical Domain

This operation is used to view a specific physical Domain information.

Processing request

Method   GET

Request URI
- XML format

  /controllers/{controller_id}/domains/{domain_id}.xml
- JSON format

  /controllers/{controller_id}/domains/{domain_id}.json

Table 357: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No
Domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	No

Request body   None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<domain domain_id="{domain_id}"
 type="{type}" description="{description}"
 operstatus="{operstatus}" />
```

- JSON format

```
{
    "domain": {
        "domain_id": "{domain_id}",
        "type": "{type}",
        "description": "{description}",
        "operstatus": "{operstatus}"
    }
}
```

313

Table 358: Description of response elements

Element	Description
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
type	Domain type . Valid value: default, normal
description	A small description of the Domain. Valid value: A string of up to 127 characters.
operstatus	The operational status. Valid value: up, down, unknown

1.45. Logical Port functions

This section lists the Logical Port functions.

1.45.1. List Logical Ports

This operation is used to list logical Port information.

Processing request

Method	GET
Request URI	• XML format

/controllers/{controller_id}/domains/{domain_id}/logical_ports.xml

/controllers/{controller_id}/domains/{domain_id}/logical_ports/detail.xml

/controllers/{controller_id}/domains/{domain_id}/logical_ports/count.xml

• JSON format

/controllers/{controller_id}/domains/{domain_id}/logical_ports.json

/controllers/{controller_id}/domains/{domain_id}/logical_ports/detail.json

/controllers/{controller_id}/domains/{domain_id}/logical_ports/count.json

• Query string

?index={logical_port_id}&max_repetition={max_repetition}

Table 359: Description of request URI element

Element	Description	Required
controller_id	Identifier of the Controller.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 360: Description of query string elements

Element	Description	Required
logical_port_id	Identifier of the logical Port. Valid value: A string of up to 319 characters.	No
max_repetition	Number of resources that are returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

  If count is not specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <logical_ports>
      <logical_port
   logical_port_id="{logical_port_id}" />
  </logical_ports>
  ```

  If detail is specified in URI

  ```
  <?xml version="1.0" encoding="UTF-8"
   standalone="yes"?>
  <logical_ports>
      <logical_port
   logical_port_id="{logical_port_id}"
   type="{type}" description="{description}"
   switch_id="{switch_id}"
   port_name="{port_name}"
   operdown_criteria="{operdown_criteria}"
   operstatus="{operstatus}">
          <member_ports>
  ```

```
            <member_ports
 switch_id="{switch_id}" />
            <member_ports
 port_name="{port_name}" />
        </member_ports>
    </logical_port>
</logical_ports>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<logical_ports count="{count}" />
```

- JSON format

    If count is not specified in URI

```
{
    "logical_ports": [
        {
            "logical_port_id":
 "{logical_port_id}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "logical_ports": [
        {
            "logical_port_id":
 "{logical_port_id}",
            "description": "{description}",
            "type": "{type}",
            "switch_id": "{switch_id}",
            "port_name": "{port_name}",
            "operdown_criteria":
 "{operdown_criteria}",
            "operstatus": "{operstatus}",
            "member_ports": [
                {
                    "switch_id": "{switch_id}",
                    "port_name": "{port_name}"
                }
            ]
        }
    ]
}
```

If count is specified in URI

```
{
    "logical_port": {
        "count": "{count}"
    }
}
```

Table 361: Description of response elements (logicalports)

Element	Description
logical_port_id	Identifier of the logical Port. Valid value: A string of up to 319 characters.
description	Additional information. Valid value: A string of up to 127 characters.
type	Logical port type. Valid value: switch, port, trunk, subdomain, tunnel_endpoint
switch_id	Identifier of the Switch. Valid value: A string of up to 255 characters.
port_name	Port name. Valid value: A string of up to 31 characters.
operdown_criteria	Operation down criteria. Valid value: any, all
operstatus	The operational status. Valid value: down, up, unknown
count	The number of logical Port. Valid value: A positive integer.

Table 362: Description of response elements (member_ports)

Element	Description
switch_id	Identifier of the switch. Valid value: A string of up to 255 characters.
port_name	External Port name. Valid value: A string of up to 31 characters.

1.45.2. Show Logical Port

This operation is used to view a specific logical Port information.

Processing request

Method	GET
Request URI	• XML format

/controllers/{controller_id}/domains/{domain_id}/logical_ports/
{logical_port_id}.xml
- JSON format

/controllers/{controller_id}/domains/{domain_id}/logical_ports/
{logical_port_id}.json

Table 363: Description of request URI elements

Element	Description	Required
controller_id	Identifier of the Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
domain_id	Identifier of the Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning)	Yes
logical_port_id	Identifier of the logical Port. Valid value: A string of up to 319 characters.	Yes

Request body

None

Processing result

Response body
- XML format

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<logical_ports>
    <logical_port
 logical_port_id="{logical_port_id}"
 type="{type}" description="{description}"
 switch_id="{switch_id}"
 port_name="{port_name}"
 operdown_criteria="{operdown_criteria}"
 operstatus="{operstatus}">
        <member_ports>
            <member_ports
 switch_id="{switch_id}" />
            <member_ports
 port_name="{port_name}" />
        </member_ports>
    </logical_port>
</logical_ports>
```

- JSON format

```json
{
    "logical_port": {
        "logical_port_id": "{logical_port_id}",
        "description": "{description}",
```

```
        "type": "{type}",
        "switch_id": "{switch_id}",
        "port_name": "{port_name}",
        "operdown_criteria":
 "{operdown_criteria}",
        "operstatus": "{operstatus}",
        "member_ports": [
            {
                "switch_id": "{switch_id}",
                "port_name": "{port_name}"
            }
        ]
    }
}
```

Table 364: Description of response elements (logicalports)

Element	Description
logical_port_id	Identifier of the logical Port. Valid value: A string of up to 319 characters.
description	Additional information. Valid value: A string of up to 127 characters.
type	Logical port type. Valid value: switch, port, trunk, subdomain, tunnel_endpoint
switch_id	Identifier of the Switch. Valid value: A string of up to 255 characters.
port_name	Port name. Valid value: A string of up to 31 characters.
operdown_criteria	Operation down criteria. Valid value: any, all
operstatus	The operational status. Valid value: down, up, unknown

Table 365: Description of response elements (member_ports)

Element	Description
switch_id	Identifier of the switch. Valid value: A string of up to 255 characters.
port_name	External Port name.

Element	Description
	Valid value: A string of up to 31 characters.

1.46. Physical Boundary functions

This section lists the physical Boundary functions.

1.46.1. Create Physical Boundary

This operation is used to add a physical Boundary information.

Processing request

Method POST

Request URI
- XML format

  /boundaries.xml
- JSON format

  /boundaries.json

Request body
- XML format

```
<boundary boundary_id="{boundary_id}"
 description="{description}">
    <link controller1_id="{controller1_id}"
 domain1_id="{domain1_id}"
 logical_port1_id="{logical_port1_id}"
 controller2_id="{controller2_id}"
 domain2_id="{domain2_id}"
 logical_port2_id="{logical_port2_id}" />
</boundary>
```

- JSON format

```
{
    "boundary": {
        "boundary_id": "{boundary_id}",
        "description": "{description}",
        "link": {
            "controller1_id":
 "{controller1_id}",
            "domain1_id": "{domain1_id}",
            "logical_port1_id":
 "{logical_port1_id}",
            "controller2_id":
 "{controller2_id}",
            "domain2_id": "{domain2_id}",
            "logical_port2_id":
 "{logical_port2_id}"
        }
    }
}
```

Table 366: Description of request elements

Element	Description	Required
boundary_id	Identifier of the Boundary. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
description	Additional information. Valid value: A string of up to 127 characters.	No
controller1_id	Identifier of the first Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
domain1_id	Identifier of the first Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning)	Yes
logical_port1_id	Name of the first logical Port. Valid value: A string of up to 319 characters. Note: This parameter cannot be specified if *type* of controller specified in *controller1_id* is bypass.	Yes
controller2_id	Identifier of the second Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
domain2_id	Identifier of the second Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning)	Yes
logical_port2_id	Name of the second logical Port. Valid value: A string of up to 319 characters. Note: This parameter cannot be specified if *type* of controller specified in *controller2_id* is bypass.	Yes

Processing result

Response body   None

1.46.2. Delete Physical Boundary

This operation is used to delete a physical Boundary.

Processing request

Method   DELETE

Request URI
- XML format

  /boundaries/{boundary_id}.xml
- JSON format

  /boundaries/{boundary_id}.json

Table 367: Description of request URI elements

Element	Description	Required
boundary_id	Identifier of the Boundary. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body   None

Remember
Physical Boundary that is set in *boundary_id* of vLink cannot be deleted.

Processing result

Response body   None

1.46.3. Update Physical Boundary

This operation is used to update a physical Boundary.

Processing request

Method   PUT

Request URI
- XML format

  /boundaries/{boundary_id}.xml
- JSON format

  /boundaries/{boundary_id}.json

Table 368: Description of request URI elements

Element	Description	Required
boundary_id	Identifier of the Boundary.	Yes

Element	Description	Required
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	

Request body

- XML format

```
<boundary description="{description}">
</boundary>
```

- JSON format

```
{
    "boundary": {
        "description": "{description}"
    }
}
```

Table 369: Description of request elements

Element	Description	Required
description	Additional information. Valid value: A string of up to 127 characters.	No

Processing result

Response body   None

1.46.4. List Physical Boundaries

This operation is used to list physical Boundary information.

Processing request

Method   GET

Request URI

- XML format

  /boundaries.xml

  /boundaries/detail.xml

  /boundaries/count.xml

- JSON format

  /boundaries.json

  /boundaries/detail.json

  /boundaries/count.json

- Query string

  ?index={boundary_id}&max_repetition={max_repetition}

  &controller1_id={controller1_id}&controller1_id={controller1_id}

Table 370: Description of request URI element

Element	Description	Required
boundary_id	Identifier of the Boundary. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Table 371: Description of query string elements

Element	Description	Required
boundary_id	Identifier of the Boundary. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
controller1_id	Identifier of the first Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
controller2_id	Identifier of the second Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes
max_repetition	The number of resources that are returned. Valid value: A positive integer. Valid range: 1 to MAX of UINT32. Default is 10000.	No

Request body

None

Processing result

Response body

- XML format

  If detail/count are not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<boundaries>
    <boundary boundary_id="{boundary_id}" />
</boundaries>
```

If detail is specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<boundaries>
    <boundary boundary_id="{boundary_id}"
 description="{description}"
 operstatus="{operstatus}">
        <link controller1_id="{controller1_id}"
 domain1_id="{domain1_id}"
 logical_port1_id="{logical_port1_id}"
 controller2_id="{controller2_id}"
 domain2_id="{domain2_id}"
 logical_port2_id="{logical_port2_id}" />
    </boundary>
</boundaries>
```

If count is specified in URI

```xml
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<boundaries count="{count}" />
```

- JSON format

If count is not specified in URI

```json
{
    "boundaries": [
        {
            "boundary_id": "{boundary_id}"
        }
    ]
}
```

If detail is specified in URI

```json
{
    "boundaries": [
        {
            "boundary_id": "{boundary_id}",
            "description": "{description}",
            "operstatus": "{operstatus}",
            "link": {
                "controller1_id":
 "{controller1_id}",
                "domain1_id": "{domain1_id}",
                "logical_port1_id":
 "{logical_port1_id}",
                "controller2_id":
 "{controller2_id}",
                "domain2_id": "{domain2_id}",
                "logical_port2_id":
 "{logical_port2_id}"
            }
        }
    ]
}
```

325

If count is specified in URI

```
{
    "boundaries": {
        "count": "{count}"
    }
}
```

Table 372: Description of response elements (boundary)

Element	Description
boundary_id	Identifier of the Boundary. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
operstatus	The operational status. Valid value: up, down, unknown
count	The number of Domains. Valid value: A positive integer.
link	Link list.

Table 373: Description of response elements (link)

Element	Description
controller1_id	Identifier of the first Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
domain1_id	Identifier of the first Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
logical_port1_id	Name of the first logical Port. Valid value: A string of up to 319 characters.
controller2_id	Identifier of the second Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
domain2_id	Identifier of the second Domain.

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning)
logical_port2_id	Name of the second logical Port. Valid value: A string of up to 319 characters.

1.46.5. Show Physical Boundary

This operation is used to view a specific physical Boundary information.

Processing request

Method GET

Request URI
- XML format

  /boundaries/{boundary_id}.xml
- JSON format

  /boundaries/{boundary_id}.json

Table 374: Description of request URI elements

Element	Description	Required
boundary_id	Identifier of the Boundary. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).	Yes

Request body None

Processing result

Response body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<boundary boundary_id="{boundary_id}"
 description="{description}"
 operstatus="{operstatus}">
    <link controller1_id="{controller1_id}"
 domain1_id="{domain1_id}"
 logical_port1_id="{logical_port1_id}"
 controller2_id="{controller2_id}"
 domain2_id="{domain2_id}"
 logical_port2_id="{logical_port2_id}" />
</boundary>
```

- JSON format

```
{
```

```
     "boundary": {
          "boundary_id": "{boundary_id}",
          "description": "{description}",
          "operstatus": "{operstatus}",
          "link": {
               "controller1_id":
 "{controller1_id}",
               "domain1_id": "{domain1_id}",
               "logical_port1_id":
 "{logical_port1_id}",
               "controller2_id":
 "{controller2_id}",
               "domain2_id": "{domain2_id}",
               "logical_port2_id":
 "{logical_port2_id}"
          }
     }
}
```

Table 375: Description of response elements (boundary)

Element	Description
boundary_id	Identifier of the Boundary. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
description	Additional information. Valid value: A string of up to 127 characters.
operstatus	The operational status. Valid value: up, down, unknown
link	Link list.

Table 376: Description of response elements (link)

Element	Description
controller1_id	Identifier of the first Controller. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
domain1_id	Identifier of the first Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
logical_port1_id	Name of the first logical Port. Valid value: A string of up to 319 characters.
controller2_id	Identifier of the second Controller.

Element	Description
	Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning).
domain2_id	Identifier of the second Domain. Valid value: Up to 31 characters that can include alphabets, numbers, and underscore (except at the beginning)
logical_port2_id	Name of the second logical Port. Valid value: A string of up to 319 characters.

1.47. Configuration functions

This section lists the Configuration functions.

1.47.1. Clear Startup Configuration

This operation is used to clear Startup Configuration information.

Processing request

Method   PUT

Request URI
- XML format

  /configuration/startup.xml
- JSON format

  /configuration/startup.json

Request body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<startup operation="{operation}" />
```

- JSON format

```
{
    "startup": {
        "operation": "{operation}"
    }
}
```

Table 377: Description of request elements

Element	Description	Required
operation	Clear startup configuration. Valid value: clear Note: The value is case insensitive	Yes

329

Processing result

Response body   None

1.47.2. Save Configuration

This operation is used to save Configuration.

Processing request

Method   PUT

Request URI
- XML format

  /configuration.xml
- JSON format

  /configuration.json

Request body
- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<configuration operation="{operation}" />
```

- JSON format

```
{
    "configuration": {
        "operation": "{operation}"
    }
}
```

Table 378: Description of request elements

Element	Description	Required
operation	Save configuration. Valid value: save Note: The value is case insensitive	Yes

Processing result

Response body   None

1.47.3. Show Configuration Difference Status

This operation is used to find the configuration difference status.

Processing request

Method   PUT

Request URI
- XML format

  /configuration/diff.xml
- JSON format

330

/configuration/diff.json

Request body   None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<diff diff_status="{diff_status}" />
```

- JSON format

```
{
    "diff": {
        "diff_status": "{diff_status}"
    }
}
```

Table 379: Description of request elements

Element	Description
diff_status	The configuration difference status. Valid value: true, false

1.47.4. Enable or Disable Auto-save Status

This operation is used to enable or disable Auto-save status. Auto-save function stores the configurations permanently without calling Save API after every commit.

Processing request

Method   PUT

Request URI

- XML format

  /configuration/autosave.xml
- JSON format

  /configuration/autosave.json

Request body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<autosave
 auto_save_status="{auto_save_status}" />
```

- JSON format

```
{
    "autosave": {
        "auto_save_status":
 "{auto_save_status}"
    }
```

331

```
}
```

Table 380: Description of request elements

Element	Description	Required
auto_save_status	Enable or disable the Auto-save setup. Valid value: enable, disable	Yes

Processing result

Response body   None

1.47.5. Show Auto-save Status

This operation is used to show the Auto-save status.

Processing request

Method   GET

Request URI   • XML format

　　　　　　　　　/configuration/autosave.xml
　　　　　　　　• JSON format

　　　　　　　　　/configuration/autosave.json

Request body   None

Processing result

Response body   • XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<autosave
 auto_save_status="{auto_save_status}"/>
```

• JSON format

```
{
    "autosave": {
        "auto_save_status": "{auto_save_status}"
    }
}
```

Table 381: Description of response elements

Element	Description	Required
auto_save_status	Enable or disable the Auto-save setup. Valid value: enable, disable	Yes

1.48. Session functions

This section lists the Session functions.

1.48.1. List Sessions

This operation is used to list sessions information.

Processing request

Method	GET
Request URI	• XML format

    /sessions.xml

    /sessions/detail.xml

    /sessions/count.xml

    • JSON format

    /sessions.json

    /sessions/detail.json

    /sessions/count.json

Request body   None

Processing result

Response body   • XML format

If count is not specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<sessions>
    [
    <session session_id="{session_id}" />
    ]
</sessions>
```

If detail is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<sessions>
    [
    <session session_id="{session_id}"
 type="{type}" username="{username}"
 usertype="{usertype}" ipaddr="{ipaddr}"
 login_name="{login_name}"
 login_time="{login_time}" info="{info}"
 mode="{mode}" configstatus="{configstatus}" />
    ]
</sessions>
```

If count is specified in URI

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
```

```
<sessions count="{count}" />
```

- JSON format

If count is not specified in URI

```
{
    "sessions": [
        {
            "session_id": "{session_id}"
        }
    ]
}
```

If detail is specified in URI

```
{
    "sessions": [
        {
            "session_id": "{session_id}",
            "type": "{type}",
            "username": "{username}",
            "usertype": "{usertype}",
            "ipaddr": "{ipaddr}",
            "login_name": "{login_name}",
            "login_time": "{login_time}",
            "info": "{info}",
            "mode": "{mode}",
            "configstatus": "{configstatus}"
        }
    ]
}
```

If count is specified in URI

```
{
    "sessions": {
        "count": "{count}"
    }
}
```

Table 382: Description of response elements

Element	Description
session_id	Identifier of the session. Valid value: A positive integer. Valid range: 1 - UINT32_MAX
type	Session type. Valid value: webapi, webui.
username	User name for which the session information is shown. Valid value: admin, oper.
usertype	Type of user.

Element	Description
	Valid value: admin, oper.
ipaddr	IPv4 IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
loginname	The login name. Valid value: A string of up to 63 characters.
logintime	Date and time of login. Valid value: Date and time from 1970-01-01 00:00:00 to current date and time.
info	Arbitrary information which is set when session is created. Valid value: A string of up to 63 characters.
mode	Value of mode. Valid value: oper, enable, del, unknown
configstatus	The configuration status.
count	The number of sessions. Valid value: A positive integer.

1.48.2. Show Session

This operation is used to view a specific session information.

Processing request

Method   GET

Request URI

- XML format

  /sessions/{session_id}.xml
- JSON format

  /sessions/{session_id}.json

Table 383: Description of request URI elements

Element	Description	Required
session_id	Identifier of the session. Valid value: A positive integer. Valid range: 1 - UINT32_MAX	Yes

Request body   None

Processing result

Response body

- XML format

```
<?xml version="1.0" encoding="UTF-8"
 standalone="yes"?>
<session session_id="{session_id}"
 type="{type}" username="{username}"
 usertype="{usertype}" ipaddr="{ipaddr}"
 login_name="{login_name}"
 login_time="{login_time}" info="{info}"
 mode="{mode}" configstatus="{configstatus}" />
```

- JSON format

```
{
    "session": {
        "session_id": "{session_id}",
        "type": "{type}",
        "username": "{username}",
        "usertype": "{usertype}",
        "ipaddr": "{ipaddr}",
        "login_name": "{login_name}",
        "login_time": "{login_time}",
        "info": "{info}",
        "mode": "{mode}",
        "configstatus": "{configstatus}"
    }
}
```

Table 384: Description of response elements

Element	Description
session_id	Identifier of the session. Valid value: A positive integer. Valid range: 1 - UINT32_MAX
type	Session type. Valid value: webapi, webui.
username	User name for which the session information is shown. Valid value: admin, oper.
usertype	Type of user. Valid value: admin, oper.
ipaddr	IPv4 IP address. Valid value: IPv4 dot-separated format (Example: 192.168.1.1)
loginname	The login name. Valid value: A string of up to 63 characters.

Element	Description
logintime	Date and time of login. Valid value: Date and time from 1970-01-01 00:00:00 to current date and time.
info	Arbitrary information which is set when session is created. Valid value: A string of up to 63 characters.
mode	Value of mode. Valid value: oper, enable, del, unknown
configstatus	The configuration status. Valid value: disable, enable

1.49. User function

This section lists the User function.

1.49.1. Set Password

This operation is used to set user password.

Processing request

Method PUT

Request URI
- XML format

  /users/{username}/password.xml
- JSON format

  /users/{username}/password .json

Table 385: Description of request URI elements

Element	Description	Required
username	The user name for which to set the password. Valid values: admin, oper (default)	Yes

Request body
- XML format

  ```
  <password password="{password}"/>
  ```
- JSON format

  ```
  {
      "password": "{password}"
  }
  ```

Table 386: Description of request elements

Element	Description	Required
password	The new password. Valid value: A string of up to 72 characters.	Yes

Processing result

Response body　　None