THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB’s public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS” WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies are considered un-controlled copies and the original on-line version should be referred to for latest version.

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: http://www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2019 Cisco Systems, Inc. All rights reserved.
# Table of Contents

Cisco Ultra Services Platform NETCONF API Guide, Release 6.7 ................................................................. 1

Conventions ......................................................................................................................................................... 9

Obtaining Documentation and Submitting a Service Request ......................................................................... 9

Ultra Services Platform (USP) Introduction ....................................................................................................... 10

API Functions ..................................................................................................................................................... 10

UAS Roles ............................................................................................................................................................ 11

AutoIT .................................................................................................................................................................. 12

AutoVNF ............................................................................................................................................................ 12

AutoDeploy ......................................................................................................................................................... 12

UAS Architecture ............................................................................................................................................... 12

UAS Workflow .................................................................................................................................................... 13

UAS Deployment Options .................................................................................................................................. 14

UAS Cluster ......................................................................................................................................................... 14

All-in-One Mode (Single NSD) ............................................................................................................................. 14

UAS in Cluster Mode (Multiple NSDs) .................................................................................................................. 15

VNF-EM Deployment via UAS ............................................................................................................................. 16

Configuration Descriptors and Operational Records ....................................................................................... 18

Descriptors Vs Instances for VIM Resources ....................................................................................................... 18

Transaction Management .................................................................................................................................. 19

Operations ............................................................................................................................................................ 19

Viewing Transaction Status ................................................................................................................................ 19

Clearing Transactions ........................................................................................................................................ 21

Transaction Logs .................................................................................................................................................. 22

Operations ............................................................................................................................................................ 22

Viewing Transaction Logs .................................................................................................................................. 22

Clearing Transaction Logs ................................................................................................................................ 23

Viewing Transaction Hierarchy ............................................................................................................................ 24

Notification Events Management .......................................................................................................................... 25

Operations ............................................................................................................................................................ 25
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subscribing to UAS Notification Stream</td>
<td>25</td>
</tr>
<tr>
<td>Obtaining Notification Events</td>
<td>25</td>
</tr>
<tr>
<td>UAS Notification Events</td>
<td>27</td>
</tr>
<tr>
<td>Deployment Events</td>
<td>27</td>
</tr>
<tr>
<td>Lifecycle Events</td>
<td>28</td>
</tr>
<tr>
<td>Event History</td>
<td>29</td>
</tr>
<tr>
<td>Event Timestamp</td>
<td>30</td>
</tr>
<tr>
<td>NSD Configuration</td>
<td>31</td>
</tr>
<tr>
<td>Operations</td>
<td>31</td>
</tr>
<tr>
<td>Configuring NSD</td>
<td>31</td>
</tr>
<tr>
<td>NSD Deployment</td>
<td>34</td>
</tr>
<tr>
<td>Operations</td>
<td>34</td>
</tr>
<tr>
<td>Activating NSD</td>
<td>34</td>
</tr>
<tr>
<td>Deactivating NSD</td>
<td>34</td>
</tr>
<tr>
<td>Uploading NSD</td>
<td>35</td>
</tr>
<tr>
<td>Updating NSD</td>
<td>36</td>
</tr>
<tr>
<td>Obtaining Deployment Notification During NSD Activation</td>
<td>38</td>
</tr>
<tr>
<td>Fetching VNFR Information</td>
<td>40</td>
</tr>
<tr>
<td>Network Service Record Parameters</td>
<td>50</td>
</tr>
<tr>
<td>Virtual Link Record Parameters</td>
<td>51</td>
</tr>
<tr>
<td>Secure Token Configuration</td>
<td>53</td>
</tr>
<tr>
<td>Operations</td>
<td>53</td>
</tr>
<tr>
<td>Configuring Secure Token</td>
<td>53</td>
</tr>
<tr>
<td>Configuring SSH Keys</td>
<td>54</td>
</tr>
<tr>
<td>Configuring Secure SCM</td>
<td>55</td>
</tr>
<tr>
<td>VIM Orchestrator Configuration</td>
<td>58</td>
</tr>
<tr>
<td>Operations</td>
<td>58</td>
</tr>
<tr>
<td>Configuring VIM- ORCHD</td>
<td>58</td>
</tr>
<tr>
<td>VIM Orchestrator Deployment</td>
<td>65</td>
</tr>
<tr>
<td>Operations</td>
<td>65</td>
</tr>
<tr>
<td>Deploying VIM Orchestrator</td>
<td>65</td>
</tr>
</tbody>
</table>
Obtaining VIM- ORCH Deployment Event Notifications .................................................. 65
Obtaining VIM- ORCH Instance Information .............................................................. 68
Deactivating VIM Orchestrator .................................................................................... 69
Obtaining Notifications Sent During Deactivation of VIM- ORCH Deployment .......... 69
VIM Configuration ....................................................................................................... 72
Operations ................................................................................................................... 72
Configuring VIMD ........................................................................................................ 72
VIM Deployment ......................................................................................................... 78
Operations ................................................................................................................... 78
Deploying VIM ............................................................................................................. 78
Obtaining VIM Deployment Event Notifications ....................................................... 78
Obtaining VIM Instance Information .......................................................................... 81
Obtaining NFVI Node Instance Information ............................................................. 82
Deactivating VIM ......................................................................................................... 83
Obtaining Notifications Sent During Deactivation of VIM Deployment ....................... 84
VNF Package Configuration ......................................................................................... 87
Operations ................................................................................................................... 88
Configuring VNF Package .......................................................................................... 88
VNF Package Deployment ............................................................................................ 90
Operations ................................................................................................................... 90
Activating VNF Package .............................................................................................. 90
Obtaining VNF Package Deployment Event Notifications ......................................... 91
Obtaining VNF Package Operational Instance Information ..................................... 92
Deactivating VNF Package ........................................................................................... 95
Obtaining Notifications Sent During Deactivation of VNF Package ............................ 95
Virtual Link Configuration ........................................................................................... 98
Virtual Link Descriptor Configuration Parameters ................................................... 98
Operations ................................................................................................................... 99
Configuring Network Descriptor ................................................................................. 99
Configuring Network Instance ..................................................................................... 101
VIM Identity Configuration ......................................................................................... 103
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>103</td>
</tr>
<tr>
<td>Configuring VIM Identity</td>
<td>103</td>
</tr>
<tr>
<td>VIM Artifacts Configuration</td>
<td>105</td>
</tr>
<tr>
<td>Operations</td>
<td>105</td>
</tr>
<tr>
<td>Configuring VIM Artifacts</td>
<td>105</td>
</tr>
<tr>
<td>Configuring VNF Rack</td>
<td>108</td>
</tr>
<tr>
<td>Defining a Tenant</td>
<td>109</td>
</tr>
<tr>
<td>VIM Artifacts Deployment</td>
<td>113</td>
</tr>
<tr>
<td>Operations</td>
<td>113</td>
</tr>
<tr>
<td>Activating VNF Rack</td>
<td>113</td>
</tr>
<tr>
<td>Obtaining Notifications During VNF Rack Activation</td>
<td>113</td>
</tr>
<tr>
<td>Obtaining VNF Rack Operational Instance Information</td>
<td>115</td>
</tr>
<tr>
<td>Deactivating VNF Rack</td>
<td>116</td>
</tr>
<tr>
<td>Obtaining Notifications During VNF Rack Deactivation</td>
<td>116</td>
</tr>
<tr>
<td>Activating a Tenant</td>
<td>118</td>
</tr>
<tr>
<td>Obtaining Notifications During Tenant Activation</td>
<td>118</td>
</tr>
<tr>
<td>Deactivating a Tenant</td>
<td>120</td>
</tr>
<tr>
<td>Obtaining Notifications During Tenant Deactivation</td>
<td>120</td>
</tr>
<tr>
<td>VNF Descriptor Configuration</td>
<td>123</td>
</tr>
<tr>
<td>VNFD Configuration Parameters</td>
<td>123</td>
</tr>
<tr>
<td>Cloud-Info Descriptor Parameters</td>
<td>124</td>
</tr>
<tr>
<td>VNFM Descriptor Parameters</td>
<td>125</td>
</tr>
<tr>
<td>VNF Instance Descriptor Parameters</td>
<td>125</td>
</tr>
<tr>
<td>VNF Configuration Parameters</td>
<td>126</td>
</tr>
<tr>
<td>Syslog Descriptor Parameters</td>
<td>126</td>
</tr>
<tr>
<td>VNF External Connection Points Parameters</td>
<td>127</td>
</tr>
<tr>
<td>Floating-IP Descriptor Parameters</td>
<td>127</td>
</tr>
<tr>
<td>Health Check Descriptor Parameters</td>
<td>127</td>
</tr>
<tr>
<td>VNFC Descriptor Parameters</td>
<td>128</td>
</tr>
<tr>
<td>VNFC Initialization Variable Descriptor Parameters</td>
<td>129</td>
</tr>
<tr>
<td>Variable Descriptor Parameters</td>
<td>129</td>
</tr>
</tbody>
</table>
VNFC Connection Point Aggregation Descriptor Parameters .......................................... 129
VNFC Connection Point Descriptor Parameters ........................................................... 130
VNFC Connection Point VLAN Descriptor Parameters .................................................. 130
VNFC Connection Point Allowed Address Pair Descriptor Parameters .......................... 130
VNFC Virtual Link Descriptor Parameters .................................................................... 131
VNFC Volume Instance Group Parameters ..................................................................... 131
VNFC-VDU Descriptor Parameters ................................................................................ 131
VNFC Syslog Descriptor Parameters ............................................................................ 132
VDU Descriptor Parameters ............................................................................................ 133
  VDU Image Descriptor Parameters ............................................................................. 135
  VDU VNF-Package Parameters .................................................................................. 135
  VDU Volume Descriptor Parameters ........................................................................... 135
  VDU Configuration Descriptor Parameters .................................................................. 136
NED Descriptor Parameters ............................................................................................ 137
UPP Descriptor Parameters ............................................................................................ 137
  VDU Flavor Descriptor Parameters ............................................................................ 137
USF Descriptor Parameters ............................................................................................. 140
  USF Host Descriptor Parameters ................................................................................ 141
  USD Network Forwarding Path (NFP) Descriptor Parameters ........................................ 142
  USF Service Function Chain Descriptor Parameters .................................................... 142
Fault Management Descriptor Configuration .................................................................... 143
  Monitoring Services .................................................................................................... 143
  Log Collection and Aggregation .................................................................................. 144
Fault Management Operational Data ................................................................................ 145
  Domain Summary Descriptor Parameters .................................................................... 145
  Fault Entry Descriptor Parameters ............................................................................. 146
  VIM Service Status Descriptor Parameters .................................................................. 147
  NFVI Node Status Descriptor Parameters ................................................................. 147
  Network Service Status Descriptor Parameters ......................................................... 147
  VNFD Status Descriptor Parameters .......................................................................... 148
  VNFC Status Descriptor Parameters .......................................................................... 148
Fault Notification Descriptor Parameters ................................................................. 148
Operations .................................................................................................................... 150
Configuring FMD ........................................................................................................ 150
Fault Management Descriptor Deployment ............................................................. 159
Operations .................................................................................................................... 159
Activating FMD .......................................................................................................... 159
Obtaining Notifications Generated During FMD Activation ......................................... 159
Conventions

This document uses the following conventions.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bold font</strong></td>
<td>Commands and keywords and user-entered text appear in <strong>bold font</strong>.</td>
</tr>
<tr>
<td><strong>italic font</strong></td>
<td>Document titles, new or emphasized terms, and arguments for which you supply values are in italic font.</td>
</tr>
<tr>
<td>[   ]</td>
<td>Elements in square brackets are optional.</td>
</tr>
<tr>
<td>{x</td>
<td>y</td>
</tr>
<tr>
<td>[ x</td>
<td>y</td>
</tr>
<tr>
<td>string</td>
<td>A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.</td>
</tr>
<tr>
<td><strong>courier font</strong></td>
<td>Terminal sessions and information the system displays appear in <strong>courier font</strong>.</td>
</tr>
<tr>
<td>&lt; &gt;</td>
<td>Nonprinting characters such as passwords are in angle brackets.</td>
</tr>
<tr>
<td>[   ]</td>
<td>Default responses to system prompts are in square brackets.</td>
</tr>
<tr>
<td>!, #</td>
<td>An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.</td>
</tr>
</tbody>
</table>

**Note:** Means reader take note. Notes contain helpful suggestions or references to material not covered in the manual.

**Caution:** Means reader be careful. In this situation, you might perform an action that could result in equipment damage or loss of data.

**Warning:** IMPORTANT SAFETY INSTRUCTIONS

Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

**Regulatory:** Provided for additional information and to comply with regulatory and customer requirements.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see What's New in Cisco Product Documentation at: http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html.

Subscribe to What's New in Cisco Product Documentation, which lists all new and revised Cisco technical documentation, as an RSS feed and deliver content directly to your desktop using a reader application. The RSS feeds are a free service.
Ultra Services Platform (USP) Introduction

Ultra Automation Services (UAS) implement Tail-F ConfD based API to provide consistency across the modules. REST/NETCONF APIs are provided for each functional block. The UAS also provides CLI to configure the descriptors.

As an UEM component, the Life Cycle Manager (LCM) includes middleware APIs that support the interface with Service Level Agreement Manager (SLA-M). The APIs are used to monitor KPIs pertaining to VNFC health and VM resource usage (for example, CPU, memory, etc.). APIs that support VNFC configuration establish interfaces to the CF via both the Management and High-availability buses to:

- Provision VMs based on information contained in virtualization descriptor units (VDUs) within the VNFD and associate the VMs to the internal network
- Add and initialize VMs as needed
- Request VNF infrastructure characteristics (for example, topology, deployment policies, etc.)
- Request VNF termination, migration, or destruction
- Request Day-N configuration for a specific VNFC
- Create and associate network ports to VDUs
- Provision networking configurations
- Provide life-cycle event notifications such as service status, configuration status, and HA events
- Provide an interface for determining NFVI information associated with the VDUs

API Functions

The UAS provides a set of API services to automate the workflow. The UAS modules are built using Cisco Tail-F technologies and the descriptor/records are modeled using YANG semantics.

The following table provides a high-level summary of the automation workflow items. For more details, see the deployment guide.
### Ultra Services Platform (USP) Introduction

#### NFVI Automation
- UCS Server Management
- Inventory Management
- Undercloud Orchestration

#### VIM Automation
- Overcloud Bring-up (RH OSP)
- Automation of VIM Provisioning (Networking, storage and security, etc.)

#### Software Management Automation
- VNF Package Management
- Automated Software Patching
- Automated Software Upgrade

#### Network Service | VNF Automation
- Network Service based deployment workflow automation to manage Ultra gateway and Policy Services.
- Workflow automation for Ultra Service Framework (USF).
- Ultra VNF Bootstrapping to pass VNF specific day-zero configurations.
- VNF Cluster management, setup, monitoring, healing and failovers.
- API endpoint management for VNFs
- VNF External and Internal networking
- Service and configuration templating

#### Fault Management
Health monitoring for:
- NFVI (undercloud, UCS servers)
- VIM (OpenStack)
- VNF
- Logs Management
- SNMP / NETCONF based notifications.
- KPIs to monitor resources
- Service level inventory to track faults and events

---

### UAS Roles

The UAS is a common software image to provide automation services. Depending on the day-zero configuration, UAS acts as one of the following roles:

- AutoIT
- AutoVNF

---

11
AutoDeploy

In this role, UAS provides a set of APIs to automate the workflow to manage NFVI resources, undercloud and overcloud (OpenStack). The set of functions include day-zero deployment orchestration, health monitoring and automated software management for these modules.

AutoVNF

AutoVNF provides Network Service based deployment workflow automation for Cisco Ultra Gateway and CPS platforms. This includes onboarding of VNF resources, generation of VNFD and monitoring of VNF/VNFM.

AutoDeploy

AutoDeploy provides workflow consolidation and resource/KPIs aggregation to manage multiple sites, where each site can be visualized as one VIM. But in terms of implementation, it is defined as nested NSDs. AutoDeploy allows you to orchestrate, monitor several VNFs using single pane of glass. Ultra-Web Services (UWS) integrates with AutoDeploy and provides UI driven workflow automation for multiple sites.

The following properties are applicable to these UAS roles:

- AutoDeploy only acts on nested NSDs. It does not activate a NSD which is not nested. None of the other modules will act on nested NSD. In other words, you cannot activate nested NSD on AutoVNF or AutoIT node.
- AutoIT is the only node, which allows activation of Fault Management, VIM-ORCH and VIM deployments. It also supports instantiating of VNFDs.

UAS Architecture

The following figure depicts the high-level architecture of UAS and some of its components. The UAS backend is based on ConfD that implements all API endpoints including REST/NETCONF/SNMP. The UAS is a multiprocessing system where some jobs are short lived (NSD/Deployment worker) and others are permanent.

UAS has internal process level heartbeat where each process punches the heartbeat every 30 seconds. UAS operational data includes information about running processes and last heartbeat timestamp.
UAS Workflow

The following figure shows a typical workflow of UAS.
Ultra Services Platform (USP) Introduction

UAS Deployment Options

The UAS in AutoDeploy is used to manage multiple sites where each site is mapped to VIM. The UAS can be deployed in different modes at site level. The “site” here is referred to single “VIM”.

UAS Cluster

The UAS runs in cluster mode to provide redundancy using VRRP as protocol. There are two VMs of UAS for each cluster.

All-in-One Mode (Single NSD)

This is useful when the UAS is used to manage a complete site for smaller deployment. One UAS cluster is provided for full automation and orchestration.

One set of infrastructure descriptor is available to orchestrate VIM, VIM-Orchestrator, Fault Management, and an array of VNFDs.
In this mode, the same UAS deploys both VIM and VIM orchestrator and the vim-identity descriptor is not required as it is within the operational record of VIM. In this mode, one VNFM can be used to deploy multiple VNFs.

It is possible to have N:M combination in Ultra VNFs to VNFM ands and not only 1:N. So, you can have two ESCs managing different sets of VNFs, similar to the depiction in the following figure:

**UAS in Cluster Mode (Multiple NSDs)**

Another deployment option is to run UAS in cluster mode to achieve better resiliency. In this mode, one UAS cluster failure does not impact the other UAS.
VNF-EM Deployment via UAS

The UAS allows VNF-EM to be deployed in two ways and hence there are two classes of Ultra VNFs defined.

**VNF-EM as VNFC:**
In this mode, VNF-EM is deployed as VNFC and part of VNF where other VNFC for VNF is defined. This is the case of UGP and UCPS type of VNF.

**VNF-EM as VNF:**

In this mode, the VNF-EM works as a separate VNF, that is, you can use the same VNF-EM to deploy multiple VNFs. This is the case for UGP-standalone and UCPS-standalone types of VNF.
Configuration Descriptors and Operational Records

The descriptors and records terminologies are from ETSI but the UAS does not implement all required interfaces for a generic NFVO.

Descriptors Vs Instances for VIM Resources

The UAS works in two modes:

- It can communicate to VIM and allocate and provision required resources to on board network services.
- You can provision and allocate resources on VIM out-of-band and create resource “instance” on UAS.

When you need a resource from VIM such as network, image, volume, flavor, tenant, you can either define a descriptor in use in NSD/VLD/VNFD or refer resource instance. The UAS either creates resources based on descriptor or uses pre-created resource.

With pre-created resources, the UAS performs validations for some of the resources before instantiating actual deployment. This is applicable only when the UAS has access to VIM. For VNFM-based deployment, the UAS initially checks the required resources from VNFM and allocates them if needed.
Transaction Management

This section provides information on how to get the transaction status and clear a specific or all transaction(s).

When an RPC is invoked to instantiate a configuration descriptor, unique transaction ID is allocated to the request and the same is used to track the progress or status. The UAS logs information messages to these transactions and the same can be fetched using the transaction ID. These transaction data is persistent hence can be fetched anytime, for example, even after the transaction is completed. APIs and commands are provided to clear the transactions on need basis.

Operations

Viewing Transaction Status

Operation: get

Namespace: transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction"

Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tx-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to specific request.</td>
</tr>
<tr>
<td>tx-type</td>
<td>M</td>
<td>choice string</td>
<td>Transaction type. See the Transaction Types for more details.</td>
</tr>
<tr>
<td>deployment-id</td>
<td>M</td>
<td>string</td>
<td>Deployment descriptor ID or configuration descriptor ID, provided as part of RPC input.</td>
</tr>
<tr>
<td>timestamp</td>
<td>M</td>
<td>yang:date-and-time</td>
<td>Date and time this transaction was executed.</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current status of the transaction (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>status-detail</td>
<td>O</td>
<td>string</td>
<td>Additional log when available.</td>
</tr>
</tbody>
</table>

Transaction Types:

<table>
<thead>
<tr>
<th>Transaction Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>activate-vim-orch-deployment</td>
<td>Activate VIM orchestrator Deployment</td>
</tr>
<tr>
<td>deactivate-vim-orch-deployment</td>
<td>Deactivate VIM Orchestrator Deployment</td>
</tr>
<tr>
<td>activate-vim-deployment</td>
<td>Activate VIM deployment</td>
</tr>
<tr>
<td>deactivate-vim-deployment</td>
<td>Deactivate VIM deployment</td>
</tr>
<tr>
<td>Transaction Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>activate-tenant-deployment</td>
<td>Set tenant parameters</td>
</tr>
<tr>
<td>deactivate-tenant-deployment</td>
<td>Remove tenant</td>
</tr>
<tr>
<td>activate-vnf-rack</td>
<td>Activate VNF rack</td>
</tr>
<tr>
<td>deactivate-vnf-rack</td>
<td>Deactivate VNF rack</td>
</tr>
<tr>
<td>activate-vnf-package</td>
<td>Onboard VNF Package</td>
</tr>
<tr>
<td>deactivate-vnf-package</td>
<td>Destroy VNF Package</td>
</tr>
<tr>
<td>activate-router-deployment</td>
<td>Create Router on VIM</td>
</tr>
<tr>
<td>deactivate-router-deployment</td>
<td>Remove Router from VIM</td>
</tr>
<tr>
<td>activate-vnf-deployment</td>
<td>Activate VNF Deployment</td>
</tr>
<tr>
<td>deactivate-vnf-deployment</td>
<td>Deactivate VNF Deployment</td>
</tr>
<tr>
<td>activate-ns-deployment</td>
<td>Activate Network Service Deployment</td>
</tr>
<tr>
<td>deactivate-ns-deployment</td>
<td>Deactivate Network Service Deployment</td>
</tr>
<tr>
<td>activate-fmd</td>
<td>Activate Fault Management</td>
</tr>
<tr>
<td>deactivate-fmd</td>
<td>Deactivate Fault Management</td>
</tr>
</tbody>
</table>

Specific Transaction Status

Example RPC
```xml
  <nc:get>
    <nc:filter type="subtree">
      <transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
        <tx-id>1509753321-443353</tx-id>
      </transaction>
    </nc:filter>
  </nc:get>
</nc:rpc>
```

Example Output
```xml
  <data>
    <transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
      <tx-id>1509753321-443353</tx-id>
      <tx-type>activate-ns-deployment</tx-type>
      <deployment-id>sjc-autoit</deployment-id>
      <timestamp>2017-11-03T23:55:21.443375-00:00</timestamp>
      <status>success</status>
    </transaction>
  </data>
</rpc-reply>
```
Clearing Transactions

Clear operation cleans up log and associated transactional status and not the actual deployment.

**Operation:** uspt:clear

**Namespace:** uspt:clear xmlns:uspt="http://www.cisco.com/usp/nfv/uspt-transaction"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>choice string</td>
<td>For a specific transaction, this is a unique transaction ID assigned to specific request. Use “all” for all transactions.</td>
</tr>
</tbody>
</table>

Specific Transaction

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:784c3bd7-449d-46b3-a4ed-d49ce2dbd6fd" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <uspt:transaction-id>1509753321-443353</uspt:transaction-id>
  </uspt:clear>
</nc:rpc>
```

**Example Output**

```xml
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:784c3bd7-449d-46b3-a4ed-d49ce2dbd6fd">
  <status xmlns="http://www.cisco.com/usp/nfv/uspt-transaction">SUCCESS</status>
</rpc-reply>
```

Clear All Transactions

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:0e3bb29f-b87f-8377-0616918e4adb" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <uspt:transaction-id>all</uspt:transaction-id>
  </uspt:clear>
</nc:rpc>
```

**Example Output**

```xml
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:0e3bb29f-b87f-8377-0616918e4adb">
  <status xmlns="http://www.cisco.com/usp/nfv/uspt-transaction">SUCCESS</status>
</rpc-reply>
```
Transaction Logs

The UAS maintains logs for each transaction that helps in getting details about the progress and errors, if any. These logs are associated with the transactions so you can easily filter the logs for a specific transaction. Further, these logs are stored on HA enabled persistent storage. Hence, you can pull these logs anytime.

Operations

Viewing Transaction Logs

Operation: get

Namespace: transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction"

Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tx-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to specific request.</td>
</tr>
</tbody>
</table>

Specific Transaction Log

Example RPC

```xml
<nc:rpc message-id="urn:uuid:ab3eab19-5d26-4565-bc03-6effb4db7fd4" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:get>
    <nc:filter type="subtree">
      <log xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
        <tx-id>1509753321-443353</tx-id>
      </log>
    </nc:filter>
  </nc:get>
</nc:rpc>
```

Example Output

```xml
  <data>
    <log xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
      <log>
        2017-11-03 23:55:21,531 - Deployment activate-ns-deployment: sjc-autoit started
        2017-11-03 23:55:21,541 - Package Activation started for usp_5_7
        2017-11-03 23:56:07,461 - ISO Downloaded Successfully
        2017-11-03 23:56:07,521 - Verifying checksum of the package
        2017-11-03 23:56:29,448 - Installing GPG key '/opt/cisco/usp/uas-staging/1509753321-443353/iso_mnt/repo/dev.gpg'
        2017-11-03 23:56:29,563 - Installing ISO
    </log>
  </data>
</rpc-reply>
```
2017-11-03 23:58:36,316 - ISO Installed successfully
2017-11-03 23:58:36,790 - Uploading Image: /opt/cisco/usb/bundles/em-bundle/em-5_7_0_1481.qcow2 to VIM as vim1-5.7.M0-2045-em
2017-11-03 23:58:54,762 - Uploading Image: /opt/cisco/usb/bundles/uas-bundle/usp-uas-5.7.0_1607.qcow2 to VIM as vim1-5.7.M0-2045-uas
2017-11-03 23:59:05,499 - Uploading Image: /opt/cisco/usb/bundles/vnfm-bundle/ESC-3_0_1_9.qcow2 to VIM as vim1-5.7.M0-2045-esc
2017-11-03 23:59:29,586 - All Images uploaded successfully
2017-11-03 23:59:29,630 - Package Activation completed successfully for usp_5_7
2017-11-03 23:59:29,634 - Uninstalling ISO
2017-11-03 23:59:43,212 - Unmounting the ISO
2017-11-03 23:59:43,260 - Send Deployment notification for: sjc-autoit</log>
</log>
</data>
</rpc-reply>

Clearing Transaction Logs

Clear operation cleans up log and the associated transaction.

Specific Transaction

Example RPC

<?xml version="1.0" encoding="UTF-8"?><nc:rpc message-id="urn:uuid:0e56c9a9-dff0-4d1b-84a3-911676" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <uspt:transaction-id>1516898683-491288/1516898683-911676</uspt:transaction-id>
  </uspt:clear>
</nc:rpc>

Example Output

<?xml version="1.0" encoding="UTF-8"?><rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:0e56c9a9-dff0-4d1b-84a3-911676"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <status xmlns="http://www.cisco.com/usp/nfv/usp-transaction">SUCCESS</status>
</rpc-reply>

Clear All Transaction Logs

Example RPC

    <uspt:transaction-id>1516898683-491288/1516898683-911676</uspt:transaction-id>
  </uspt:clear>
</nc:rpc>

Example Output

<?xml version="1.0" encoding="UTF-8"?><rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:6d22ef07-39ee-4191-bd27-c94b4eac0c3"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <status xmlns="http://www.cisco.com/usp/nfv/usp-transaction">SUCCESS</status>
</rpc-reply>
Viewing Transaction Hierarchy

A network service-level transaction can spawn in several other transactions as the case for AutoDeploy. Transaction IDs are created in hierarchical manner separated by “/” hence the top-most ID represents the service-level transaction ID.

When you fetch the logs or clear the transaction logs, it acts on the complete tree starting the transaction ID as root.

Example Configuration

```xml
<config xmlns="http://tail-f.com/ns/config/1.0">
  <transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
    <tx-id>1511907644-746851</tx-id>
    <tx-type>activate-ns-deployment</tx-type>
    <deployment-id>fremont</deployment-id>
    <timestamp>2017-11-28T22:20:44.74687-00:00</timestamp>
    <status>in-progress</status>
  </transaction>

  <transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
    <tx-id>1511907644-746851/1511907644-977631</tx-id>
    <tx-type>activate-vnf-rack</tx-type>
    <deployment-id>vnf-rack1</deployment-id>
    <timestamp>2017-11-28T22:20:44.977656-00:00</timestamp>
    <status>success</status>
  </transaction>

  <transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
    <tx-id>1511907644-746851/1511907645-034804</tx-id>
    <tx-type>activate-vnf-package</tx-type>
    <deployment-id>usp_5_7</deployment-id>
    <timestamp>2017-11-28T22:20:45.034814-00:00</timestamp>
    <status>in-progress</status>
  </transaction>

  <transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
    <tx-id>1511907644-746851/1511907645-067965</tx-id>
    <tx-type>activate-vnf-deployment</tx-type>
    <deployment-id>esc</deployment-id>
    <timestamp>2017-11-28T22:20:45.067985-00:00</timestamp>
    <status>requested</status>
  </transaction>

  <transaction xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
    <tx-id>1511907644-746851/1511907645-112822</tx-id>
    <tx-type>activate-vnf-deployment</tx-type>
    <deployment-id>vpc</deployment-id>
    <timestamp>2017-11-28T22:20:45.112833-00:00</timestamp>
    <status>requested</status>
  </transaction>
</config>
```
Notification Events Management

The UAS module supports NETCONF-based notifications that provide an API agent to be notified about an event in real time. For information on the different events sent to the UAS notification stream, see the UAS Notification Events.

Operations

Subscribing to UAS Notification Stream

**Operation:** `ncn:create-subscription`

**Namespace:** `xmlns:ncn="urn:ietf:params:xml:ns:netconf:notification:1.0"`

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ncn:stream</code></td>
<td>M</td>
<td>string</td>
<td>UAS notification stream</td>
</tr>
</tbody>
</table>

**Example RPC**

```
<nc:rpc message-id="urn:uuid:0cde0ae8-c496-4224-be94-d8a447b918c7"
         xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
      <ncn:stream>uas_notify</ncn:stream>
   </ncn:create-subscription>
</nc:rpc>
```

**Example Output**

```
   <ok/>
</rpc-reply>
```

Obtaining Notification Events

**Operation:** `get`

**Namespace:** `event xmlns="http://www.cisco.com/usp/nfv/usp-transaction"`

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>timestamp</td>
<td>M</td>
<td>yang:date-and-time</td>
<td>Date and time this notification event was triggered.</td>
</tr>
<tr>
<td>deployment-id</td>
<td>M</td>
<td>reference</td>
<td>Deployment descriptor ID.</td>
</tr>
</tbody>
</table>

Cisco Systems, Inc.  www.cisco.com
## Notification Events Management

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tx-type</td>
<td>M</td>
<td>choice string</td>
<td>Transaction type string. See the Transaction Types for more details.</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Operational status associated with this event.</td>
</tr>
</tbody>
</table>

### Example RPC

```xml
  <nc:get>
    <nc:filter type="subtree">
      <event xmlns="http://www.cisco.com/usp/nfv/usp-transaction"/>
    </nc:filter>
  </nc:get>
</nc:rpc>
```

### Example Output

```xml
  <data>
    <event xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
      <tx-id>1511907644-746851</tx-id>
      <timestamp>2017-11-28T22:20:44.774036-00:00</timestamp>
      <deployment-id>fremont</deployment-id>
      <tx-type>activate-ns-deployment</tx-type>
      <status>requested</status>
    </event>
    <event xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
      <tx-id>151907644-746851</tx-id>
      <deployment-id>vnf-rack1</deployment-id>
      <tx-type>activate-vnf-rack</tx-type>
      <status>requested</status>
    </event>
    ... (more events) ...
    <event xmlns="http://www.cisco.com/usp/nfv/usp-transaction">
      <tx-id>15128722:20:45.087105-00:00</tx-id>
      <deployment-id>vnf-package</deployment-id>
      <tx-type>activate-vnf-package</tx-type>
      <status>requested</status>
    </event>
  </data>
</rpc-reply>
```
UAS Notification Events

The UAS module supports NETCONF-based notifications that provide an API agent to be notified about an event in real time.

There are two types of events sent to the UAS notification stream:

- Deployment Events
- Lifecycle Events

Deployment Events

Deployment events are the NETCONF notifications the UAS generates as it progresses through the different stages of deployment.

- Request: This event is generated when the API backend is scheduled to handle the RPC request.
- In-progress: This event is generated when the API backend starts processing the RPC requested.
- Success: Request has been processed successfully
- Error: An error occurred while processing the request. You can pull the transaction log for additional details. Now the deployment is in error state and you must deactivate before activating it again.

The deployment event includes the following parameters. Mandatory parameters are always present and optional only when available.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
</tbody>
</table>
## Parameter Name | Required | Type     | Description                                                                 |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice string</td>
<td>Type of operation (same as type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

### Deployment Event Notifications:

<table>
<thead>
<tr>
<th>Notification ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-package-deployment-event</td>
<td>VNF package deployment.</td>
</tr>
<tr>
<td>ns-deployment-event</td>
<td>Network service deployment.</td>
</tr>
<tr>
<td>vnf-deployment-event</td>
<td>VNF deployment event.</td>
</tr>
<tr>
<td>vim-deployment-event</td>
<td>VIM deployment event.</td>
</tr>
<tr>
<td>vim-orchestrator-deployment-event</td>
<td>VIM orchestrator deployment event.</td>
</tr>
<tr>
<td>fmd-deployment-event</td>
<td>Fault management deployment event.</td>
</tr>
</tbody>
</table>

All these types of deployments have “request”, “in-progress”, “error”, and “success” events as discussed in the previous section.

### Lifecycle Events

These events are generated to notify a change in the operational state of running instance. The following parameters are included:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>reference</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>vnfd-id</td>
<td>M</td>
<td>reference</td>
<td>Reference to VNF Descriptor.</td>
</tr>
</tbody>
</table>
### Notification Events Management

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnfc-instance-id</td>
<td>O</td>
<td>reference</td>
<td>Reference to VNFC Instance ID. This is available only if this lifecycle event belongs to a VNFC Instance instead of VNF.</td>
</tr>
<tr>
<td>event</td>
<td>M</td>
<td>choice string</td>
<td>Type of change as defined in the following list:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• starting - VNF/VNFC Instance is booting.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• started - VNF/VNFC Instance has been deployed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• online - VNF/VNFC Instance is alive as defined by health monitoring.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• offline - VNF/VNFC instance has been undeployed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• failed - VNF/VNFC instance failed to come alive.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• shutting-down - VNF/VNFC Instance is shutting down.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• restarting - VNF/VNFC instance is restarting, mostly during a recovery event.</td>
</tr>
</tbody>
</table>

This lifecycle event is sent to notify VNF level changes and VNFC level changes. For example, VNF might completely fail, come online as a service, or a specific VNFC within VNF might trigger a lifecycle event such as one of the VMs within the cluster failed.

Following are the lifecycle events defined for different domains.

<table>
<thead>
<tr>
<th>Notification ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>uas-lifecycle-event</td>
<td>UAS lifecycle event</td>
</tr>
<tr>
<td>vnf-em-lifecycle-event</td>
<td>VNF-EM lifecycle event</td>
</tr>
<tr>
<td>vnfm-lifecycle-event</td>
<td>VNFM lifecycle event</td>
</tr>
<tr>
<td>ugp-lifecycle-event</td>
<td>Ultra Gateway Platform event</td>
</tr>
<tr>
<td>vim-lifecycle-event</td>
<td>Lifecycle event at VIM level</td>
</tr>
<tr>
<td>vim-orchestrator-lifecycle-event</td>
<td>Lifecycle event at VIM-orchestrator level</td>
</tr>
</tbody>
</table>

### Event History

The UAS maintains a list of events on the basis of transactions.
Notification Events Management

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tx-id</td>
<td>M</td>
<td>string</td>
<td>Transaction ID these notification events are generated</td>
</tr>
<tr>
<td>timestamp</td>
<td>M</td>
<td>array of event-timestamp</td>
<td>One or more events. See the following section for more details.</td>
</tr>
</tbody>
</table>

**Event Timestamp**

This is the actual event identified uniquely by timestamp. Following are the other parameters defined as part of this event.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>timestamp</td>
<td>M</td>
<td>yang:date-and-time</td>
<td>Date and time this notification event was triggered.</td>
</tr>
<tr>
<td>deployment-id</td>
<td>M</td>
<td>reference</td>
<td>Deployment descriptor ID.</td>
</tr>
<tr>
<td>tx-type</td>
<td>M</td>
<td>choice string</td>
<td>Transaction type string. See the <a href="#">Transaction types</a> for more details.</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Operational status associated with this event.</td>
</tr>
</tbody>
</table>
NSD Configuration

NSD is the highest level container to group the configuration descriptors. Nested NSD is a mechanism used to group the NSDs. NSD attributes are descriptors which are grouped to logically create a service model. The UAS acts on the descriptors in the order you have defined within the NSD.

Depending on the use case, you should provide input for different parameters. For example, one NSD might be used to deploy undercloud (VIM-ORCH) / overcloud VIM and the other NSD might be used to deploy gateway service. In later case, the NSD will not have VIM or VIM-ORCH.

For example, to manage multiple sites (as the case of AutoDeploy) as a single logical entity, you should define an NSD at regional level which will internally have multiple NSDs to define services at each site level. If we map each site to a VIM or NFVI-PoP, each NSD at site level has one VIM/VIM-ORCH/FMD and so on.

Operations

Configuring NSD

Operation: edit-config

Namespace: nsd xmlns="http://www.cisco.com/usp/nfv/usps-n/sdk"

Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd-id</td>
<td>M</td>
<td>string (1-16)</td>
<td>NSD identifier</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>NSD version</td>
</tr>
<tr>
<td>vim-orch</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM ORCH descriptor.</td>
</tr>
<tr>
<td>vim-orch-identity</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM descriptor used to provide vim-orchestrator identity for deployment.</td>
</tr>
<tr>
<td>vim</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM descriptor.</td>
</tr>
<tr>
<td>fmd</td>
<td>O</td>
<td>reference</td>
<td>Reference to Fault Management Descriptor.</td>
</tr>
<tr>
<td>vim-artifact</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM artifact descriptor used to provision the set of parameters on VIM to on board VNFs.</td>
</tr>
<tr>
<td>vim-identity</td>
<td>O*</td>
<td>reference</td>
<td>Reference to VIM identity used to manage VIM resources. It is mandatory to deploy VNFs.</td>
</tr>
<tr>
<td>vnf-package</td>
<td>O</td>
<td>array of references</td>
<td>This is the list of references to VNF Package descriptors. These VNF packages are referred under VDU for software images and day-zero/day-n configuration files.</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>----------</td>
<td>-------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>vld</td>
<td>O</td>
<td>array of references</td>
<td>List of Virtual Link Descriptors (VLD) to be used to connect different VNFs within Network Service domain.</td>
</tr>
<tr>
<td>vnfd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to VNFDs used to build this NSD.</td>
</tr>
<tr>
<td>nested-nsd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
<tr>
<td>require</td>
<td>O</td>
<td>array of references</td>
<td>List of references to NSDs</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

Example RPC

```xml
<nc:rpc message-id="urn:uuid:b0923afe-b32a-4531-9add-7fe32db40c81"
        xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target/>
    <config>
      <nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds">
        <nsd-id>sjc-autoit</nsd-id>
        <vim-identity>vim1</vim-identity>
        <vnf-package>usp_5_7</vnf-package>
        <vld>
          <vld-id>mgmt</vld-id>
          <vl-type>management</vl-type>
          <network-instance>akshay-management</network-instance>
        </vld>
        <vld>
          <vld-id>orch</vld-id>
          <vl-type>orchestration</vl-type>
          <network>orchestration</network>
        </vld>
        <vnf-rack>vnf-rack1</vnf-rack>
        <vnfd>
          <vnfd-id>autovnf</vnfd-id>
          <vnf-type>usp-uas</vnf-type>
          <external-connection-point>
            <vnfc>avf</vnfc>
            <aggregate-connection-point>eth0</aggregate-connection-point>
          </external-connection-point>
          <vnfc>
            <vnfc-id>avf</vnfc-id>
            <vdu>
              <vdu-id>autovnf</vdu-id>
            </vdu>
          </vnfc>
        </vnfd>
      </nsd>
    </config>
  </nc:edit-config>
</nc:rpc>
```
<virtual-link>
  <service-vl>mgmt</service-vl>
</virtual-link>
</aggregate-connection-point>
</vnfc>
</vnfd>
</nsd>
</nc:edit-config>
</nc:rpc>

Example Output

  <ok/>
</rpc-reply>
NSD Deployment

This section describes the parameters used to activate and deactivate the NSD. This section also describes the parameters related to Network Service Record (NSR), Virtual Link and VNF records.

Operations

Activating NSD

Operation: nsd:activate


Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
</tbody>
</table>

Example RPC

```xml
<nc:rpc message-id="urn:uuid:d7b503c2-48f3-4be5-8ae0-73aa100a3e0a" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <nsd:nsd>sjc-autoit</nsd:nsd>
  </nsd:activate>
</nc:rpc>
```

Example Output

```xml
  <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1516895453-047925</transaction-id>
</rpc-reply>
```

Deactivating NSD

Operation: nsd:deactivate


Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
</tbody>
</table>
Example RPC

```xml
<nc:rpc message-id="urn:uuid:2b9f45b7-bb77-4dfd-9540-f5bald5ac4e" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <nsd:nsd-id>fremont-autovnf</nsd:nsd-id>
  </nsd:deactivate>
</nc:rpc>
```

Example Output

```xml
  <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1516898683-491288</transaction-id>
</rpc-reply>
```

Uploading NSD

**Operation:** nsd:upload-file

**Namespace:** xmlns:nsd=http://www.cisco.com/usp/nfv/usp-nsds

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd-id</td>
<td>M</td>
<td>string</td>
<td>NSD name</td>
</tr>
<tr>
<td>source</td>
<td>M</td>
<td>boolean</td>
<td>SRC path and file name</td>
</tr>
<tr>
<td>destination</td>
<td>M</td>
<td>string</td>
<td>Destination path</td>
</tr>
<tr>
<td>vnfd</td>
<td>O</td>
<td>string</td>
<td>VNFD name</td>
</tr>
</tbody>
</table>

**NOTES:**

- upload-file can be invoked from AutoDeploy or AutoVNF, but nsd-id must be AutoVNF name only.
- Generic File Transfer is supported for all components except the vnfds with the following vnf-types – UEM, USP-UAS, ESC.
- Ensure that the specified vnfd(s) are valid so that file or image is transferred successfully.
- RPC can have multiple combination of vnfd parameter
  - With valid vnfd(s)
  - Combination of valid and invalid vnfd(s)
  - With invalid vnfd(s)
  - Without the vnfd(s)

Example RPC

```xml
<ns0:vnf-put-file xmlns:ns0="http://www.cisco.com/usp/scm/vnf-utils">
  <vnfs xmlns="http://www.cisco.com/usp/scm/vnf-utils">
```

35
Example Output


Updating NSD

**Operation:** nsd:update  
**Namespace:** nsd:update xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd</td>
<td>M</td>
<td>string</td>
<td>NSD name</td>
</tr>
<tr>
<td>rolling</td>
<td>M</td>
<td>boolean</td>
<td>Specifies if the rolling is enabled (true) / disabled (false).</td>
</tr>
<tr>
<td>vnfd</td>
<td>M</td>
<td>string</td>
<td>VNFD name, mandatory in case of rolling upgrade</td>
</tr>
</tbody>
</table>
| package        | M        | string | Package descriptor name that should be used to update the vnfd instance mentioned by “vnfd”.

**NOTES:**

- If the rolling false operator is used, the upgrade terminates the entire deployment. In this scenario, the vnfd <vnfd_name> operator should not be included in the command. If it is included, a transaction ID for the upgrade is generated and failed. The AutoVNF upstart log reflects this status.

- Ensure that the upgrade package is defined as a VNF package descriptor within the NSD and that it is specified as the primary package in the VNFM VDU configuration.

- Ensure that the current (pre-upgrade) package is specified as the secondary package in the VNFM VDU configuration in order to provide rollback support in the event of errors.

Example RPC

<brpc message-id="urn:uuid:bac690a2-08af-4c9f-8765-3c907d6e12ba" xmlns="http://www.cisco.com/usp/nfv/usp-nsds">  
<nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds">  
<nsd-id>fremont-avnovn</nsd-id>  
<vim-identity>vim1</vim-identity>  
<vnfd xmlns="http://www.cisco.com/usp/nfv/usp-nsds">  
<vnfd-id>esc</vnfd-id>  
<vnfd-type>esc</vnfd-type>  
<version>6.0</version>  
<configuration>  
<boot-time>1800</boot-time>  
<set-vim-instance-name>true</set-vim-instance-name>  
</configuration>  
</vnfd>  
</nsd>  
</rpc>
<?xml version="1.0" encoding="UTF-8"?>
<nsd xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
  <configuration>
    <external-connection-point>
      <vnfc>esc</vnfc>
      <connection-point>eth0</connection-point>
    </external-connection-point>
    <high-availability>true</high-availability>
    <vnfc>
      <vnfc-id>esc</vnfc-id>
      <health-check>
        <enabled>false</enabled>
      </health-check>
      <vdu>
        <vdu-id>esc</vdu-id>
        <connection-point>
          <connection-point-id>eth0</connection-point-id>
          <virtual-link>
            <service-vl>mgmt</service-vl>
          </virtual-link>
        </connection-point>
        <connection-point>
          <connection-point-id>eth1</connection-point-id>
          <virtual-link>
            <service-vl>orch</service-vl>
          </virtual-link>
        </connection-point>
      </vdu>
    </vnfc>
  </configuration>

  <vim xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
    <vim-id>vim1</vim-id>
    <api-version>v2</api-version>
    <auth-url>http://172.21.201.218:5000/v2.0</auth-url>
    <user>vim-admin-creds</user>
    <tenant>abcxyz</tenant>
  </vim>
  <secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
    <secure-id>vim-admin-creds</secure-id>
    <user>abcxyz</user>
    <password>******</password>
  </secure-token>
  <vdu xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
    <vdu-id>esc</vdu-id>
    <vdu-type>cisco-esc</vdu-type>
    <flavor>
      <vcpus>2</vcpus>
      <ram>4096</ram>
      <root-disk>40</root-disk>
      <ephemeral-disk>0</ephemeral-disk>
      <swap-disk>0</swap-disk>
    </flavor>
    <login-credential>esc_login</login-credential>
    <netconf-credential>esc_netconf</netconf-credential>
    <image>
      <vnf-package>usw_throttle</vnf-package>
    </image>
    <vnf-rack>abcxyz-vnf-rack</vnf-rack>
    <vdu>
      <primary>usw_6.2t</primary>
      <secondary>usw_throttle</secondary>
    </vdu>
    <volume/>
  </vdu>

  <vdu xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
    <vdu-id>esc</vdu-id>
    <vdu-type>esc</vdu-type>
    <flavor>
      <vcpus>2</vcpus>
      <ram>4096</ram>
      <root-disk>40</root-disk>
      <ephemeral-disk>0</ephemeral-disk>
      <swap-disk>0</swap-disk>
    </flavor>
    <login-credential>esc_login</login-credential>
    <netconf-credential>esc_netconf</netconf-credential>
    <image>
      <vnf-package>usw_throttle</vnf-package>
    </image>
    <vnf-rack>abcxyz-vnf-rack</vnf-rack>
    <vdu>
      <primary>usw_6.2t</primary>
      <secondary>usw_throttle</secondary>
    </vdu>
    <volume/>
  </vdu>

  <vim xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
    <vim-id>vim1</vim-id>
    <api-version>v2</api-version>
    <auth-url>http://172.21.201.218:5000/v2.0</auth-url>
    <user>vim-admin-creds</user>
    <tenant>abcxyz</tenant>
  </vim>
  <secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
    <secure-id>vim-admin-creds</secure-id>
    <user>abcxyz</user>
    <password>******</password>
  </secure-token>
  <vdu xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
    <vdu-id>esc</vdu-id>
    <vdu-type>esc</vdu-type>
    <flavor>
      <vcpus>2</vcpus>
      <ram>4096</ram>
      <root-disk>40</root-disk>
      <ephemeral-disk>0</ephemeral-disk>
      <swap-disk>0</swap-disk>
    </flavor>
    <login-credential>esc_login</login-credential>
    <netconf-credential>esc_netconf</netconf-credential>
    <image>
      <vnf-package>usw_throttle</vnf-package>
    </image>
    <vnf-rack>abcxyz-vnf-rack</vnf-rack>
    <vdu>
      <primary>usw_6.2t</primary>
      <secondary>usw_throttle</secondary>
    </vdu>
    <volume/>
  </vdu>
</nsd>
<secure-id esc_netconf</secure-id>
<user>admin</user>
<password>******</password>
</secure-token>
<vnf-packaged xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
  <vnf-package-id>vus_throttle</vnf-package-id>
  <location>http://192.168.200.61:5000/isos/fremont-autovnf_usp_throttle</location>
  <validate-signature>false</validate-signature>
  <configuration>
    <name>staros</name>
  </configuration>
</vnf-packaged>
</config>

Example Output

  <ok/>
</rpc-reply>

Obtaining Deployment Notification During NSD Activation

**Operation:** get

**Namespace:** ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice</td>
<td>Type of operation (same as the type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>----------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

**Example Notification**

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-07T17:29:36+00:00</eventTime>
<ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
  <instance-id>sjc-autoit-instance</instance-id>
  <descriptor-id>sjc-autoit</descriptor-id>
  <transaction-id>1510075776-580848</transaction-id>
  <operation-type>activate-ns-deployment</operation-type>
  <status>requested</status>
</ns-deployment-event>
</notification>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-07T17:29:36+00:00</eventTime>
<ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
  <instance-id>sjc-autoit-instance</instance-id>
  <descriptor-id>sjc-autoit</descriptor-id>
  <transaction-id>1510075776-580848</transaction-id>
  <operation-type>activate-ns-deployment</operation-type>
  <status>in-progress</status>
</ns-deployment-event>
</notification>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-07T17:29:36+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
  <instance-id>vnf-rack1-instance</instance-id>
  <descriptor-id>vnf-rack1</descriptor-id>
  <transaction-id>1510075776-580848</transaction-id>
  <operation-type>activate-vmf-rack</operation-type>
  <status>in-progress</status>
</vim-deployment-event>
</notification>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-07T17:29:36+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
  <instance-id>vnf-rack1-instance</instance-id>
  <descriptor-id>vnf-rack1</descriptor-id>
  <transaction-id>1510075776-580848</transaction-id>
  <operation-type>activate-vmf-rack</operation-type>
  <status>success</status>
</vim-deployment-event>
</notification>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-07T17:29:36+00:00</eventTime>
  <instance-id>usp_5_7-instance</instance-id>
  <descriptor-id>usp_5_7</descriptor-id>
  <transaction-id>1510075776-580848</transaction-id>
  <operation-type>activate-vnf-package</operation-type>
  <status>in-progress</status>
</vnf-package-deployment-event>
</notification>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-07T17:29:36+00:00</eventTime>
  <instance-id>usp_5_7-instance</instance-id>
  <descriptor-id>usp_5_7</descriptor-id>
  <transaction-id>1510075776-580848</transaction-id>
  <operation-type>activate-vnf-package</operation-type>
  <status>success</status>
</vnf-package-deployment-event>
</notification>
```
<transaction-id>1510075776-580848</transaction-id>
<operation-type>activate-vnf-package</operation-type>
<status>success</status>
</vnf-package-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-07T17:36:18.284608+00:00</eventTime>
<vnf-deployment-event xmlns='http://www.cisco.com/usp/nfv/usp-uas-common-oper'>
<instance-id>autovnf-instance</instance-id>
<descriptor-id>autovnf</descriptor-id>
<transaction-id>1510076177-753529</transaction-id>
<operation-type>activate-vnf-deployment</operation-type>
<status>in-progress</status>
</vnf-deployment-event>
</notification>

Fetching VNFR Information

VNF record contains the operational data of an active VNF instance.

Operation: get

Namespace: nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds"

VNF Record (VNFR) Parameters:
### Parameter Name | Required | Type | Description
---|---|---|---
vnfr-id | M | string | Unique ID for this VNFR identity.
vnfd | M | reference | Reference to VNF Descriptor.
state | M | choice string | Overall VNF state. It can be one of the following:
• starting: VNF is starting, instances are being deployed.
• started: All instances of VNF are deployed.
• online: All instances are live. This is done by checking the liveness of the VNFC using one of the health monitoring policies.
• shutting down: VNF is shutting down.
• offline: VNF has been destroyed. This state is never reflected in operational record as VNFR itself is destroyed. This state is applicable to notification sent over UAS stream.
• restarting: One or more instances of VNF are restarting. This state is also used to send notification for VNFC restart.
• failed: UAS failed to deploy the VNF.
• ha-failed: VNF is not in HA state as one of the VNFCs never came online or in failed state. One of the VNFC in this case will be in failed state.
external-connection-point | O | array of external-connection-point | This contains a list of external connection point instances.
vdu | M | array of vdu descriptor | List of VDUs used to deploy this VNF.

### VNFR VDU Descriptor Parameters:
This contains the list of VDUs used to deploy the VNF. Each VDU has a set of VNFC instances running as part of the deployment.

| Parameter Name | Required | Type | Description |
---|---|---|---|
vdu-id | M | reference | Reference to VNFC descriptor
vnfc-instance | M | array of vnfc-instance | One or more lists of VNFC instances.

### VNFC Instance Descriptor Parameters:

| Parameter Name | Required | Type | Description |
---|---|---|---|
vnfc-instance-id | M | string | Unique ID assigned to VNFC Instance.|

41
<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>state</td>
<td>M</td>
<td>choice string</td>
<td>Operational state of VNFC. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• start- requested: Deployment requested.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• reboot- requested: Reboot requested.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• deploying: Deployment in- progress.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• deployed: Deployment completed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• alive: Operational, all good.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• stop- requested: Stop requested.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• undeploying: Un- deployment in- progress.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• undeployed: Instance is undeployed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• rebooting: Instance is rebooting.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• error: Instance in error state. Failed to boot and deploy.</td>
</tr>
<tr>
<td>vnfc</td>
<td>M</td>
<td>reference</td>
<td>Reference to VNFC used as descriptor for this instance.</td>
</tr>
<tr>
<td>flavor- key</td>
<td>M</td>
<td>string</td>
<td>VIM Flavor used to allocate compute resources to this VNFC Instance.</td>
</tr>
<tr>
<td>uuid</td>
<td>M</td>
<td>string</td>
<td>VIM allocated UUID to this VNFC Instance.</td>
</tr>
<tr>
<td>image</td>
<td>M</td>
<td>string</td>
<td>Image name as exists on VIM used for this VNFC instance.</td>
</tr>
<tr>
<td>version</td>
<td>M</td>
<td>string</td>
<td>Software Version for the Image.</td>
</tr>
<tr>
<td>package</td>
<td>O</td>
<td>string</td>
<td>VNF Package reference if applicable.</td>
</tr>
<tr>
<td>uuid</td>
<td>O</td>
<td>string</td>
<td>Image UUID as exists on VIM used for this VNFC instance.</td>
</tr>
<tr>
<td>host</td>
<td>O</td>
<td>string</td>
<td>Compute host ID where this instance is placed. If available. This might not</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>be available if tenant does not have admin privileges.</td>
</tr>
<tr>
<td>connection-</td>
<td>M</td>
<td>array of connection</td>
<td>List of connection point instances.</td>
</tr>
<tr>
<td>point- instance</td>
<td></td>
<td>point instance</td>
<td></td>
</tr>
<tr>
<td>volume- instance</td>
<td>O</td>
<td>array of volume</td>
<td>List of volume instances associated to this VNFC Instance, if available.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>instance</td>
<td></td>
</tr>
</tbody>
</table>
**VNFC Volume Instance Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>volume-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID to volume instance.</td>
</tr>
<tr>
<td>volume</td>
<td>M</td>
<td>reference</td>
<td>Reference to Volume Descriptor.</td>
</tr>
</tbody>
</table>

**VNFC Connection Point Instance Parameters:**

This contains the operational data pertaining to a specific connection point (network-interface).

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connection-point-instance-id</td>
<td>M</td>
<td>string</td>
<td>Connection Point Instance ID where the IP address is associated.</td>
</tr>
<tr>
<td>virtual-link-ref</td>
<td>M</td>
<td>reference</td>
<td>Virtual link provides the network connectivity.</td>
</tr>
<tr>
<td>ip-address</td>
<td>M</td>
<td>inet-ip-address</td>
<td>Virtual IP Address (VIP) assigned to this external connection point.</td>
</tr>
<tr>
<td>mac-address</td>
<td>M</td>
<td>string</td>
<td>MAC address assigned to VIP port.</td>
</tr>
<tr>
<td>connection-point-type</td>
<td>M</td>
<td>choice string</td>
<td>Connection Point type. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• virtual-port: Virtual Port (virtio)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• pnic-pcipt: PCI Pass-thru Port</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• pnic-sriov: PCI SRIOV Port</td>
</tr>
<tr>
<td>port-id</td>
<td>M</td>
<td>string</td>
<td>Unique UUID of the port as assigned by VIM.</td>
</tr>
</tbody>
</table>

**External Connection Point Instance Parameters:**

External Connection Point instance provides necessary information for outside entity to connect to VNF. There are one or more entries when high-availability feature is enabled. A VNF can have multiple external connection points, for example in the UGP, there is one connection exposed by element-manager (UEM) and other connection point by control-function (UGP-CF).

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnfc</td>
<td>M</td>
<td>string</td>
<td>VNFC Instance ID that exposes the connection point.</td>
</tr>
<tr>
<td>aggregate-connection-point</td>
<td>M</td>
<td>connection point instance</td>
<td>See the connection point instance for more details.</td>
</tr>
</tbody>
</table>

**Example RPC**

```
<nc:rpc message-id="urn:uuid:d57b5e07-618b-427f-9c6a-159e815e14b1"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
```
Example Output

<nf:filter type="subtree">
        <data>
                <vnfd>esc</vnfd>
                    <vnfc>esc</vnfc>
                    <vnfc>esc</vnfc>
                    <vdu>
                        <external-connection-point>
                            <vdu id="urn:uuid:853dd378-53dd-4093-985e-e81580eeabe Vanderpool"/>
                            <vdu id="urn:uuid:853dd378-53dd-4093-985e-e81580eeabe Vanderpool"/>
                        </external-connection-point>
                        <connection-point-instance>
                            <vdu id="urn:uuid:853dd378-53dd-4093-985e-e81580eeabe Vanderpool"/>
                            <vdu id="urn:uuid:853dd378-53dd-4093-985e-e81580eeabe Vanderpool"/>
                        </connection-point-instance>
                    </vdu>
                </vnfr>
            </vnfr>
        </data>
    </vnfr>
</nf:filter>
<ip-address>192.168.1.97</ip-address>
<mac-address>fa:16:3e:25:b9:33</mac-address>
<connection-point-type>vnfc</connection-point-type>
<port-id>7219cabe-f442-401a-865f-fa875403406c</port-id>
</connection-point-instance>
<connection-point-instance>
<connection-point-instance-id>eth1</connection-point-instance-id>
<virtual-link-ref>bmarconi-management</virtual-link-ref>
<ip-address>50.50.50.13</ip-address>
<mac-address>fa:16:3e:71:2a:63</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>634f2902-583c-473a-9418-67187b3d018d</port-id>
</connection-point-instance>
<connection-point-instance>
<connection-point-instance-id>eth2</connection-point-instance-id>
<virtual-link-ref>bmarconi-management</virtual-link-ref>
<ip-address>10.2.3.21</ip-address>
<mac-address>fa:16:3e:d6:65:2f</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>c2bf33d9-0b4c-4de6-856d-359d785eeda2</port-id>
</connection-point-instance>
<connection-point-instance>
<connection-point-instance-id>eth3</connection-point-instance-id>
<virtual-link-ref>di internal2</virtual-link-ref>
<ip-address>192.168.2.8</ip-address>
<mac-address>fa:16:3e:81:1e:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>07ecd637-baa2-48b0-bbce-8c6572d2d32e</port-id>
</connection-point-instance>
</vnfc-instance>
<vnfc-instance id="fremont-avontnf-vpc-aselvana-cf-2">
<state>alive</state>
<vnfc>cf</vnfc>
<flavor-key>freemont-avontnf-vpc-aselvana-cf</flavor-key>
<uuid>8e946784-ca87-4b3a-9977-43daf149df7f</uuid>
</vnfc-instance>
<connection-point-instance>
<connection-point-instance-id>eth0</connection-point-instance-id>
<virtual-link-ref>di internal1</virtual-link-ref>
<ip-address>192.168.1.124</ip-address>
<mac-address>fa:16:3e:6f:68:d4</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>f82ce0ad-1284-47bf-b747-119cb0de987e</port-id>
</connection-point-instance>
</vnfc-instance>
<connection-point-instance>
<connection-point-instance-id>eth1</connection-point-instance-id>
<virtual-link-ref>bmarconi-management</virtual-link-ref>
<ip-address>50.50.50.31</ip-address>
<mac-address>fa:16:3e:5f:c3:57</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>d1f9dbbb-77e4-4333-a9e2-5ec66f2fca20</port-id>
</connection-point-instance>
<connection-point-instance>
<connection-point-instance-id>eth2</connection-point-instance-id>
<virtual-link-ref>bmarconi-management</virtual-link-ref>
<ip-address>10.2.3.5</ip-address>
<mac-address>fa:16:3e:63:cl:29</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>19abe3ba-e5ac-43e9-b72f-197d0d6bb44e</port-id>
</connection-point-instance>
<connection-point-instance>
<connection-point-instance-id>eth3</connection-point-instance-id>
</virtual-link-ref>di-internal2</virtual-link-ref>
</ip-address>192.168.2.35</ip-address>
</mac-address>fa:16:3e:82:93:59</mac-address>
</connection-point-type>pnic-sriov</connection-point-type>
<port-id>a9e3ad25-3b37-49a5-ba9f-98bab91c07b6</port-id>
</connection-point-instance>
</vnfc-instance>
</vdu>
</vdu-id>em</vdu-id>
</vnfc-instance>
<vnfc-instance-id>fremont-avonvnf-vpc-aselvana-em-1</vnfc-instance-id>
</state>alive</state>
</vnfc-instance>
</vnfc-instance>
</vnfc-instance-id>fremont-autovnf-vpc-aselvana-em</vnfc-instance-id>
</flavor-key>fremont-autovnf-vpc-aselvana-em</flavor-key>
<uuid>7db9ae6-9b1e-48e6-a2d2-bb7a16a1d3b3</uuid>
</image>
<name>/em-bootstrap/</name>
</version>Version: 6.0.0, SHA1: cfbdb8b1, Date: Tue Jan 23 22:36:19 EST 2018</version>
</package>uspn_5.7</package>
<uuid>b4706545-a424-4897-9f61-6a6ae2401139</uuid>
</image>
</host>tb2-compute-3.localdomain</host>
</vdu-type>element-manager</vdu-type>
</connection-point-instance>
</connection-point-instance-id>eth0</connection-point-instance-id>
</virtual-link-ref>autoit-88c-orch</virtual-link-ref>
</ip-address>10.2.3.17</ip-address>
</mac-address>fa:16:3e:bc:bb:68</mac-address>
</connection-point-type>virtual-port</connection-point-type>
<port-id>51c860fe-ba7f-4a97-9399-0d1ae8486a21</port-id>
</connection-point-instance>
</connection-point-instance>
</connection-point-instance-id>em</connection-point-instance-id>
</virtual-link-ref>autoit-sjc-orch</virtual-link-ref>
</ip-address>50.50.50.26</ip-address>
</mac-address>fa:16:3e:2b:20:e0</mac-address>
</connection-point-type>virtual-port</connection-point-type>
<port-id>96adb0b36-394e-4156-9d19-e938e236da8e</port-id>
</connection-point-instance>

<connection-point-instance>
<connection-point-instance-id>eth1</connection-point-instance-id>
</virtual-link-ref>bmarconi-management</virtual-link-ref>
</ip-address>50.50.50.26</ip-address>
</mac-address>fa:16:3e:2b:20:e0</mac-address>
</connection-point-type>virtual-port</connection-point-type>
<port-id>96adb0b36-394e-4156-9d19-e938e236da8e</port-id>
</connection-point-instance>
</vnfc-instance>
<vnfc-instance-id>fremont-avonvnf-vpc-aselvana-em-2</vnfc-instance-id>
</state>alive</state>
</vnfc-instance>
</vnfc-instance>
<vnfc-instance-id>fremont-avonvnf-vpc-aselvana-em</vnfc-instance-id>
</flavor-key>fremont-avonvnf-vpc-aselvana-em</flavor-key>
<uuid>8e9fc918-8c17-4ace-9e45-0750df10a365</uuid>
</image>
<name>/em-bootstrap/</name>
</version>Version: 6.0.0, SHA1: cfbdb8b1, Date: Tue Jan 23 22:36:19 EST 2018</version>
</package>uspn_5.7</package>
<uuid>b4706545-a424-4897-9f61-6a6ae2401139</uuid>
</image>
</host>tb2-compute-11.localdomain</host>
</vdu-type>element-manager</vdu-type>
</connection-point-instance>
</connection-point-instance-id>eth0</connection-point-instance-id>
</virtual-link-ref>autoit-88c-orch</virtual-link-ref>
</ip-address>10.2.3.12</ip-address>
</mac-address>fa:16:3e:af:b2:22</mac-address>
</connection-point-type>virtual-port</connection-point-type>
<port-id>f3292817-00e6-468d-bcfb-b74cad336815</port-id>
</connection-point-instance>
</connection-point-instance>
<connection-point-instance-id>eth1</connection-point-instance-id>
</virtual-link-ref>bmarconi-management</virtual-link-ref>
</ip-address>50.50.50.27</ip-address>
</mac-address>fa:16:3e:23:e0:70</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>641e039-0a12-46d4-984d-3deca819edd1</port-id>
</connection-point-instance>
</vnfc-instance>
</vnfc-instance>

<vnfc-instance>
<vnfc-instance-id>fremont-autovnf-vpc-aselvana-em-3</vnfc-instance-id>
<state>alive</state>
<vnfc>em</vnfc>
</vnfc-instance>

<flavor-key>fremont-autovnf-vpc-aselvana-em</flavor-key>
<uuid>196b0644-b385-439f-9549-54aaf982fcfe</uuid>
</image>

<name>/em-bundle/em-6_0_0_2166.qcow2</name>
</version>Version: 6.0.0, SHA1: cbfd8bb1, Date: Tue Jan 23 22:36:19 EST 2018</version>
</package>
</package>
</package>
</vnfc-instance>
</vdu>

<vdu>
<vdu-id>sf</vdu-id>
</vdu>

<vnfc-instance>
<vnfc-instance-id>fremont-autovnf-vpc-aselvana-sf-1</vnfc-instance-id>
<state>error</state>
<vnfc>sf</vnfc>
</vnfc-instance>

<flavor-key>fremont-autovnf-vpc-aselvana-sf</flavor-key>
<uuid>12642ff4-42cd-498e-98c6-12667a38441</uuid>
</image>

<name>/ugp-bundle/qvpc-dlxf.qcow2</name>
</package>
</package>
</package>
</vnfc-instance>
</vdu>

<vdu>
<vdu-id>sriov</vdu-id>
</vdu>

<vnfc-instance>
<vnfc-instance-id>eth0</vnfc-instance-id>
<state>error</state>
<vnfc>sf</vnfc>
</vnfc-instance>

<flavor-key>fremont-autovnf-vpc-aselvana-sf</flavor-key>
<uuid>12642ff4-42cd-498e-98c6-12667a38441</uuid>
</image>

<name>/em-bundle/em-6_0_0_2166.qcow2</name>
</version>Version: 6.0.0, SHA1: cbfd8bb1, Date: Tue Jan 23 22:36:19 EST 2018</version>
</package>
</package>
</package>
</vnfc-instance>
</vdu>

<vdu>
<vdu-id>sriov</vdu-id>
</vdu>
<connection-point-instance-id>eth2</connection-point-instance-id>
<virtual-link-ref>aselsvna-vpc-svc</virtual-link-ref>
<ip-address>22.12.11.7</ip-address>
<mac-address>fa:16:3e:9a:74:2f</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>055f5c03437f419</port-id>

<connection-point-instance-id>eth0</connection-point-instance-id>
<virtual-link-ref>di-internal1</virtual-link-ref>
<ip-address>10.2.3.18</ip-address>
<mac-address>fa:16:3e:34:dd:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>2c3d37c06a-8928-97f0c6a909bb</port-id>

<connection-point-instance-id>eth1</connection-point-instance-id>
<virtual-link-ref>autolit-sjc-orclh</virtual-link-ref>
<ip-address>10.2.3.18</ip-address>
<mac-address>fa:16:3e:34:dd:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>2c3d37c06a-8928-97f0c6a909bb</port-id>

<connection-point-instance-id>eth2</connection-point-instance-id>
<virtual-link-ref>aselsvna-vpc-svc</virtual-link-ref>
<ip-address>22.12.11.7</ip-address>
<mac-address>fa:16:3e:9a:74:2f</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>055f5c03437f419</port-id>

<connection-point-instance-id>eth3</connection-point-instance-id>
<virtual-link-ref>di-internal2</virtual-link-ref>
<ip-address>192.168.2.29</ip-address>
<mac-address>fa:16:3e:5d:77:62</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>3c2d802c-4294-4c4c-able-7f8ecca21794</port-id>

</vnfc-instance>

<vnfc-instance-id>fremont-autovnf-vpc-aselsvna-sf-2</vnfc-instance-id>
<state>alive</state>
<vnfc-sf/>
<flavor-key>fremont-autovnf-vpc-aselsvna-sf</flavor-key>
<uuid>5ccb21d7-4b02-4b9b-9edf-773ea61ea83</uuid>
<image>
<name>/ugp-bundle/qvpc-di-xf.qcow2</name>
.VERSION: 21.6.0.68695, SHA1: NA, Date: Wed Jan 24 21:02:06 PST 2018
</version>
</vnfc-instance>

<vnfc-instance-id>fremont-autovnf-vpc-aselsvna-sf</vnfc-instance-id>
<state>alive</state>
<vnfc-sf/>
<flavor-key>fremont-autovnf-vpc-aselsvna-sf</flavor-key>
<uuid>055f5c03437f419</uuid>
<image>
<name>/ugp-bundle/qvpc-di-xf.qcow2</name>
.VERSION: 21.6.0.68695, SHA1: NA, Date: Wed Jan 24 21:02:06 PST 2018
</version>
</vnfc-instance>

<connection-point-instance-id>eth0</connection-point-instance-id>
<virtual-link-ref>di-internal1</virtual-link-ref>
<ip-address>10.2.3.18</ip-address>
<mac-address>fa:16:3e:34:dd:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>2c3d37c06a-8928-97f0c6a909bb</port-id>

<connection-point-instance-id>eth1</connection-point-instance-id>
<virtual-link-ref>autolit-sjc-orclh</virtual-link-ref>
<ip-address>10.2.3.18</ip-address>
<mac-address>fa:16:3e:34:dd:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>2c3d37c06a-8928-97f0c6a909bb</port-id>

<connection-point-instance-id>eth2</connection-point-instance-id>
<virtual-link-ref>aselsvna-vpc-svc</virtual-link-ref>
<ip-address>22.12.11.7</ip-address>
<mac-address>fa:16:3e:9a:74:2f</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>055f5c03437f419</port-id>

<connection-point-instance-id>eth3</connection-point-instance-id>
<virtual-link-ref>di-internal2</virtual-link-ref>
<ip-address>192.168.2.29</ip-address>
<mac-address>fa:16:3e:5d:77:62</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>3c2d802c-4294-4c4c-able-7f8ecca21794</port-id>

</vnfc-instance>

<connection-point-instance-id>eth0</connection-point-instance-id>
<virtual-link-ref>di-internal1</virtual-link-ref>
<ip-address>10.2.3.18</ip-address>
<mac-address>fa:16:3e:34:dd:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>2c3d37c06a-8928-97f0c6a909bb</port-id>

<connection-point-instance-id>eth1</connection-point-instance-id>
<virtual-link-ref>autolit-sjc-orclh</virtual-link-ref>
<ip-address>10.2.3.18</ip-address>
<mac-address>fa:16:3e:34:dd:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>2c3d37c06a-8928-97f0c6a909bb</port-id>

<connection-point-instance-id>eth2</connection-point-instance-id>
<virtual-link-ref>aselsvna-vpc-svc</virtual-link-ref>
<ip-address>22.12.11.7</ip-address>
<mac-address>fa:16:3e:9a:74:2f</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>055f5c03437f419</port-id>

<connection-point-instance-id>eth3</connection-point-instance-id>
<virtual-link-ref>di-internal2</virtual-link-ref>
<ip-address>192.168.2.29</ip-address>
<mac-address>fa:16:3e:5d:77:62</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>3c2d802c-4294-4c4c-able-7f8ecca21794</port-id>

</vnfc-instance>

<connection-point-instance-id>eth0</connection-point-instance-id>
<virtual-link-ref>di-internal1</virtual-link-ref>
<ip-address>10.2.3.18</ip-address>
<mac-address>fa:16:3e:34:dd:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>2c3d37c06a-8928-97f0c6a909bb</port-id>

<connection-point-instance-id>eth1</connection-point-instance-id>
<virtual-link-ref>autolit-sjc-orclh</virtual-link-ref>
<ip-address>10.2.3.18</ip-address>
<mac-address>fa:16:3e:34:dd:9e</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>2c3d37c06a-8928-97f0c6a909bb</port-id>

<connection-point-instance-id>eth2</connection-point-instance-id>
<virtual-link-ref>aselsvna-vpc-svc</virtual-link-ref>
<ip-address>22.12.11.7</ip-address>
<mac-address>fa:16:3e:9a:74:2f</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>055f5c03437f419</port-id>

<connection-point-instance-id>eth3</connection-point-instance-id>
<virtual-link-ref>di-internal2</virtual-link-ref>
<ip-address>192.168.2.29</ip-address>
<mac-address>fa:16:3e:5d:77:62</mac-address>
<connection-point-type>virtual-port</connection-point-type>
<port-id>3c2d802c-4294-4c4c-able-7f8ecca21794</port-id>

</vnfc-instance>
Network Service Record Parameters

NSR contains the operational information about a running instance of NSD.
### NSD Deployment

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsr-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID for this NSR identity.</td>
</tr>
<tr>
<td>nsd</td>
<td>M</td>
<td>reference</td>
<td>Reference to NSD.</td>
</tr>
<tr>
<td>vlr</td>
<td>M</td>
<td>array of vlr</td>
<td>List of Virtual Link Record (VLR) as used at NSD level.</td>
</tr>
<tr>
<td>vnfr</td>
<td>O</td>
<td>array of references to vnfr</td>
<td>Reference to one or more VNFRs on boarded as part of this NSD.</td>
</tr>
<tr>
<td>vnf-package</td>
<td>O</td>
<td>array of references to vnf-packager</td>
<td>Reference to one or more vnf-packagers on boarded as part of this NSD.</td>
</tr>
<tr>
<td>vnf-rack</td>
<td>O</td>
<td>array of references to vnf-rackr</td>
<td>Reference to one or more vnf-rackr on boarded as part of this NSD.</td>
</tr>
</tbody>
</table>

#### Virtual Link Record Parameters

Virtual links are shared across VNFs at network service level. VLR provides housekeeping to manage the shared resource. It creates virtual link at once and keeps references of VNF that are using it. It is done by adding VNFR references to VLR. During deactivation of VNF/NSD, the reference is removed as VNF stops using the resource and the virtual link is deactivated if there are no more references to it.

VLR records are only maintained for the resources UAS creates on VIM. For virtual link, where UAS uses a pre-created network-instance from VIM, this entry is not created.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vlr-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID for this VLR identity.</td>
</tr>
<tr>
<td>network</td>
<td>M</td>
<td>reference</td>
<td>Reference to Network Descriptor used to create this virtual link record.</td>
</tr>
<tr>
<td>vnfr</td>
<td>M</td>
<td>array of references to vnfr</td>
<td>List of VNFs using this virtual link.</td>
</tr>
</tbody>
</table>
Secure Token Configuration

The UAS defines the concept of secure token, an object that protects confidentiality. Whenever a configuration parameter is defined, the UAS makes sure:

- this parameter is stored securely in a database by applying AES based encryption.
- it is not shown in plain text either in configuration or in any logs.
- the parameters are passed over secure channel when the UAS passes these secure tokens to other entities over network. For example, over SSH or HTTPS or via encrypted storage (link cinder volumes).

When booting up the UAS, an encryption key is passed to the UAS and is stored in file system which only the root user can access. The UAS uses this encryption key to apply the encryption. Hence, none over API interface can have access to this key so there is no way to decrypt the encrypted parameters on boarded to UAS as part of the configuration.

The UAS uses these secure token-based parameters in secure token credential descriptors or secure token SSH key descriptors to define credentials (user name/password pair) or SSH keys.

The UAS defines the following three roles based on which the secure token configuration is managed:

- admin: Admin user which can edit the configuration.
- oper: Operational user which has read-only access.
- security-admin: This user has all admin privileges along with the capabilities to add and remove users, access rules.

The UAS defines a special type of secure token as object called Service Configuration Manager (SCM) to define these roles. This secure token can be for the VDUs which provide API services. For example, UAS, VNF-EM, CF, ESC, and so on.

Operations

Configuring Secure Token

Operation: edit-config

Namespace: secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token"

Parameters:
Secure Token Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>secure-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this secure token.</td>
</tr>
<tr>
<td>user</td>
<td>O</td>
<td>string</td>
<td>User name for this credential. This is optional and used to allow secure token within a descriptor that requires only password. For example, you have a fixed user name, this string is encrypted via AES, 128-bits encryption using CBC mode.</td>
</tr>
<tr>
<td>password</td>
<td>M</td>
<td>string</td>
<td>Password for this credential. This is mandatory. This string is encrypted via AES, 128-bits encryption using CBC mode.</td>
</tr>
</tbody>
</table>

Example RPC

```xml
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
      <secure-token xmlns="http://www.cisco.com/usf/nfv/usp-secure-token">
        <secure-id>openstack</secure-id>
        <user>admin</user>
        <password>Cisco@123=</password>
      </secure-token>
      <secure-token xmlns="http://www.cisco.com/usf/nfv/usp-secure-token">
        <secure-id>esc-login</secure-id>
        <user>$8$Pv1HrYu+DSfLz70gDxwzL9v4g6r2qDRMgq7SnwjljMMF4=</user>
        <password>$8$RZ5GEoxZKKZ0Tg4ru8d5/D8wYFXr/c0vjrZ/FWxjw1iw=</password>
      </secure-token>
    </config>
  </nc:edit-config>
</nc:rpc>
```

Notes:

- For security reasons, the input parameters for the secure tokens can be provided in any of the following ways:
  - Use a secure transport. NETCONF is a secure transport over SSH. Hence, the input provided over this transport is protected. The same applies to REST over HTTPS.
  - Log in to the UAS CLI over SSH.
  - As these channels are secure, you can provide the parameter values securely. Alternatively, you can encrypt the strings using the key provided to UAS and pass that as input.
- OpenStack password is passed as plaintext while ESC user name and passwords are encrypted and sent. As this is NETCONF RPC, the payload is encrypted on wire and there is no need to provide encrypted input.

Example Output

```xml
  <ok/>
</rpc-reply>
```

Configuring SSH Keys

**Operation:** `edit-config`
**Namespace:** ssh-key xmlns="http://www.cisco.com/usp/nfv/usf-secure-token"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ssh-key-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this secure token.</td>
</tr>
<tr>
<td>private-key</td>
<td>O</td>
<td>string</td>
<td>Private Key. This is optional and used to allow secure token within a descriptor that requires only public-key, for example, you need authorized key to pass public key. This string is encrypted via AES, 128-bit encryption using CBC mode.</td>
</tr>
<tr>
<td>public-key</td>
<td>M</td>
<td>string</td>
<td>Public Key. This is mandatory. This string is encrypted via AES, 128-bit encryption using CBC mode.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
      <ssh-key xmlns="http://www.cisco.com/usp/nfv/usf-secure-token">
        <ssh-key-id>esc_login</ssh-key-id>
        <public-key>AAAAB3NzaC1yc2EAAAADAQABAAABAQDnMi+HxGSKDbELL4sWAJROV5WPaJcJt7KGAoG3Ec70PiYtKfOLB1a-CNnBICeCT4/KxhrLFujqAIH0W0lletX3xsvZGFNYrwLfqXCK-wN0fx4/4L3J1JAPcVFSpQyt5/s583SMyn1EZ0NkZoFqG/nXPaGxfaqUGDrlGwnQ8wsZdF/xg2zykBGd9eEpwuOLLNxfEq-DtvOvKcQJ6zMN58qH55TKIe5mtd7d+8z58dzCD4eMKSM31Nzt1H75fnhVvDIF8aTGF-7rGJ2z4xRqQ8uNMe3j3pRm3oUXUMSEFXE34ZBBbrsS02tqkJuBK4fPNL0P6G8w0cH0t root@akshay-uas16</public-key>
      </ssh-key>
    </config>
  </nc:edit-config>
</nc:rpc>
```

**Example Output**

```xml
  <ok/>
</rpc-reply>
```

**Configuring Secure SCM**

**Operation:** edit-config

**Namespace:** scm xmlns="http://www.cisco.com/usp/nfv/usf-secure-token"

**Parameters:**
Secure Token Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>scm-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this secure token.</td>
</tr>
<tr>
<td>admin</td>
<td>M</td>
<td>reference</td>
<td>Login credentials for SCM Admin role. Reference to secure-token-credential descriptor to enable “admin” role on API engine.</td>
</tr>
<tr>
<td>oper</td>
<td>M</td>
<td>reference</td>
<td>Login Credentials for SCM Operator role. Reference to secure-token-credential descriptor to enable “oper” role on API engine.</td>
</tr>
<tr>
<td>security</td>
<td>M</td>
<td>reference</td>
<td>Login Credentials for SCM Security role. Reference to secure-token-credential descriptor to enable “security” role on API engine.</td>
</tr>
<tr>
<td>encryption-key</td>
<td>M</td>
<td>string</td>
<td>Encryption key for API engine to support secure tokens.</td>
</tr>
</tbody>
</table>

Example RPC

```xml
<nc:rpc message-id="urn:uuid:43e3f76b-6f5d-4bb1-8d26-f13176f7fb5d" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
      <secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
        <secure-id>em-admin</secure-id>
        <user>admin</user>
        <password>Cisco@123</password>
      </secure-token>
      <secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
        <secure-id>em-oper</secure-id>
        <user>oper</user>
        <password>Cisco$12345</password>
      </secure-token>
      <secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
        <secure-id>em-security</secure-id>
        <user>security</user>
        <password>Cisco$123</password>
      </secure-token>
      <scm xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
        <scm-id>scm</scm-id>
        <admin>em-admin</admin>
        <oper>em-oper</oper>
        <security>em-security</security>
      </scm>
    </config>
  </nc:edit-config>
</nc:rpc>
```

Example Output

```xml
  <ok/>
</rpc-reply>
```
VIM Orchestrator Configuration

The UAS provides VIM-ORCHD configuration to deploy VIM orchestrator or undercloud. The UAS provides a generic set of APIs to deploy undercloud and overcloud deployment. It integrates with different plugins to manage specific types of deployment.

The VIM-ORCHD configuration comprises the following:

- Network Service Descriptor (NSD)
- Secure token
- VIM Orchestrator descriptor (vim-orchd)
  - NFVI node
  - NFVI Point-of-Presence descriptor (nfvi-popd)
  - Security server
  - Content Delivery Network (CDN) server
  - External Network
  - Provisioning Network

For information on configuring the secure tokens, see the Secure Token Configuration.

Operations

Configuring VIM-ORCHD

**Operation:** edit-config

**Namespace:** vim-orchd xmlns="http://www.cisco.com/usp/nfv/usps-vim-orch"

**VIM-ORCHD Parameters:**
## VIM Orchestrator Configuration

### Parameter Name | Required | Type          | Description
--- | --- | --- | ---
vim-orch-id | M | string (1-16) | Unique identifier for VIM Orchestrator descriptor.
hostname | M | string | Hostname for VIM Orchestrator node.
domain-name | M | string | Domain to be used for this VIM Orchestrator deployment.
dns | M | array of strings | List of DNS servers to be used for name resolution in orchestrator node.
login-credential | M | secure-token | Login credential to be configured in VIM Orchestrator for authentication.
image | M | Inet:uri | Red Hat image to be used for deploying VIM Orchestrator.

### NFVI PoPD Parameters:

### Parameter Name | Required | Type | Description
--- | --- | --- | ---
nfvi-pop-id | M | string | Unique identifier for NFVI Point-Of-Presence descriptor.
description | O | string | Detailed information about the use of this POPD.
deployment-flavor | M | choice-string | This specifies the number of VNFs that are supported on the NFVI-POP along with the type of hardware used. The number of VNFs that can be supported on an NFVI-POP depends on the availability of the compute nodes.

- ucs-1-vnf: NFVI POP supporting 1 VNF deployment on UCS hardware
- ucs-2-vnf: NFVI POP supporting 2 VNF deployment on UCS hardware
- ucs-3-vnf: NFVI POP supporting 3 VNF deployment on UCS hardware
- ucs-4-vnf: NFVI POP supporting 4 VNF deployment on UCS hardware
nfvi-node-id | M | string | Unique identifier for a hardware node within a POP.
physical-server-manager/ip-address | M | inet:ip-address | IP-address to access the IPMI interface for that hardware node.
physical-server-manager/login-credential | M | secure token | Login credentials to be used for IPMI interface authentication.

### NFVI Parameters:
**VIM Orchestrator Configuration**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>M</td>
<td>string</td>
<td>Associate the NFVI node identifier defined in NFVI- POPd that will be used to host VIM Orchestrator VM.</td>
</tr>
<tr>
<td>ip-address</td>
<td>M</td>
<td>inet:ip-address</td>
<td>IP address to access the host OS of the NFVI node (SSH).</td>
</tr>
<tr>
<td>login-credential</td>
<td>M</td>
<td>secure-token</td>
<td>Authentication information to access the host OS of the NFVI node (SSH).</td>
</tr>
</tbody>
</table>

**Satellite Server Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>inet:ip-address</td>
<td>IP address of satellite server.</td>
</tr>
<tr>
<td>hostname</td>
<td>M</td>
<td>string</td>
<td>Hostname of satellite server.</td>
</tr>
<tr>
<td>domain-name</td>
<td>M</td>
<td>string</td>
<td>Domain name of satellite server.</td>
</tr>
<tr>
<td>organization</td>
<td>M</td>
<td>string</td>
<td>Organization name to be used for Red Hat subscription.</td>
</tr>
<tr>
<td>activation-key</td>
<td>M</td>
<td>string</td>
<td>Activation key associated with the organization provided for Red Hat subscription.</td>
</tr>
<tr>
<td>pool-id</td>
<td>M</td>
<td>string</td>
<td>Pool identifier associated with a view exposed from satellite server. Identifies all the RPM's that are allowed to be accessed.</td>
</tr>
<tr>
<td>release-version</td>
<td>O</td>
<td>string</td>
<td>RHEL release version to be locked.</td>
</tr>
</tbody>
</table>

This is an optional parameter and used to enable RHEL version locking to host their own local repository using Red Hat's Satellite server. If release-version is not provided, then it assumes the default version which already exists.

**NOTE:** This configuration is applicable for organizations that like to host their own local repository using Red Hat's Satellite Server configuration.

**CDN Server Parameters:**
### VIM Orchestrator Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>login-credential</td>
<td>M</td>
<td>secure-token</td>
<td>Login credentials to authenticate Red Hat CDN service for repository access.</td>
</tr>
<tr>
<td>proxy</td>
<td>M</td>
<td>string</td>
<td>Proxy server along with port number to access the external CDN service provided by Red Hat.</td>
</tr>
<tr>
<td>pool-id</td>
<td>M</td>
<td>string</td>
<td>Pool identifier associated with a view exposed from CDN server. Identifies all the RPMs that are allowed to be accessed.</td>
</tr>
</tbody>
</table>

**NOTE:** This configuration is applicable for organizations that like to use the Red Hat CDN Server to access the repositories for installation.

### External Network Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>inet:ip-address</td>
<td>IP address to be assigned to the interface in VIM Orchestrator attached to the external network.</td>
</tr>
<tr>
<td>netmask</td>
<td>M</td>
<td>inet:ip-address</td>
<td>Netmask associated with the external network.</td>
</tr>
<tr>
<td>gateway-ip</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Gateway IP address to be associated with the interface attached to the external network.</td>
</tr>
</tbody>
</table>

### Provisioning Network Parameters:
### VIM Orchestrator Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>inet:ip-address</td>
<td>IP address to be assigned to the interface in VIM Orchestrator attached to the provisioning network.</td>
</tr>
<tr>
<td>netmask</td>
<td>M</td>
<td>inet:ip-address</td>
<td>Netmask associated with the provisioning network.</td>
</tr>
<tr>
<td>gateway-ip</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Gateway IP address to be associated with the interface attached to the provisioning network.</td>
</tr>
<tr>
<td>public-vip</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Virtual IP address for public endpoints of undercloud services.</td>
</tr>
<tr>
<td>admin-vip</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Virtual IP address for admin endpoints of undercloud services.</td>
</tr>
<tr>
<td>dhcp-ip-range/start</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Start of DHCP allocation range (inclusive) for PXE and DHCP of overcloud instances.</td>
</tr>
<tr>
<td>dhcp-ip-range/end</td>
<td>O</td>
<td>inet:ip-address</td>
<td>End of DHCP allocation range (inclusive) for PXE and DHCP of overcloud instances.</td>
</tr>
<tr>
<td>inspection-ip-range/start</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Start of temporary IP addresses (inclusive) assigned to nodes while doing introspection.</td>
</tr>
<tr>
<td>inspection-ip-range/end</td>
<td>O</td>
<td>inet:ip-address</td>
<td>End of temporary IP addresses (inclusive) assigned to nodes while doing introspection.</td>
</tr>
<tr>
<td>network-cidr</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Network CIDR for neutron-managed network for overcloud instances.</td>
</tr>
<tr>
<td>masquerade-network</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Network that is masqueraded for external access.</td>
</tr>
</tbody>
</table>

### VCPU and Memory Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vcpus</td>
<td>M</td>
<td>uint16</td>
<td>Number of virtual CPUs to be allocated for VIM Orchestrator VM.</td>
</tr>
<tr>
<td>ram</td>
<td>M</td>
<td>uint32</td>
<td>Amount of memory (in Megabytes) to be allocated for VIM Orchestrator VM.</td>
</tr>
<tr>
<td>root-disk</td>
<td>M</td>
<td>uint16</td>
<td>Amount of disk space (in Gigabytes) to be allocated for VIM Orchestrator VM for root disk.</td>
</tr>
</tbody>
</table>

### Example RPC

```xml
<nc:rpc message-id="urn:uuid:f49e76f0-bebf-4597-8eed-7660ab9996f7"
     xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
```

62
<nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds">
  <nsd-id>sjc</nsd-id>
  <vim-orch>sjc-vim-orch</vim-orch>
</nsd>

<secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
  <secure-id>cimc</secure-id>
  <user>admin</user>
  <password>Cisco@123</password>
</secure-token>

<secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
  <secure-id>stack</secure-id>
  <user>stack</user>
  <password>Cisco@123</password>
</secure-token>

<secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
  <secure-id>ssh-baremetal</secure-id>
  <user>nfvi</user>
  <password>Cisco@123</password>
</secure-token>

<nfvi-popd xmlns="http://www.cisco.com/usp/nfv/usp-vim-orch">
  <nfvi-pop-id>seattle</nfvi-pop-id>
  <deployment-flavor>ucs-1-vnf</deployment-flavor>
  <nfvi-node>
    <nfvi-node-id>node_0</nfvi-node-id>
    <physical-server-manager>
      <ip-address>192.100.4.1</ip-address>
      <login-credential>cimc</login-credential>
    </physical-server-manager>
  </nfvi-node>
</nfvi-popd>

<nfvi-orchd xmlns="http://www.cisco.com/usp/nfv/usp-vim-orch">
  <vim-orch-id>sjc-vim-orch</vim-orch-id>
  <hostname>undercloud</hostname>
  <domain-name>cisco.com</domain-name>
  <dns>171.70.168.183</dns>
  <login-credential>stack</login-credential>
  <satellite-server>
    <ip-address>10.23.252.119</ip-address>
    <hostname>rh-satellite</hostname>
    <domain-name>cisco.com</domain-name>
    <organization>ultram</organization>
    <activation-key>openstack 10</activation-key>
    <pool-id>8a977c7f50ca9df015d2ce1cb4d06a</pool-id>
    <release-version>7.5</release-version>
  </satellite-server>
  <external-network>
    <ip-address>172.28.185.157</ip-address>
    <netmask>255.255.255.0</netmask>
  </external-network>
  <provisioning-network>
    <ip-address>192.200.0.1</ip-address>
    <netmask>255.0.0.0</netmask>
    <public-vip>192.200.0.2</public-vip>
    <admin-vip>192.200.0.3</admin-vip>
    <dhcp-ip-range>
      <start>192.200.0.101</start>
      <end>192.200.0.150</end>
    </dhcp-ip-range>
    <inspection-ip-range>
      <start>192.200.0.201</start>
      <end>192.200.0.250</end>
    </inspection-ip-range>
  </provisioning-network>
  <network-cidr>192.0.0.0/8</network-cidr>
  <masquerade-network>192.0.0.0/8</masquerade-network>
  <flavor>
    <vcpus>4</vcpus>
    <ram>16384</ram>
    <root-disk>100</root-disk>
  </flavor>
</nfvi-orchd>
<ephemeral-disk>0</ephemeral-disk>
<swap-disk>0</swap-disk>
</flavor>
<nfvi-node>
  <id>node_0</id>
  <ip-address>172.28.185.124</ip-address>
  <login-credential>ssh-baremetal</login-credential>
</nfvi-node>
</image>redhat</image>
</vim-orchd>
</config>
</nc:edit-config>
</nc:rpc>

Example Output

  <ok/>
</rpc-reply>
VIM Orchestrator Deployment

This section lists the parameters that can be configured to activate and deactivate the deployment of VIM orchestrator, and get the VIM- ORCH deployment event notifications.

Operations

Deploying VIM Orchestrator

**Operation:** nsd:activate

**Namespace:** xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd</td>
<td>0</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
<tr>
<td>vim-orch</td>
<td>0</td>
<td>reference</td>
<td>Reference to VIM ORCH descriptor.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
    <nsd:nsd>sjc</nsd:nsd>
    <nsd:vim-orch>sjc-vim-orch</nsd:vim-orch>
  </nsd:activate>
</nc:rpc>
```

**Example Output**

```xml
  xmlns:nsd="" xmlns:vim="" xmlns:uas="" xmlns:common="" xmlns:oper="">
  <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1510937422-175985</transaction-id>
</rpc-reply>
```

Obtaining VIM- ORCH Deployment Event Notifications

**Operation:** get

**Namespace:** vim-orchestrator-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"

**Parameters:**
Parameter Name | Required | Type   | Description
--- | --- | --- | ---
instance-id | M | string | A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.
descriptor-id | M | string | Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.
transaction-id | M | string | Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.
operaion-type | M | choice string | Type of operation (same as the type of transaction)
status | M | choice string | Current deployment status (start, in-progress, error, or success).
steps-total | O | uint16 | Total number of steps required to be performed.
steps-completed | O | uint16 | Total number of steps completed.
version | O | string | Deployment version.

For more information on these notifications, see the UAS Notification Events.

Example Notifications

```xml
<.notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
    <eventTime>2017-11-18T18:25:39.780562+00:00</eventTime>
    <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usf-uas-common-oper">
        <instance-id>sjc-instance</instance-id>
        <descriptor-id>sjc</descriptor-id>
        <transaction-id>1511029539-662920</transaction-id>
        <operation-type>activate-ns-deployment</operation-type>
        <status>requested</status>
    </ns-deployment-event>
</notification>
```
VIM Orchestrator Deployment
Obtaining VIM-ORCH Instance Information

Operation: get


Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>M</td>
<td>choice</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

Example RPC

```xml
<nc:rpc message-id="urn:uuid:e4840216-12e4-44ed-b888-3321a8a7a711" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:get>
    <nc:filter type="subtree" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    </nc:filter>
  </nc:get>
</nc:rpc>
```

Example Output

```xml
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:e4840216-12e4-44ed-b888-3321a8a7a711">
  <data>
    <vim-orch xmlns="http://www.cisco.com/usf/nfv/vim-orch">
      <status>success</status>
      <steps-total>84</steps-total>
    </vim-orch>
  </data>
</rpc-reply>
```
Deactivating VIM Orchestrator

**Operation:** nsd: deactivate

**Namespace:** nsd:deactivate xmlns:nsd="http://www.cisco.com/usp/nfv/usps-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
<tr>
<td>vim-orch</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM ORCH descriptor.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:ec1d6011-7e8-4e6d-ad7b-f562d574c94a" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <nsd:nsd>sjc</nsd:nsd>
    <nsd:vim-orch>sjc-vim-orch</nsd:vim-orch>
  </nsd:deactivate>
</nc:rpc>
```

**Example Output**

```xml
  <transaction-id xmlns="http://www.cisco.com/usp/nfv/usps-nsds">1510937718-671857</transaction-id>
</rpc-reply>
```

Obtaining Notifications Sent During Deactivation of VIM- ORCH Deployment

**Operation:** get

**Namespace:** vim-orchestrator-deployment-event xmlns="http://www.cisco.com/usp/nfv/usps-uas-common-oper"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>----------</td>
<td>-----------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice string</td>
<td>Type of operation (same as type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

For more information on these notifications, see the UAS Notification Events.

Example Notifications

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-17T16:55:18.841218+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <transaction-id>1510937718-671857</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>requested</status>
  </ns-deployment-event>
</notification>
```

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-17T16:55:18.984458+00:00</eventTime>
  <vim-orchestrator-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-vim-orch</instance-id>
    <transaction-id>1510937718-671857/1510937718-893572</transaction-id>
    <operation-type>deactivate-vim-orch-deployment</operation-type>
    <status>requested</status>
  </vim-orchestrator-deployment-event>
</notification>
```

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-17T16:55:19.111137+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <transaction-id>1510937718-671857</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>in-progress</status>
  </ns-deployment-event>
</notification>
```

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-17T16:55:19.380541+00:00</eventTime>
  <vim-orchestrator-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-vim-orch</instance-id>
    <transaction-id>1510937718-671857/1510937718-893572</transaction-id>
    <operation-type>deactivate-vim-orch-deployment</operation-type>
    <status>in-progress</status>
  </vim-orchestrator-deployment-event>
</notification>
```
VIM Configuration

The UAS provides VIM Descriptor (VIMD) configuration to deploy VIM such as OpenStack. The UAS provides a generic set of APIs for VIM deployment. It uses different plugins for different types of VIM deployments.

The VIMD configuration comprises the following:

- Network Security Descriptor (NSD)
  - VIM descriptor (vimd)
    - NFVI node
    - NFVI Point-of-Presence descriptor (nfvi-popd)
  - VLAN pool
  - Allocation pool
  - Provisioning Network

Networking configuration parameters can be configured for various networks that are used for deployment and maintainence of the VIM operational state.

VLAN configuration specifies the VLAN range that is used to create neutron networks of type VLAN. The corresponding VLAN is configured on the Switch and Spine for enabling trunked traffic from VMs.

Allocation pool configuration specifies the IP address range that is used by VIM-ORCH while orchestrating the VIM deployment. These IP addresses are allocated to NFVI nodes for various network types as part of VIM installation.

Operations

Configuring VIMD

**Operation:** edit-config

**Namespace:** vimd xmlns="http://www.cisco.com/usp/nfv/usp-vim"

**VIMD Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vim-orch-id</td>
<td>M</td>
<td>string</td>
<td>Name of VIM-ORCH (OSPD) deployment that is used to deploy the VIM.</td>
</tr>
<tr>
<td>nfvi-pop-id</td>
<td>M</td>
<td>string</td>
<td>Name of the NFVI Point-of-Presence that is used to deploy the VIM.</td>
</tr>
</tbody>
</table>

**NOTE:** This is the same nfvi-pop-id descriptor provided as part of VIM-ORCH descriptor associated with this VIM deployment.
### VIM Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| role           | M        | choice-string | Each identified nfvi-node is associated with a role that is installed and instantiated as part of the VIM deployment. Possible roles are as follows:  
- vim-orch: NFVI node that is used for hosting VIM-ORCH VM  
- vim-compute: NFVI node that is used for hosting compute services.  
- vim-controller: NFVI node that is used for hosting controller services.  
- vim-osd-compute: NFVI node that is used for hosting both compute as well as storage services.  
- vim-ceph-storage: NFVI node that is used for hosting ceph storage service only.  
**NOTE:** For a successful VIM deployment, a minimum of 3 nodes for vim-controller, 3 nodes for vim-osd-compute and 1 node for vim-compute are required. |
| nfvi-nodes      | M        | string      | Name of NFVI node that needs to be associated/used for the VIM deployment. |

**Networking Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dns</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Domain name server IP address used for name resolution.</td>
</tr>
<tr>
<td>ntp</td>
<td>M</td>
<td>inet:ip-address</td>
<td>Network Time Protocol server that is used for time synchronization across all NFVI nodes that are used in VIM deployment.</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>----------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>network-types</td>
<td>M</td>
<td>choice string</td>
<td>The following network types are used by VIM depending on the type of traffic and node responsibility:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• internal-api: Internal API used for private OpenStack Traffic</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• tenant: Tenant Network Traffic used for VXLAN over VLAN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• storage: Public Storage Access. For example, Nova/Glance &gt; Ceph</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• storage-mgmt: Private Storage Access – Ceph background cluster/replication</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• external: External Networking Access - Public API Access</td>
</tr>
<tr>
<td>ip-prefix</td>
<td>M</td>
<td>inet:ip-address</td>
<td>CIDR used for the selected network.</td>
</tr>
<tr>
<td>vlan-id</td>
<td>M</td>
<td>uint16</td>
<td>Segmentation ID used for the selected network.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>NOTE:</strong> If vlan-tag is set to true, then this attribute is required.</td>
</tr>
</tbody>
</table>

**IP Allocation Pool Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>M</td>
<td>uint16</td>
<td>Start IP address of the range to be used for allocation (inclusive).</td>
</tr>
<tr>
<td>end</td>
<td>M</td>
<td>uint16</td>
<td>End IP address of the range to be used for allocation (inclusive).</td>
</tr>
</tbody>
</table>

**VLAN Pool Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>M</td>
<td>uint16</td>
<td>Start VLAN-ID of the range to be used for allocation (inclusive).</td>
</tr>
<tr>
<td>end</td>
<td>M</td>
<td>uint16</td>
<td>End VLAN-ID of the range to be used for allocation (inclusive).</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:af7c95b7-2e02-4dca-ad39-55572b898951"
  xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
```

74
<nc:target>
  <nc:running/>
</nc:target>

<config>
  <nsd xmlns="http://www.cisco.com/usp/nfv/nsds">
    <nsd-id>sjc</nsd-id>
    <vim>overc</vim>
  </nsd>

  <vimd xmlns="http://www.cisco.com/usp/nfv/vim">
    <vim-id>overc</vim-id>
    <vim-orch-id>underc</vim-orch-id>
    <nfvi-pop-id>sjc-pop</nfvi-pop-id>
    <nfvi-nodes>
      <id>node_1</id>
      <role>vim-compute</role>
    </nfvi-nodes>
    <nfvi-nodes>
      <id>node_2</id>
      <role>vim-controller</role>
    </nfvi-nodes>
    <nfvi-nodes>
      <id>node_3</id>
      <role>vim-controller</role>
    </nfvi-nodes>
    <nfvi-nodes>
      <id>node_4</id>
      <role>vim-osd-compute</role>
    </nfvi-nodes>
    <nfvi-nodes>
      <id>node_5</id>
      <role>vim-osd-compute</role>
    </nfvi-nodes>
    <nfvi-nodes>
      <id>node_6</id>
      <role>vim-osd-compute</role>
    </nfvi-nodes>

  </vimd>

  <networking>
    <dns>171.70.168.183</dns>
    <ntp>172.24.167.109</ntp>
    <network-types>
      <type>internal-api</type>
      <ip-prefix>11.120.0.0/24</ip-prefix>
      <vlan-id>20</vlan-id>
      <allocation-pool>
        <start>11.120.0.10</start>
        <end>11.120.0.200</end>
      </allocation-pool>
    </network-types>
    <network-types>
      <type>tenant</type>
      <ip-prefix>11.117.0.0/24</ip-prefix>
      <vlan-id>17</vlan-id>
      <allocation-pool>
        <start>11.117.0.10</start>
        <end>11.117.0.200</end>
      </allocation-pool>
    </network-types>
    <network-types>
      <type>storage</type>
      <ip-prefix>11.118.0.0/24</ip-prefix>
      <vlan-id>18</vlan-id>
      <allocation-pool>
        <start>11.118.0.10</start>
        <end>11.118.0.200</end>
      </allocation-pool>
    </network-types>
    <network-types>
      <type>storage-mgmt</type>
      <ip-prefix>11.119.0.0/24</ip-prefix>
    </network-types>
  </networking>
</config>
<vlan-id>19</vlan-id>
<allocation-pool>
  <start>11.119.0.10</start>
  <end>11.119.0.200</end>
</allocation-pool>
</network-types>
<network-types>
  <type>external</type>
  <ip-prefix>172.21.203.0/24</ip-prefix>
  <vlan-id>101</vlan-id>
  <allocation-pool>
    <start>172.21.203.120</start>
    <end>172.21.203.150</end>
  </allocation-pool>
  <default-route>172.21.203.1</default-route>
</network-types>
<vlan-pool>
  <start>1001</start>
  <end>1150</end>
</vlan-pool>
</networking>
</vimd>
<nfvi-popd xmlns="http://www.cisco.com/usp/nfv/usp-vim-orch">
  <nfvi-pop-id>sjc-pop</nfvi-pop-id>
  <deployment-flavor>ucs-1-vnf</deployment-flavor>
  <nfvi-node>
    <nfvi-node-id>node_1</nfvi-node-id>
    <physical-server-manager>
      <ip-address>192.100.1.2</ip-address>
      <login-credential>cimc</login-credential>
    </physical-server-manager>
  </nfvi-node>
  <nfvi-node>
    <nfvi-node-id>node_2</nfvi-node-id>
    <physical-server-manager>
      <ip-address>192.100.1.3</ip-address>
      <login-credential>cimc</login-credential>
    </physical-server-manager>
  </nfvi-node>
  <nfvi-node>
    <nfvi-node-id>node_3</nfvi-node-id>
    <physical-server-manager>
      <ip-address>192.100.1.4</ip-address>
      <login-credential>cimc</login-credential>
    </physical-server-manager>
  </nfvi-node>
  <nfvi-node>
    <nfvi-node-id>node_4</nfvi-node-id>
    <physical-server-manager>
      <ip-address>192.100.1.5</ip-address>
      <login-credential>cimc</login-credential>
    </physical-server-manager>
  </nfvi-node>
  <nfvi-node>
    <nfvi-node-id>node_5</nfvi-node-id>
    <physical-server-manager>
      <ip-address>192.100.1.6</ip-address>
      <login-credential>cimc</login-credential>
    </physical-server-manager>
  </nfvi-node>
  <nfvi-node>
    <nfvi-node-id>node_6</nfvi-node-id>
    <physical-server-manager>
      <ip-address>192.100.1.7</ip-address>
      <login-credential>cimc</login-credential>
    </physical-server-manager>
  </nfvi-node>
</nfvi-popd>
</config>
Example Output

```xml
  <ok/>
</rpc-reply>
```
VIM Deployment

This section lists the parameters that can be configured to activate and deactivate the deployment of VIM, and also to receive the VIM deployment event notifications.

Operations

Deploying VIM

**Operation:** nsd:activate

**Namespace:** xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd</td>
<td>0</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
<tr>
<td>vim-orch</td>
<td>0</td>
<td>reference</td>
<td>Reference to VIM ORCH descriptor.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:b1c07c6a-a423-4bde-87ab-325772b74057" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
        <nsd:nsd>sjc</nsd:nsd>
        <nsd:vim>overc</nsd:vim>
    </nsd:activate>
</nc:rpc>
```

**Example Output**

```xml
    <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1512559306-003767</transaction-id>
</rpc-reply>
```

Obtaining VIM Deployment Event Notifications

**Operation:** get

**Namespace:** vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"

**Parameters:**
<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice string</td>
<td>Type of operation (same as type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

For more information on these notifications, see the UAS Notification Events.

Example Notifications

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T11:21:46.041655+00:00</eventTime>
  <ns:deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <descriptor-id>sjc</descriptor-id>
    <transaction-id>1512559306-003767</transaction-id>
    <operation-type>activate-ns-deployment</operation-type>
    <status>requested</status>
  </ns:deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T11:21:46.084914+00:00</eventTime>
  <vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc:overc</instance-id>
    <descriptor-id>overc</descriptor-id>
    <transaction-id>1512559306-003767</transaction-id>
    <operation-type>activate-vim-deployment</operation-type>
    <status>requested</status>
  </vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T11:21:46.149838+00:00</eventTime>
  <ns:deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <descriptor-id>sjc</descriptor-id>
    <transaction-id>1512559306-003767</transaction-id>
  </ns:deployment-event>
</notification>
```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:21:46.233761+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:21:46.685003+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:16.490033+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:16.571842+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:16.685003+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:16.798921+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:16.809649+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:16.821255+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:16.980949+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:17.122332+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:17.124711+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T11:23:17.124873+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usf/nfv/usf-uas-common-oper">
<instance-id>a:cover</instance-id>
<descriptor-id>cover</descriptor-id>
<transaction-id>1512559306-003767/1512559306-061122</transaction-id>
<operation-type>activate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>
Obtaining VIM Instance Information

**Operation:** get

**Namespace:** vimr xmlns="http://www.cisco.com/usf/nfv/vim-uas-oper"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>
Example RPC

```xml
<nc:rpc message-id="urn:uuid:196d4145-bdc5-4537-9aa0-655a3dd597b2" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:get>
    <nc:filter type="subtree">
    </nc:filter>
    <data></nc:get>
  </nc:rpc>
```

Example Output

```xml
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:196d4145-bdc5-4537-9aa0-655a3dd597b2">
  <vimr xmlns="http://www.cisco.com/usp/nfv/nfvi-oper">
    <status>success</status>
    <steps-total>16</steps-total>
    <steps-completed>16</steps-completed>
    <version>Red Hat OpenStack Platform release 10.0 (Newton)</version>
  </vimr>
  <data></data>
</rpc-reply>
```

Obtaining NFVI Node Instance Information

**Operation:** get

**Namespace:** nfvi-node xmlns="http://www.cisco.com/usp/nfv/nfvi-oper" /

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>M</td>
<td>string</td>
<td>Associate the NFVI node identifier defined in NFVI-POPd that will be used to host VIM Orchestrator VM.</td>
</tr>
<tr>
<td>ip-address</td>
<td>M</td>
<td>inet:ip-address</td>
<td>IP address to access the host OS of the NFVI node (SSH).</td>
</tr>
<tr>
<td>login-credential</td>
<td>M</td>
<td>secure-token</td>
<td>Authentication information to access the host OS of the NFVI node (SSH).</td>
</tr>
</tbody>
</table>

Example RPC

```xml
<nc:rpc message-id="urn:uuid:196d4145-bdc5-4537-9aa0-655a3dd597b2" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:get>
    <nc:filter type="subtree">
    </nc:filter>
    <data></nc:get>
  </nc:rpc>
```

Example Output

```xml
  <vimr xmlns="http://www.cisco.com/usp/nfv/nfvi-oper">
    <status>success</status>
    <steps-total>16</steps-total>
    <steps-completed>16</steps-completed>
    <version>Red Hat OpenStack Platform release 10.0 (Newton)</version>
  </vimr>
  <data></data>
</rpc-reply>
```
<nfvi-node xmlns="http://www.cisco.com/usp/nfv/usp-nfvi-oper">
  <nfvi-node-id>node_1</nfvi-node-id>
  <status>up</status>
  <role>vim-compute</role>
  <physical-server-manager>
    <firmware-version>2.0(10e)</firmware-version>
    <ip-address>192.100.1.2</ip-address>
    <bios-version>C240M4.2.0.10e.0.0620162114</bios-version>
  </physical-server-manager>
  <storage-device>
    <id>/dev/sda</id>
    <size>1143845</size>
    <is-journal>false</is-journal>
  </storage-device>
  <network-device>
    <id>eno1</id>
  </network-device>
  <network-device>
    <id>eno2</id>
  </network-device>
  <network-device>
    <id>enp10s0f0</id>
    <physnet-id>phys_pcie1_0</physnet-id>
  </network-device>
  <network-device>
    <id>enp10s0f1</id>
    <physnet-id>phys_pcie1_1</physnet-id>
  </network-device>
  <network-device>
    <id>enp133s0f0</id>
    <physnet-id>phys_pcie4_0</physnet-id>
  </network-device>
  <network-device>
    <id>enp133s0f1</id>
    <physnet-id>phys_pcie4_1</physnet-id>
  </network-device>
  <network-device>
    <id>enp6s0</id>
  </network-device>
  <network-device>
    <id>enp7s0</id>
  </network-device>
</nfvi-node>

Deactivating VIM

**Operation**: nsd:deactivate

**Namespace**: nsd:deactivate xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters**:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
<tr>
<td>vim-orch</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM ORCH descriptor.</td>
</tr>
</tbody>
</table>
Example RPC

```xml
<nc:rpc message-id="urn:uuid:79d8620c-7f2e-4177-b249-c64e6cfa75a7" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <nsd:nsd>sjc</nsd:nsd>
    <nsd:vim>overc</nsd:vim>
  </nsd:deactivate>
</nc:rpc>
```

Example Output

```xml
  <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1510937718-671857</transaction-id>
</rpc-reply>
```

Obtaining Notifications Sent During Deactivation of VIM Deployment

**Operation:** get

**Namespace:** `vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usps-common-oper"`

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice string</td>
<td>Type of operation (same as type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

For more information on these notifications, see the UAS Notification Events.
Example Notifications

---NOTIFICATION---
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.134662+00:00</eventTime>
<ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203</transaction-id>
<operation-type>deactivate-ns-deployment</operation-type>
<status>requested</status>
</ns-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.217999+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203/1512580815-176728</transaction-id>
<operation-type>deactivate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.30663+00:00</eventTime>
<ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203</transaction-id>
<operation-type>deactivate-ns-deployment</operation-type>
<status>in-progress</status>
</ns-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.386493+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:cover</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203/1512580815-176728</transaction-id>
<operation-type>deactivate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.481178+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:cover</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203/1512580815-176728</transaction-id>
<operation-type>deactivate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.589042+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:cover</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203/1512580815-176728</transaction-id>
<operation-type>deactivate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.676728+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:cover</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203/1512580815-176728</transaction-id>
<operation-type>deactivate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.774391+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:cover</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203/1512580815-176728</transaction-id>
<operation-type>deactivate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:20:15.86708+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:cover</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203/1512580815-176728</transaction-id>
<operation-type>deactivate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T17:22:29.376443+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:cover</instance-id>
<descriptor-id>overc</descriptor-id>
<transaction-id>1512580815-086203/1512580815-176728</transaction-id>
<operation-type>deactivate-vim-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T17:22:29.482085+00:00</eventTime>
  <vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usf-usas-common-oper">
    <instance-id>sjc:overc</instance-id>
    <descriptor-id>overc</descriptor-id>
    <transaction-id>1512580815-086203/1512580815-176728</transaction-id>
    <operation-type>deactivate-vim-deployment</operation-type>
    <status>success</status>
  </vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T17:22:29.540793+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usf-usas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <descriptor-id>sjc</descriptor-id>
    <transaction-id>1512580815-086203</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>success</status>
  </ns-deployment-event>
</notification>

--NOTIFICATION--
VNF Package Configuration

VNF package contains the software images and the configuration files. This VNF software package is based on USP packaging manifest to identify the images for different VDUs automatically.

Typically, VNF packages are defined within NSD. Hence, a complete NSD can be activated or a specific VNF package within NSD can be activated.

Once the VNF package is on boarded successfully, images are created on VIM and local vnf-package-instance is created. The UAS removes the package content from local storage and keeps metadata to have mapping between on-boarded artifacts to VDU image and the configuration files. The VDU refers the VNF package and the UAS automatically populates correct resource based on this manifest information.

The following figure depicts the UAS package management workflow.

UAS Package Management Workflow

www.websequencediagrams.com
Operations

Configuring VNF Package

**Operation:** edit-config

**Namespace:** vnf-packaged xmlns="http://www.cisco.com/usp/nfv/usp-uas-common"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-package-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this package instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>name</td>
<td>O</td>
<td>string</td>
<td>Name of the VNF package to be onboarded.</td>
</tr>
<tr>
<td>checksum</td>
<td>O</td>
<td>string</td>
<td>MD5 checksum of the onboarded VNF package.</td>
</tr>
<tr>
<td>location</td>
<td>M</td>
<td>url string</td>
<td>URL location to download this package.</td>
</tr>
<tr>
<td>validate-signature</td>
<td>O</td>
<td>bool</td>
<td>Specifies whether the Digital signature validation functionality is enabled or disabled. Default value is “true”.</td>
</tr>
<tr>
<td>vdu-image</td>
<td>O</td>
<td>string</td>
<td>VDU manifest to map images. This is a place holder to onboard third-party images. For Ultra-M UAS automatically drives this binding and nothing is required explicitly.</td>
</tr>
<tr>
<td>configuration</td>
<td>O</td>
<td>array of configurations</td>
<td>Set of Day-O / Day-N configuration files which can be shared across VNFs during deployment.</td>
</tr>
</tbody>
</table>

Example RPC

```
<nc:rpc message-id="urn:uuid:e2f5e6f5-9aab-4941-b19d-e2f9ebe17e2f"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
      <vnf-packaged xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
        <vnf-package-id>usp_5_7</vnf-package-id>
        <checksum>3e76b110843d2167befe0b9bd350589c</checksum>
        <location>http://10.23.252.162/firmwares/usp_5_7_M0-2045.iso</location>
        <configuration>
          <name>staros</name>
          <external-url>http://10.23.252.162/system.cfg</external-url>
        </configuration>
      </vnf-packaged>
    </config>
  </nc:edit-config>
</nc:rpc>
```
Example Output

```xml
  <ok/>
</rpc-reply>
```
VNF Package Deployment

This section describes the parameters that can be used to activate or deactivate the VNF package within NSD.

Operations

Activating VNF Package

Operation: nsd:activate


Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd-id</td>
<td>M</td>
<td>string (1-16)</td>
<td>NSD identifier</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>NSD version</td>
</tr>
<tr>
<td>vim-orch</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM ORCH descriptor.</td>
</tr>
<tr>
<td>vim-orch-identity</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM descriptor used to provide vim-orchestrator identity for deployment.</td>
</tr>
<tr>
<td>vim</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM descriptor.</td>
</tr>
<tr>
<td>fmd</td>
<td>O</td>
<td>reference</td>
<td>Reference to Fault Management Descriptor.</td>
</tr>
<tr>
<td>vim-artifact</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM artifact descriptor used to provision the set of parameters on VIM to on board VNFs.</td>
</tr>
<tr>
<td>vim-identity</td>
<td>O*</td>
<td>reference</td>
<td>Reference to VIM identity used to manage VIM resources. It is mandatory to deploy VNFs.</td>
</tr>
<tr>
<td>vnf-package</td>
<td>O</td>
<td>array of references</td>
<td>This is the list of references to VNF package descriptors. These VNF packages are referred under VDU for software images and day-zero/day-n configuration files.</td>
</tr>
<tr>
<td>vld</td>
<td>O</td>
<td>array of references</td>
<td>List of Virtual Link Descriptors (VLD) to be used to connect different VNFs within Network Service domain.</td>
</tr>
<tr>
<td>vnfd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to VNFDs used to build this NSD.</td>
</tr>
<tr>
<td>nsd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.
Example RPC

```xml
<nc:rpc message-id="urn:uuid:bac690a2-08af-4c9f-8765-3c907d6e12ba"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
   <nsd:nsd>sjc-autoit</nsd:nsd>
   <nsd:vnf-package>usp_5_7</nsd:vnf-package>
 </nsd:activate>
</nc:rpc>
```

Example Output

```xml
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:bac690a2-08af-4c9f-8765-3c907d6e12ba"
 xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
</rpc-reply>
```

Obtaining VNF Package Deployment Event Notifications

**Operation:** get

**Namespace:** vnf-package-deployment-event xmlns='http://www.cisco.com/usp/nfv/usp-uas-common-oper'

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice string</td>
<td>Type of operation (same as the type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

For more information on these notifications, see the UAS Notification Events.

**Example Notifications**

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0" eventTime="2017-11-04T00:26:39.1485+00:00">
 <ns-deployment-event xmlns='http://www.cisco.com/usp/nfv/usp-uas-common-oper'>
```

91
VNF Package Deployment

Obtaining VNF Package Operational Instance Information

Operation: get


VNF Package Operational Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-package-id</td>
<td>M</td>
<td>string</td>
<td>VNF Package Instance ID. This is built using the version pulled from actual build-manifest file.</td>
</tr>
</tbody>
</table>
VNF Package Deployment

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-package</td>
<td>M</td>
<td>reference</td>
<td>Reference to VNF Package Descriptor used for onboarding.</td>
</tr>
<tr>
<td>signature</td>
<td>O</td>
<td>string</td>
<td>VN package signature</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Software version at the VNF package level.</td>
</tr>
<tr>
<td>image</td>
<td>O</td>
<td>array of VDU image instance</td>
<td>List of VDU images.</td>
</tr>
<tr>
<td>configuration</td>
<td>O</td>
<td>array of configurations</td>
<td>List of configuration files on boarded as part of this package.</td>
</tr>
</tbody>
</table>

**VDU Image Instance Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image-id</td>
<td>M</td>
<td>string</td>
<td>Unique image ID for this VDU.</td>
</tr>
<tr>
<td>image-uri</td>
<td>M</td>
<td>string</td>
<td>Image location within the VNF package.</td>
</tr>
<tr>
<td>vim-id</td>
<td>M</td>
<td>string</td>
<td>Image UUID as onboarded on VIM.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Software version for this VDU image.</td>
</tr>
<tr>
<td>disk-format</td>
<td>M</td>
<td>string</td>
<td>Disk format for the image. Most of the USP VDU images are in 'qcow2' format.</td>
</tr>
</tbody>
</table>

**VDU Configuration Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>configuration-id</td>
<td>M</td>
<td>string</td>
<td>Unique configuration ID.</td>
</tr>
<tr>
<td>data-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID of this configuration as stored in the backend database.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:76a6c455-38bf-456d-b3a4-6cdef85ca06"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:get>
    <nc:filter type="subtree">
        <vnf-package-id>5.7.M0-2045</vnf-package-id>
      </vnf-packager>
    </nc:filter>
  </nc:get>
</nc:rpc>
```
Example Output

```xml
  <data>
    <vnf packager xmlns="http://www.cisco.com/usp/nfv/usp-vnf-packager">
      <configuration>
        <package xmlns:pub="2045" id="pub-2045">
          <data/>
        </package>
      </configuration>
    </vnf>
  </data>
</rpc-reply>
```
Deactivating VNF Package

**Operation:** nsd:deactivate

**Namespace:** nsd:deactivate xmlns:nsd="http://www.cisco.com/usp/nfv/nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd</td>
<td>O</td>
<td>array of references</td>
<td>List of references to nested NSDs.</td>
</tr>
<tr>
<td>vnf-package</td>
<td>O</td>
<td>array of references</td>
<td>This is the list of references to VNF package descriptors. These VNF packages are referred under VDU for software images and day-zero/day-n configuration files.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
        <nsd:nsd>sjc-autoit</nsd:nsd>
        <nsd:vnf-package>usp_5_7</nsd:vnf-package>
    </nsd:deactivate>
</nc:rpc>
```

**Example Output**

```xml
    <transaction-id xmlns="http://www.cisco.com/usp/nfv/nsds">1509763741-816325</transaction-id>
</rpc-reply>
```

Obtaining Notifications Sent During Deactivation of VNF Package

**Operation:** get

**Namespace:** vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/uas-common-oper"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice string</td>
<td>Type of operation (same as type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

For more information on these notifications, see the UAS Notification Events.

Example Notifications

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-04T02:49:01.834933+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>jgc-autoit-instance/instance-id>
    <transaction-id>1509763741-816325</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>requested</status>
  </ns-deployment-event>
</notification>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-04T02:49:01.906268+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>jgc-autoit-instance/instance-id>
    <transaction-id>1509763741-816325</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>in-progress</status>
  </ns-deployment-event>
</notification>
```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-04T02:49:13.768601+00:00</eventTime>
  <vnf-package-deployment-event xmlns="http://www.cisco.com/usf/usf-uas-common-oper">
    <instance-id>usp_5_7-instance</instance-id>
    <descriptor-id>/descriptor-id>
    <transaction-id>1509763741-816325</transaction-id>
    <operation-type>deactivate-vnf-package</operation-type>
    <status>success</status>
  </vnf-package-deployment-event>
</notification>
Virtual Link Configuration

Virtual links provide connectivity among VNFs within NSD or among VNFC instances within VNFD. Hence, the following two types of virtual links are defined:

- **External Virtual Link**: These are the virtual links defined at NSD level and used to connect different VNFs within NSD.
- **Internal Virtual Link**: These are virtual links defined within VNFD and provide local networking within VNF domain.

The UAS is meant to onboard only Ultra VNFs hence named VLDs are defined at NSD level. These VLDs are used as virtual link within connection points in VNFD.

The following are the different types of VLDs defined at NSD level:

- **Orchestration**: This virtual link is used for orchestration purposes. The UAS, VNF-EM and VNFM use this virtual link to perform lifecycle management.
- **Management**: This virtual link is used in Ultra VNFs to expose external connection point mainly to connect VNF outside, for example, associating floating IP.
- **Service**: This is an external network which VNF can use to communicate with, for example, service network for Ultra UGP platform or Policy network for Ultra CPS platform.

The following figure shows how these networks are used. Management network is optional and orchestration network is connected to router and associated with a floating IP. But it is recommended to have orchestration network as non-routable to isolate internal traffic within service domain.

![Virtual Link Diagram](image_url)

Typically, internal virtual links are OAM networks used to form clustering within VNF itself. For example, VPC DI internal network. These networks are available within the VNF domain.

**Virtual Link Descriptor Configuration Parameters**

**Operation**: edit-config
**Virtual Link Configuration**

**Namespace:** nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vld-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID for the virtual link.</td>
</tr>
<tr>
<td>vl-type</td>
<td>M</td>
<td>choice string</td>
<td>Virtual link type. It can be one of the following values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• orchestration</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• management</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• service</td>
</tr>
<tr>
<td>nw-type</td>
<td>M</td>
<td>choice string</td>
<td>Network type can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• descriptor: UAS uses the referred descriptor to create network on VIM.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vim-resource: UAS assumes the network is pre-created on network.</td>
</tr>
<tr>
<td>network</td>
<td>M*</td>
<td>reference</td>
<td>This is required when nw-type is selected as “descriptor” and it should point to a valid network descriptor from the catalog.</td>
</tr>
</tbody>
</table>
| network-instance | M*       | reference | Reference to network instance which provided details about the network pre-created on VIM. This is required when ‘nw-type” is selected as “vim-resource”.

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

**Operations**

**Configuring Network Descriptor**

**Operation:** edit-config

**Namespace:** network xmlns="http://www.cisco.com/usp/nfv/usps-uas-common"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>network-id</td>
<td>M</td>
<td>string</td>
<td>Network Descriptor ID.</td>
</tr>
<tr>
<td>ip-prefix</td>
<td>M</td>
<td>inet:ip-prefix</td>
<td>IP subnet for this network, defined in CIDR format.</td>
</tr>
</tbody>
</table>
### Virtual Link Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| type           | M        | choice string | Network type can be one of the following:  
- vlan: VLAN Based virtio network  
- sriov-flat: SRIOV Flat network  
- sriov-vlan: SRIOV VLAN-based network  
- internal: Internal networks. These networks are not created on VIM but passed to backend and networks are internal to the VM.  
- local: Host local network  
(Default value is “vlan”). |
| dhcp           | O        | bool       | Enable or disable DHCP. Default is “true”. |
| gateway        | O        | inet:ip-address | Gateway IP address if applicable. |
| vlan-tag       | O        | bool       | Applicable only when “type” is “sriov-flat”. Set it true if VM is supposed to tag VLAN on SRIOV flat network. |
| vlan           | O        | int (0..4096) | VLAN ID for this network. It is a mandatory configuration when vlan-tag is enabled. |
| ip-allocation-pool | O*      | array of IP allocation pool | List of IP Allocation Pool descriptors. This is required only when DHCP is disabled and you want UAS to manage the allocation of IPs from the specific pool. Hence, it is a mandatory parameter if DHCP is set to False. |

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

#### Example RPC

```xml
<nc:rpc message-id="urn:uuid:fa4fc696-0c89-454e-b438-ea8b0ac53eb3"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
      <network xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
        <network-id>orchestration</network-id>
        <ip-prefix>10.10.10.0/24</ip-prefix>
        <gateway>10.10.10.1</gateway>
        <router>main</router>
      </network>
      <network xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
        <network-id>di-internal</network-id>
        <ip-prefix>192.168.2.0/24</ip-prefix>
      </network>
      <network xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
        <network-id>service-network</network-id>
        <ip-prefix>88.88.88.0/24</ip-prefix>
      </network>
    </config>
  </nc:edit-config>
</nc:rpc>
```
Example Output

```xml
  <ok/>
</rpc-reply>
```

Configuring Network Instance

**Operation:** edit-config

**Namespace:** network-instance xmlns="http://www.cisco.com/usp/nfv/usp-uas-common"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>network-id</td>
<td>M</td>
<td>string</td>
<td>Network Name or UUID as created on VIM.</td>
</tr>
<tr>
<td>ip-prefix</td>
<td>M</td>
<td>inet:ip-prefix</td>
<td>IP subnet for this network, defined in CIDR format.</td>
</tr>
<tr>
<td>type</td>
<td>O</td>
<td>choice string</td>
<td>Network type can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vlan: VLAN Based virtio network</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• sriov-flat: SRIOV Flat network</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• sriov-vlan: SRIOV VLAN-based network</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• internal: Internal networks. These networks are not created on VIM but passed to backend and networks are internal to the VM.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• local: Host local network</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Default value is “vlan”.</td>
</tr>
<tr>
<td>dhcp</td>
<td>O</td>
<td>bool</td>
<td>Enable or disable DHCP. Default is “true”.</td>
</tr>
<tr>
<td>gateway</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Gateway IP address if applicable.</td>
</tr>
<tr>
<td>vlan-tag</td>
<td>O</td>
<td>bool</td>
<td>Applicable only when “type” is “sriov-flat”. Set it true if VM is supposed to tag VLAN on SRIOV flat network. Default is “false”.</td>
</tr>
<tr>
<td>vlan</td>
<td>O</td>
<td>int (0..4096)</td>
<td>VLAN ID for this network. It is a mandatory configuration when vlan-tag is enabled. By default VIM auto-assigns the VLAN ID for this network. You can also set this value if you want a specific VLAN ID to be used on this network.</td>
</tr>
</tbody>
</table>
Virtual Link Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-allocation-pool</td>
<td>O*</td>
<td>array of IP allocation pool</td>
<td>List of IP Allocation Pool descriptors. This is required only when DHCP is disabled and you want UAS to manage the allocation of IPs from the specific pool. Hence, it is a mandatory parameter if DHCP is set to False.</td>
</tr>
<tr>
<td>physnet</td>
<td>O*</td>
<td>string</td>
<td>Physnet name in case network type is based on SRIOV - &quot;sriov-flat&quot; or &quot;sriov-vlan&quot;.</td>
</tr>
<tr>
<td>router</td>
<td>O</td>
<td>string</td>
<td>Name of the router if you want this network to have external connectivity. The UAS creates an interface on router for this network hence this network becomes “routable”.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

Example RPC

```xml
<n:rpc message-id="urn:uuid:b0a509c7-9e67-475b-9c1d-bd7d45b28c1a" xmlns:n="urn:ietf:params:xml:ns:netconf:base:1.0">
  <n:edit-config>
    <n:target>
      <n:running/>
    </n:target>
    <config>
        <network-id>sjc-pgw-mgmt</network-id>
        <ip-prefix>100.100.100.0/24</ip-prefix>
        <gateway>100.100.100.1</gateway>
      </network-instance>
    </config>
  </n:edit-config>
</n:rpc>
```

Example Output

```xml
  <ok/>
</rpc-reply>
```
VIM Identity Configuration

VIM identity contains the necessary information to access VIM. The UAS uses this VIM identity for the following two purposes:

- Access VIM to allocate/provision resources
- Pass on the VIM identity to VNFM to onboard VNF on a specific VIM. For example, in case of Cisco ESC as VNFM, this VIM identity serves the purpose of VIM connector. You can use different VIM identities to onboard different VNFs on VIM. This provides user/project/tenant level isolation in terms of security and access.

Operations

Configuring VIM Identity

Operation: edit-config


Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vim-id</td>
<td>M</td>
<td>string (1..16)</td>
<td>Unique ID for this VIM identity.</td>
</tr>
<tr>
<td>api-version</td>
<td>O</td>
<td>choice string</td>
<td>API version. For example, on OpenStack, it should be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• “v2”: OS Identity API Version-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• “v3”: OS Identity API Version-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Default value is OS Identity API Version-2.</td>
</tr>
<tr>
<td>auth-url</td>
<td>M</td>
<td>url</td>
<td>Authentication URL</td>
</tr>
<tr>
<td>user</td>
<td>M</td>
<td>secure-token</td>
<td>User credential</td>
</tr>
<tr>
<td>tenant</td>
<td>M</td>
<td>string</td>
<td>Tenant name on this VIM.</td>
</tr>
<tr>
<td>domain-name</td>
<td>O*</td>
<td>string</td>
<td>Domain name on VIM. This is required only if VIM type is OpenStack and API version is selected as V3.</td>
</tr>
<tr>
<td>project</td>
<td>O*</td>
<td>string</td>
<td>Project Name. This is required only if VIM type is OpenStack and API version is selected as V3.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.
Example RPC

<nc:rpc message-id="urn:uuid:5c9e7ee5-79b6-416d-a096-6ebc32d28709"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:running/>
    <config>
      <vim xmlns="http://www.cisco.com/usp/nfv/usps-common">
        <vim-id>vim1</vim-id>
        <api-version>v2</api-version>
        <auth-url>http://172.21.201.213:5000/v2.0</auth-url>
        <user>openstack</user>
        <tenant>akssingh</tenant>
      </vim>
    </config>
  </nc:edit-config>
</nc:rpc>

Example Output

<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:5c9e7ee5-79b6-416d-a096-6ebc32d28709"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
VIM Artifacts Configuration

VIM Artifact descriptor contains a set of parameters to be provisioned on VIM before VNFs are on boarded. The actual operations are performed at each descriptor level; hence the notifications events are generated at each descriptor.

Operations

Configuring VIM Artifacts

Operation: edit-config


VIM Artifacts Descriptor Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vim-artifact-id</td>
<td>M</td>
<td>string</td>
<td>Artifact descriptor ID, uniquely identifies the descriptor within the service.</td>
</tr>
<tr>
<td>description</td>
<td>O</td>
<td>string</td>
<td>User augmented notes/descriptor for this.</td>
</tr>
<tr>
<td>router-id</td>
<td>O</td>
<td>array of references</td>
<td>List of references to router descriptors to be created.</td>
</tr>
<tr>
<td>vnf-rack-id</td>
<td>O</td>
<td>array of references</td>
<td>List of references to VNF rack descriptors to be created.</td>
</tr>
<tr>
<td>tenant</td>
<td>O</td>
<td>array of references</td>
<td>List of references to tenant descriptors.</td>
</tr>
</tbody>
</table>

Router Descriptor Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>router-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier to router descriptor.</td>
</tr>
<tr>
<td>external-network</td>
<td>M</td>
<td>string</td>
<td>External network name to associate this router.</td>
</tr>
</tbody>
</table>

VNF Rack Management Parameters:

VNF rack is a logical grouping of compute hosts and used to map the resources for specific VNFs. This is equivalent to Availability Zones / Host aggregate in case of OpenStack. For the deployment where UAS is used to on board multiple VNFs such as Ultra-M, it becomes handy to manage placements. VNF racks are created at NSD level and it can be used in VDUs as one of the placement filters.
### VIM Artifacts Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-rack-id</td>
<td>M</td>
<td>string</td>
<td>VNF rack descriptor ID.</td>
</tr>
<tr>
<td>description</td>
<td>O</td>
<td>string</td>
<td>User-defined metadata for description/notes.</td>
</tr>
<tr>
<td>host-aggregate</td>
<td>M</td>
<td>array of host-aggregation</td>
<td>List of host aggregate descriptors.</td>
</tr>
</tbody>
</table>

#### Tenant Provisioning and Management Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tenant-id</td>
<td>M</td>
<td>string</td>
<td>Tenant name.</td>
</tr>
<tr>
<td>login-credential</td>
<td>M</td>
<td>secure-token</td>
<td>Login credentials to be configured for this tenant authorization.</td>
</tr>
<tr>
<td>tenant-role</td>
<td>M</td>
<td>choice string</td>
<td>Tenant role that is mapped to rights and privileges on the VIM. Default value is “admin”. The role can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• resellerAdmin: Cloud provider administrator</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• admin: Administrative role that includes permission to administrate all projects and tenants hosted on the VIM.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• swiftoperator: Provides privilege to modify SWIFT containers.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• <em>member</em>: Role assigned to a new user with administrative privilege only for that specific tenant.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• heat_stack_user: Assigned to user that is used to create stack deployment.</td>
</tr>
<tr>
<td>no-of-instances-allowed</td>
<td>O</td>
<td>uint32</td>
<td>Number of instances allowed per tenant (default: 1000).</td>
</tr>
<tr>
<td>no-of-cores-allowed</td>
<td>O</td>
<td>uint32</td>
<td>Number of instance cores (VCPUs) allowed per tenant (default: 1000).</td>
</tr>
</tbody>
</table>
### VIM Artifacts Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>no-of-injected-files</td>
<td>0</td>
<td>uint32</td>
<td>Number of injected files allowed per tenant (default: 50).</td>
</tr>
<tr>
<td>no-of-injected-files-content-in-bytes</td>
<td>0</td>
<td>uint64</td>
<td>Number of content bytes allowed per injected file (default: 1024000).</td>
</tr>
<tr>
<td>megabytes-of-ram-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Megabytes of instance RAM allowed per tenant.</td>
</tr>
<tr>
<td>no-of-floating-ips</td>
<td>0</td>
<td>uint32</td>
<td>Number of floating IPs allowed per tenant (default: 50).</td>
</tr>
<tr>
<td>max-no-of-subnets-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Number of subnets allowed per tenant (default: 1000).</td>
</tr>
<tr>
<td>max-no-of-ports-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Number of ports allowed per tenant (default: 1000).</td>
</tr>
<tr>
<td>no-of-volumes-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Number of volumes allowed per tenant (default: 500).</td>
</tr>
<tr>
<td>volumes-allowed-gb</td>
<td>0</td>
<td>uint32</td>
<td>Gigabytes of volume allowed per tenant (default: 4000).</td>
</tr>
<tr>
<td>volumes-backup-size-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Gigabytes of volume backup allowed per tenant (default: 4000).</td>
</tr>
</tbody>
</table>

### Example RPC

```xml
<nc:rpc message-id="urn:uuid:361fb8c6-000d-4e22-a2d6-4ee17412fb2e"
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target/>
    <config>
      <nsd xmlns="http://www.cisco.com/usp/nfv/usp-nads">
        <vim-identity>vim_one</vim-identity>
        <vim-artifact>vim_one_artifact</vim-artifact>
      </nsd>
      <secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
        <secure-id>vim-admin-creds</secure-id>
        <user>admin</user>
        <password>Cisco@123</password>
      </secure-token>
      <secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
        <secure-id>sjc-core</secure-id>
        <user>sjccore</user>
        <password>Cisco@123</password>
      </secure-token>
      <vim xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
        <vim-id>vim_one</vim-id>
      </vim>
    </config>
  </nc:edit-config>
</nc:rpc>
```
Example Output

```xml
  <ok/>
</rpc-reply>
```

Configuring VNF Rack

**Operation:** edit-config

**Namespace:** vnf-rackd xmlns="http://www.cisco.com/usp/nfv/usp-uas-common"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-rack-id</td>
<td>M</td>
<td>string</td>
<td>VNF rack descriptor ID.</td>
</tr>
<tr>
<td>description</td>
<td>O</td>
<td>string</td>
<td>User-defined metadata for description/notes.</td>
</tr>
<tr>
<td>host-aggregate</td>
<td>M</td>
<td>array of host-aggregation</td>
<td>List of host aggregate descriptors.</td>
</tr>
</tbody>
</table>

The host aggregate is a grouping of named hosts within a group or zone.
VIM Artifacts Configuration

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host-aggregate-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID for the Host Aggregate Descriptor.</td>
</tr>
<tr>
<td>host</td>
<td>M</td>
<td>array of strings</td>
<td>List of host names as exists on VIM.</td>
</tr>
</tbody>
</table>

Example RPC

```xml
<nc:rpc message-id="urn:uuid:5cdea00b-c4df-40cd-9de5-2acfc530a89f"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
      <vnf-rackd xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
        <vnf-rack-id>vnf-rack1</vnf-rack-id>
        <host-aggregate>
          <host-aggregate-id>mgmt</host-aggregate-id>
          <host>
            <host-id>tb1ano-compute-1.localdomain</host-id>
          </host>
          <host>
            <host-id>tb1ano-compute-10.localdomain</host-id>
          </host>
          <host>
            <host-id>tb1ano-compute-11.localdomain</host-id>
          </host>
          -------
        </host-aggregate>
      </vnf-rackd>
    </config>
  </nc:edit-config>
</nc:rpc>
```

Example Output

```xml
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:5cdea00b-c4df-40cd-9de5-2acfc530a89f"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

Defining a Tenant

A new tenant can be provisioned on the VIM using the VIM-ARTIFACT RPC. This section describes the tenant-specific attributes.

**Operation:** edit-config

**Namespace:** tenantd xmlns="http://www.cisco.com/usp/nfv/usp-uas-common"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tenant-id</td>
<td>M</td>
<td>string</td>
<td>Tenant name.</td>
</tr>
</tbody>
</table>
### Parameter Name | Required | Type | Description
--- | --- | --- | ---
login-credential | M | secure-token | Login credentials to be configured for this tenant authorization.
tenant-role | M | choice string | Tenant role that is mapped to rights and privileges on the VIM. Default value is “admin”. The role can be one of the following:
  - resellerAdmin: Cloud provider administrator
  - admin: Administrative role that includes permission to administrate all projects and tenants hosted on the VIM.
  - swiftoperator: Provides privilege to modify SWIFT containers.
  - _member_: Role assigned to a new user with administrative privilege only for that specific tenant.
  - heat_stack_user: Assigned to user that is used to create stack deployment.

### Parameter Name | Required | Type | Description
--- | --- | --- | ---
no-of-instances-allowed | 0 | uint32 | Number of instances allowed per tenant (default: 1000).
no-of-cores-allowed | 0 | uint32 | Number of instance cores (VCPUs) allowed per tenant (default: 1000).
no-of-injected-files | 0 | uint32 | Number of injected files allowed per tenant (default: 50).
no-of-injected-files-content-in-bytes | 0 | uint64 | Number of content bytes allowed per injected file (default: 1024000).
megabytes-of-ram-allowed | 0 | uint32 | Megabytes of instance RAM allowed per tenant.
no-of-floating-ips | 0 | uint32 | Number of floating IPs allowed per tenant (default: 50).
<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>max-no-of-subnets-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Number of subnets allowed per tenant (default: 1000).</td>
</tr>
<tr>
<td>max-no-of-ports-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Number of ports allowed per tenant (default: 1000).</td>
</tr>
<tr>
<td>no-of-volumes-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Number of volumes allowed per tenant (default: 500).</td>
</tr>
<tr>
<td>volumes-allowed-gb</td>
<td>0</td>
<td>uint32</td>
<td>Gigabytes of volume allowed per tenant (default: 4000).</td>
</tr>
<tr>
<td>volumes-backup-size-allowed</td>
<td>0</td>
<td>uint32</td>
<td>Gigabytes of volume backup allowed per tenant (default: 4000).</td>
</tr>
</tbody>
</table>

Example RPC

```xml
<nc:rpc message-id="urn:uuid:361fb8c6-000d-4e22-a2d6-4ee17412fb2e"
  xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
      <nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds">
        <nsd id="sjc"/>
        <vim-identity vim_one identity=vim_one/>
      </nsd>
      <secure-token xmlns="http://www.cisco.com/usp/nfv/usp-secure-token">
        <secure-id id="vim-admin-creds"/>
        <user>admin</user>
        <password>Cisco@123</password>
      </secure-token>
      <vim xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
        <vim-id vim_one identity=vim_one/>
        <auth-url>http://172.21.203.31:5000/v2.0</auth-url>
        <user>admin</user>
      </vim>
      <tenant xmlns="http://www.cisco.com/usp/nfv/usp-uas-common">
        <tenant-id sjccore identity=sjccore/>
        <login-credential ljccore identity=sjccore-creds/>
        <tenant-role>admin</tenant-role>
        <tenant-quota>
          <no-of-instances-allowed>10000</no-of-instances-allowed>
          <no-of-cores-allowed>2000000</no-of-cores-allowed>
          <no-of-injected-files>0</no-of-injected-files>
          <no-of-injected-files-content-in-bytes>0</no-of-injected-files-content-in-bytes>
          <megabytes-of-ram-allowed>500000000</megabytes-of-ram-allowed>
          <no-of-floating-ips>600</no-of-floating-ips>
          <max-no-of-subnets-allowed>700</max-no-of-subnets-allowed>
          <max-no-of-ports-allowed>8000</max-no-of-ports-allowed>
          <no-of-volumes-allowed>900</no-of-volumes-allowed>
          <volumes-allowed-gb>10000</volumes-allowed-gb>
        </tenant-quota>
      </tenant>
    </config>
  </nc:edit-config>
</nc:rpc>
```
VIM Artifacts Configuration

Example Output

  <ok/>
</rpc-reply>
VIM Artifacts Deployment

This section describes the parameters that can be used to manage the deployment of VIM artifacts – VNF rack, routers, and tenants.

Operations

Activating VNF Rack

Operation: nsd:activate


Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-rack</td>
<td>M</td>
<td>string</td>
<td>Reference to one or more vnf-rackr on boarded as part of this NSD.</td>
</tr>
</tbody>
</table>

Example RPC

```
<nc:rpc message-id="urn:uuid:eec12ec0-d879-4737-807d-3cf00bdf5558" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
    xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds">
        <nsd:nsd>sjc-autoit</nsd:nsd>
        <nsd:vnf-rack>vnf-rack1</nsd:vnf-rack>
    </nsd:activate>
</nc:rpc>
```

Example Output

```
    <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1509826883-420684</transaction-id>
</rpc-reply>
```

Obtaining Notifications During VNF Rack Activation

Operation: get


Parameters:
<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice</td>
<td>Type of operation (same as the type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

Example Notification

```xml
<nvim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"
<instance-id>sjc-autoit-instance</instance-id>
<descriptor-id>sjc-autoit</descriptor-id>
<transaction-id>1509826883-420684</transaction-id>
<operation-type>activate-vnf-rack</operation-type>
<status>in-progress</status>
</vim-deployment-event>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-04T20:21:23.523572+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"
<instance-id>vnf-rack1-instance</instance-id>
<descriptor-id>vnf-rack1</descriptor-id>
<transaction-id>1509826883-420684</transaction-id>
<operation-type>activate-vnf-rack</operation-type>
<status>in-progress</status>
</vim-deployment-event>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-04T20:21:23.507322+00:00</eventTime>
<ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"
<instance-id>sjc-autoit-instance</instance-id>
<descriptor-id>sjc-autoit</descriptor-id>
<transaction-id>1509826883-420684</transaction-id>
<operation-type>activate-vnf-rack</operation-type>
<status>in-progress</status>
</ns-deployment-event>
```

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-11-04T20:21:23.439984+00:00</eventTime>
<ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"
<instance-id>sjc-autoit-instance</instance-id>
<descriptor-id>sjc-autoit</descriptor-id>
<transaction-id>1509826883-420684</transaction-id>
<operation-type>activate-vnf-rack</operation-type>
<status>requested</status>
</ns-deployment-event>
```
Obtaining VNF Rack Operational Instance Information

The UAS maintains the operational data as manifest and uses this to automatically place the VDUs on these VNF racks. In VDU, the VNF rack descriptor is inherited as part of the placement policy.

**Operation:** get

**Namespace:** vim xmlns="http://www.cisco.com/usf/vnf/usf-uas-common"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-rack-id</td>
<td>M</td>
<td>string</td>
<td>VNF rack descriptor ID.</td>
</tr>
<tr>
<td>description</td>
<td>O</td>
<td>string</td>
<td>User-defined metadata for description/notes.</td>
</tr>
<tr>
<td>host-aggregate</td>
<td>M</td>
<td>array of host-aggregation</td>
<td>List of host aggregate descriptors.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:8300f38c-dd03-448b-97f5-c8ca7f3d038" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:get>
    <nc:filter type="subtree">
      <vnf-rack xmlns="http://www.cisco.com/usf/vnf/usf-uas-common"/>
    </nc:filter>
  </nc:get>
</nc:rpc>
```

**Example Output**

```xml
  <data>
    <vnf-rack xmlns="http://www.cisco.com/usf/vnf/usf-uas-common">
      <vnf-rack-id>vnf-rack1</vnf-rack-id>
    </vnf-rack>
  </data>
</rpc-reply>
```
<vnf-rack>vnf-rack1</vnf-rack>
</host-aggregate>
</host-aggregate>
</vnf-rack>
</data>
</rpc-reply>

Deactivating VNF Rack

**Operation:** nsd:deactivate

**Namespace:** nsd:deactivate xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-rack</td>
<td>M</td>
<td>string</td>
<td>Reference to one or more vnf-rack on boarded as part of this NSD.</td>
</tr>
</tbody>
</table>

Example RPC

```xml
    <nsd:vnf-rack>vnf-rack1</nsd:vnf-rack>
  </nsd:deactivate>
</nc:rpc>
```

Example Output

```xml
  <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1509828954-260046</transaction-id>
</rpc-reply>
```

Obtaining Notifications During VNF Rack Deactivation

**Operation:** get

**Namespace:** vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"

**Parameters:**
### Parameter Name | Required | Type | Description
--- | --- | --- | ---
instance-id | M | string | A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.
descriptor-id | M | string | Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.
transaction-id | M | string | Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.
operation-type | M | choice string | Type of operation (same as the type of transaction)
status | M | choice string | Current deployment status (start, in-progress, error, or success).
steps-total | O | uint16 | Total number of steps required to be performed.
steps-completed | O | uint16 | Total number of steps completed.
version | O | string | Deployment version.

**Example Notification**

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-04T20:55:54.285399+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-autoit-instance</instance-id>
    <descriptor-id>sjc-autoit</descriptor-id>
    <transaction-id>1509828954-260046</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>requested</status>
  </ns-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-04T20:55:54.351214+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-autoit-instance</instance-id>
    <descriptor-id>sjc-autoit</descriptor-id>
    <transaction-id>1509828954-260046</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>in-progress</status>
  </ns-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-04T20:55:54.360301+00:00</eventTime>
  <vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>vnf-rack1-instance</instance-id>
    <descriptor-id>vnf-rack1</descriptor-id>
    <transaction-id>1509828954-260046</transaction-id>
    <operation-type>deactivate-vnf-rack</operation-type>
    <status>in-progress</status>
  </vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-04T20:55:56.354056+00:00</eventTime>
</notification>
```
Activating a Tenant

**Operation:** nsd:activate

**Namespace:** nsd:activate xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vim-artifact</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM artifact descriptor used to provision the set of parameters on VIM to onboard VNFs.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:0d365b37-6744-4d1a-807c-fbaa1690d53e" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <nsd:nsd>sjc</nsd:nsd>
    <nsd:vim-artifact>vim_one_artifact</nsd:vim-artifact>
  </nsd:activate>
</nc:rpc>
```

**Example Output**

```xml
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:0d365b37-6744-4d1a-807c-fbaa1690d53e">
  <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1512588866-645290</transaction-id>
</rpc-reply>
```

**Obtaining Notifications During Tenant Activation**

**Operation:** get

**Namespace:** vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"

**Parameters:**
## Parameter Name | Required | Type | Description
--- | --- | --- | ---
instance-id | M | string | A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.
descriptor-id | M | string | Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.
transaction-id | M | string | Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.
operation-type | M | choice string | Type of operation (same as the type of transaction)
status | M | choice string | Current deployment status (start, in-progress, error, or success).
steps-total | O | uint16 | Total number of steps required to be performed.
steps-completed | O | uint16 | Total number of steps completed.
version | O | string | Deployment version.

### Example Notification

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T19:34:26.668906+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <descriptor-id>sjc</descriptor-id>
    <transaction-id>1512588866-645290</transaction-id>
    <operation-type>activate-ns-deployment</operation-type>
    <status>requested</status>
  </ns-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T19:34:26.730665+00:00</eventTime>
  <vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc:sjccore</instance-id>
    <descriptor-id>sjccore</descriptor-id>
    <transaction-id>1512588866-645290/1512588866-699182</transaction-id>
    <operation-type>activate-tenant-deployment</operation-type>
    <status>requested</status>
  </vim-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T19:34:26.857343+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <descriptor-id>sjc</descriptor-id>
    <transaction-id>1512588866-645290</transaction-id>
    <operation-type>activate-ns-deployment</operation-type>
    <status>in-progress</status>
  </ns-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T19:34:26.989887+00:00</eventTime>
  <vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc:sjccore</instance-id>
    <descriptor-id>sjccore</descriptor-id>
    <transaction-id>1512588866-645290/1512588866-699182</transaction-id>
    <operation-type>activate-tenant-deployment</operation-type>
    <status>requested</status>
  </vim-deployment-event>
</notification>
```
Deactivating a Tenant

**Operation:** nsd:deactivate

**Namespace:** nsd:deactivate xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vim-artifact</td>
<td>O</td>
<td>reference</td>
<td>Reference to VIM artifact descriptor used to provision the set of parameters on VIM to on board VNFs.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:29317423-3ba4-4356-972c-e6a154a5731e" xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
        <nsd:nsd>sjc</nsd:nsd>
        <nsd:vim-artifact>vim_one_artifact</nsd:vim-artifact>
    </nsd:deactivate>
</nc:rpc>
```

**Example Output**

```xml
    <transaction-id>1512588970-645290</transaction-id>
</rpc-reply>
```

**Obtaining Notifications During Tenant Deactivation**

**Operation:** get

**Namespace:** vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"
### Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to create operational record for this instance. Hence, you can pull operational record for this instance at a given time.</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice string</td>
<td>Type of operation (same as the type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

### Example Notification

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T19:29:21.05899+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <descriptor-id>sjc</descriptor-id>
    <transaction-id>1512588560-927573</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>requested</status>
  </ns-deployment-event>
</notification>
```

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T19:29:21.107291+00:00</eventTime>
  <vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc:sjccore</instance-id>
    <descriptor-id>sjccore</descriptor-id>
    <transaction-id>1512588560-927573/1512588561-086910</transaction-id>
    <operation-type>deactivate-tenant-deployment</operation-type>
    <status>requested</status>
  </vim-deployment-event>
</notification>
```

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T19:29:21.171115+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc-instance</instance-id>
    <descriptor-id>sjc</descriptor-id>
    <transaction-id>1512588560-927573</transaction-id>
    <operation-type>deactivate-ns-deployment</operation-type>
    <status>in-progress</status>
  </ns-deployment-event>
</notification>
```

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-12-06T19:29:21.21764+00:00</eventTime>
  <vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>sjc:sjccore</instance-id>
    <descriptor-id>sjccore</descriptor-id>
    <transaction-id>1512588560-927573/1512588561-086910</transaction-id>
    <operation-type>deactivate-tenant-deployment</operation-type>
    <status>requested</status>
  </vim-deployment-event>
</notification>
```
<instance-id>sjc:sjccore</instance-id>
<descriptor-id>sjccore</descriptor-id>
<transaction-id>1512588560-927573/1512588561-086910</transaction-id>
<operation-type>deactivate-tenant-deployment</operation-type>
<status>in-progress</status>
</vim-deployment-event>
</notification>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T19:29:22.521953+00:00</eventTime>
<vim-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:sjccore</instance-id>
<descriptor-id>sjccore</descriptor-id>
<transaction-id>1512588560-927573/1512588561-086910</transaction-id>
<operation-type>deactivate-tenant-deployment</operation-type>
<status>success</status>
</vim-deployment-event>
</notification>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
<eventTime>2017-12-06T19:29:22.590785+00:00</eventTime>
<ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
<instance-id>sjc:instance</instance-id>
<descriptor-id>sjc</descriptor-id>
<transaction-id>1512588560-927573</transaction-id>
<operation-type>deactivate-ns-deployment</operation-type>
<status>success</status>
</ns-deployment-event>
</notification>
VNF Descriptor Configuration

NSD can have one or more VNFDs where each VNFD defines the deployment flavor for a specific VNF that includes all the aspects of VNF resources and associated networking.

VNF Descriptor Configuration Parameters

**Operation**: edit-config

**Namespace**: nsd xmlns="http://www.cisco.com/usp/nfv/usps-nfds"

**Parameters**: 
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnfd-id</td>
<td>M</td>
<td>string</td>
<td>VNFD descriptor ID to uniquely identify the VNF.</td>
</tr>
<tr>
<td>vnf-type</td>
<td>M</td>
<td>choice string</td>
<td>VNF type. This can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• usp-uas: Cisco Ultra Automation Services</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• uem: VNF-EM Deployment as separate VNF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• esc: Cisco ESC VNFM Deployment</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• ugp: Cisco Ultra Gateway Platform</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• ucps: Cisco Ultra Policy Platform</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• ugp-standalone: Cisco Ultra Gateway Platform with external UEM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• ucps-standalone: Cisco Ultra Policy Platform with external UEM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• usp-gilan: Ultra Gi-LAN Deployment</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>VNF descriptor version.</td>
</tr>
<tr>
<td>configuration</td>
<td>O</td>
<td>array of names</td>
<td>For details, see the VNF Configuration Parameters section.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>configuration</td>
<td></td>
</tr>
<tr>
<td>vld</td>
<td>O</td>
<td>array of vlds</td>
<td>List of virtual links used internally within VNF.</td>
</tr>
<tr>
<td>external-</td>
<td>O</td>
<td>array of</td>
<td>For details, see the VNF External Connection Points Parameters section.</td>
</tr>
<tr>
<td>connection-</td>
<td></td>
<td>external-</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>connection-point</td>
<td></td>
</tr>
<tr>
<td>high-availability</td>
<td>O</td>
<td>bool</td>
<td>Enable/disable high availability at VNF level. High Availability controls anti-affinity and cluster size. In case of UAS and ESC, if High</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Availability feature is turned off, only single instance of VNFC is deployed and for VNF-EM, UGP, UCPS, this flag controls the placement. By</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>default anti-affinity is turned on, two VNFC instances of the same VDU are placed on different hosts. If High-Availability is turned off, the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>UAS turns off this placement rule and multiple VNFCs of the same VDU can be placed on the same host, given a host has enough resources to host</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>VNFC instance. Default is “false”.</td>
</tr>
<tr>
<td>vnfc</td>
<td>M</td>
<td>array of vnfc</td>
<td>List of VNFC descriptor used to deploy this VNF.</td>
</tr>
<tr>
<td>vnfm</td>
<td>O*</td>
<td>reference</td>
<td>Reference to VNFD providing VNFM. This is required only for UGP-standalone and UCPS-standalone VNF where VNFM is deployed as separate VNF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and not as VNFC.</td>
</tr>
<tr>
<td>vim-identity</td>
<td>M</td>
<td>reference</td>
<td>Reference to VIM identity used by UAS to access VIM.</td>
</tr>
<tr>
<td>usp-usf</td>
<td>O</td>
<td>array of service</td>
<td>List of service element groups to enable Ultra Service Framework services. This is applicable only for Gi-LAN based deployment. See the USF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USF entries</td>
<td>Descriptor Parameters section for more details.</td>
</tr>
<tr>
<td>deploy-nsd</td>
<td>O</td>
<td>reference</td>
<td>Reference to NSD that should be deployed.</td>
</tr>
<tr>
<td>cloud-info</td>
<td>O</td>
<td>descriptor</td>
<td>See the Cloud-Info Descriptor Parameters section for more details.</td>
</tr>
</tbody>
</table>

**Cloud-Info Descriptor Parameters**

This allows you to define VNFM/VIM related parameters.

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.
**Operation:** edit-config

**Namespace:** nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnfm</td>
<td>O</td>
<td>reference</td>
<td>Unique ID to VNF instance. See the VNFM Descriptor Parameters for more details.</td>
</tr>
<tr>
<td>vim</td>
<td>O</td>
<td>reference</td>
<td>VIM identity to access VIM from VNFM. By default, the UAS uses default tenant, often the tenant VNFM has been bootstrapped with. Previously this was defined under the VNFM descriptor.</td>
</tr>
<tr>
<td>vnf-em</td>
<td>O</td>
<td>reference</td>
<td>Reference to VNF descriptor</td>
</tr>
</tbody>
</table>

**VNFM Descriptor Parameters**

VNFM-EM does not communicate directly with VIM hence all the resources should be onboarded either via VNFM or UAS. The UAS also uses VNFM to onboard VNF artifacts. VNFM is required configuration for all the VNFs except UAS and ESC VNF.

There are two ways VNFM can be configured either by defining VNF descriptor for VNFM and the UAS will orchestrate that VNF as dependent VNF or by referring a VNF Instance. You can create VNF instance for VNFM that is not brought up as part of this NSD. For example, you can share one ESC across several NSDs so one NSD can bring up the VNFM in other NSD, you refer it as reference. The connectivity networks are allowed to be as well pre-created so you can always share the same VNFM across several NSDs.

**Operation:** edit-config

**Namespace:** nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnfm-type</td>
<td>M</td>
<td>choice string</td>
<td>VNFM type. It should be either “descriptor” or “instance”.</td>
</tr>
<tr>
<td>vnfd</td>
<td>M*</td>
<td>reference</td>
<td>Reference to VNF descriptor. Required when vnfm-type is set to descriptor.</td>
</tr>
<tr>
<td>vnf-instance</td>
<td>M*</td>
<td>reference</td>
<td>Reference to VNF instance. Required when vnfm-type is set to instance.</td>
</tr>
</tbody>
</table>

* Denotes the default requirement of the attribute. However, it can be modified from (O)ntional to (M)andatory or (M)andatory to (O)ntional depending on use case scenario.

**VNF Instance Descriptor Parameters**

This allows you to onboard pre-created VNF onto UAS. For example, pre-created VNFM.

**Operation:** edit-config

**Namespace:** nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**
VNF Descriptor Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-instance-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID to VNF instance.</td>
</tr>
<tr>
<td>vnf-type</td>
<td>M</td>
<td>choice string</td>
<td>One of the defined VNF types.</td>
</tr>
<tr>
<td>external-connection-point</td>
<td>M</td>
<td>inet:ip-address</td>
<td>External IP address on this VNF to access APIs.</td>
</tr>
<tr>
<td>netconf-credential</td>
<td>M</td>
<td>secure-token</td>
<td>Reference to secure token to authenticate API endpoint over NETCONF.</td>
</tr>
</tbody>
</table>

VNF Configuration Parameters

Following are the set of configuration parameters that are defined to personalize a VNF deployment. These configurations are applied at VNF level. There are separate configurations available at VNFC level.

**Operation:** edit-config

**Namespace:** nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>internal-network-mtu</td>
<td>uint16</td>
<td>MTU for internal networks. This is passed as one of the bootstrapping parameters to VPC DI VDU to set the interface MTU to value specified here.</td>
</tr>
<tr>
<td>openstack.endpoint</td>
<td>string</td>
<td>Endpoint value in case of OpenStack. By default, its value is &quot;publicURL&quot;.</td>
</tr>
<tr>
<td>secure-login</td>
<td>bool</td>
<td>Enable secure-login. This is used to disable password based authentication and allow only SSH key based authentication.</td>
</tr>
<tr>
<td>boot-time</td>
<td>uint16</td>
<td>Maximum time to allow VNF to be alive in seconds. Default value is 1800 seconds.</td>
</tr>
<tr>
<td>domain-name</td>
<td>string</td>
<td>Domain name. You can define ${CF_DOMAIN_NAME}, UAS will replace that variable with actual configured domain name.</td>
</tr>
<tr>
<td>set-vim-instance-name</td>
<td>bool</td>
<td>Set VM Instance names based on NSD/VNFD. By default it is set to True.</td>
</tr>
<tr>
<td>dns-server</td>
<td>array of inet:ip-address</td>
<td>List of DNS servers.</td>
</tr>
<tr>
<td>syslog-server</td>
<td>array of syslog descriptors</td>
<td>For details, see the following section.</td>
</tr>
</tbody>
</table>

Syslog Descriptor Parameters

Based on this configuration, the UAS automatically configures syslog within VNFC for UAS, VNF-EM, and UGP.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>inet:ip-address</td>
<td>Remote server IP address.</td>
</tr>
<tr>
<td>port</td>
<td>O</td>
<td>uint16</td>
<td>Remote server port number. Default value is 514.</td>
</tr>
<tr>
<td>severity</td>
<td>O</td>
<td>uint8 (0..6)</td>
<td>Limit syslogs specifying the severity level. Default value is set to 6.</td>
</tr>
</tbody>
</table>
VNF External Connection Points Parameters

External connection points are a set of virtual ports the UAS creates to provide the connectivity to VNF cluster. This is required when high-availability is turned on. Sometimes a VNF can have multiple external connection points if there are multiple OAM VDUs. For example, in case of UGP and UCPS type of VNF, both VNF-EM and CF are connected via a HA VIP. Other VNFs like UAS, Cisco ESC will have one connection point.

**Operation:** edit-config

**Namespace:** nsd xmlns="http://www.cisco.com/usp/nfv/usp-nfds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vnfc</td>
<td>M</td>
<td>reference</td>
<td>Reference to VNFC within VNF.</td>
</tr>
<tr>
<td>aggregate-connection-point</td>
<td>M</td>
<td>reference</td>
<td>Reference to connection point within VNFC. This is to ensure a virtual port is created on the same network and port security rule is configured to allow the traffic.</td>
</tr>
<tr>
<td>ip-address</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Virtual IP address, by default a free IP is allocated. You can provide the input for VIP if you need a fixed IP address to given VNFC.</td>
</tr>
<tr>
<td>floating-ip</td>
<td>O</td>
<td>floating-ip</td>
<td>Floating IP descriptor. For more details, see the following section.</td>
</tr>
</tbody>
</table>

Floating-IP Descriptor Parameters

Use this descriptor to associate floating IP to virtual port.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>O</td>
<td>bool</td>
<td>Enable/disable floating IP assignment. Default is “false”.</td>
</tr>
<tr>
<td>ip-address</td>
<td>O</td>
<td>inet:ip-address</td>
<td>Floating IP address, by default a free IP will be allocated if allowed.</td>
</tr>
<tr>
<td>external-network</td>
<td>O*</td>
<td>string</td>
<td>External network name. It is mandatory if enabled is set to true.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

Health Check Descriptor Parameters

Health check descriptor contains a set of parameters used to control the health check at VNFC level.

**Operation:** edit-config

**Namespace:** nsd xmlns="http://www.cisco.com/usp/nfv/usp-nfds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enabled</td>
<td>O</td>
<td>bool</td>
<td>Enable/Disable health monitoring.</td>
</tr>
<tr>
<td>probe-frequency</td>
<td>O</td>
<td>uint16</td>
<td>Health Check Frequency in seconds. UAS uses this as health probe time, meaning every polling interval UAS will invoke health check. Default value is 10 seconds.</td>
</tr>
</tbody>
</table>
## VNFC Descriptor Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>probe-max-miss</td>
<td>O</td>
<td>uint16</td>
<td>Maximum number of health probe misses before VNFC instance is declared dead. Default value is 6.</td>
</tr>
</tbody>
</table>
| recovery-type   | O        | choice string | Recovery type. It can be one of the following:  
• restart: Recovery only by restarting, move the VNFC instance to error after max retries  
• external: Recovery performed by external entity. No auto-recovery  
• restart-then-redeploy: Restart the VM on failure. After maximum retries, redeploy the failed VNFC instances. Default value is restart-then-redeploy. |
| retry-count     | O        | uint16   | Number of retries to recover the VNFC Instance. Default value is set to restart-then-redeploy. |
| boot-time       | O        | uint16   | Initial Bootup time for the VNFC. Default value is 300 seconds. |
| script          | O        | string   | Script to check VNFC health, by default UAS ICMP script will be used. |

### VNFC Descriptor Parameters

**Operation:** edit-config

**Namespace:** nsd xmlns="http://www.cisco.com/usp/nfv/usp-nsds"

### Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnfc-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier for VNFC.</td>
</tr>
<tr>
<td>number-of-instances</td>
<td>O</td>
<td>int32</td>
<td>Number of VNFC instances.</td>
</tr>
<tr>
<td>vdu</td>
<td>M</td>
<td>vdu</td>
<td>VDU information. See the VNFC-VDU Descriptor Parameters section for more details.</td>
</tr>
<tr>
<td>volume</td>
<td>O</td>
<td>array of volume-instance-group reference</td>
<td>These are a set of pre-created volumes to be used. Once this option is used for boot or storage volumes, volume descriptors defined at VDU level are ignored. Instead, UAS uses pre-created set of volumes to associate with VNFC instance. See the VNFC Volume Instance Group Parameters for more details.</td>
</tr>
<tr>
<td>aggregate-connection-point</td>
<td>M</td>
<td>array of connection-point descriptor</td>
<td>List of connection points for the VNFC Instance. There should be at least one connection point.</td>
</tr>
<tr>
<td>aggregate-connection-points</td>
<td>O</td>
<td>array of connection-point-aggregation-descriptor</td>
<td>List of connection point aggregator descriptor. See the VNFC Connection Point Aggregation Descriptor Parameters section for more details.</td>
</tr>
<tr>
<td>initialization-variable</td>
<td>O</td>
<td>array of initialization-variable descriptor</td>
<td>List of initialization variables to personalize per-instance configuration. See the VNFC Initialization Variable Descriptor Parameters section for more details.</td>
</tr>
<tr>
<td>syslog</td>
<td>O</td>
<td>syslog descriptor</td>
<td>Remote syslog parameters. Applicable to VNFC type - UAS, EM, ESC and CF. See the VNFC Syslog Descriptor Parameters for more details.</td>
</tr>
</tbody>
</table>
VNFC Initialization Variable Descriptor Parameters

VDU defines a set of metadata files which UAS will use for VNFC instances at the time of instantiation either via configuration drive (day-zero) or via NED (day-one). These configuration files can be templates (for example, service template) with variables defined in these files. These variables can be populated at two places:

- **VDU level**: You can define variable name and value at VDU level and those will be applicable for all the instances created from this VDU.
- **VNFC Instance level**: This is to personalize the data at per instance level. The variables should be listed to satisfy all the instances of VNFC created.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>configuration</td>
<td>M</td>
<td>reference</td>
<td>Reference to configuration file defined at VDU level.</td>
</tr>
<tr>
<td>variable</td>
<td>M</td>
<td>array of variable Descriptor</td>
<td>List of configured variables. See the Variable descriptor parameter section for more details</td>
</tr>
</tbody>
</table>

Variable Descriptor Parameters

This parameter is used to configure the value for a particular variable for one or more instance(s).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>variable</td>
<td>M</td>
<td>string</td>
<td>Name of the variable.</td>
</tr>
<tr>
<td>instance</td>
<td>M*</td>
<td>array of instances</td>
<td>List of instances in the form &quot;instance-number&quot; and &quot;value&quot;. List is defined to personalize the value of the variable for each instance.</td>
</tr>
</tbody>
</table>
| type        | M*       | choice string | Type for group of instances:  
  • prefix: Prefix name to be used for the variable in case the variable name corresponds to any name.  
  • ip-pool: Ip pool to be used to configure IP address in case the variable name corresponds to IPV4 |
| value       | M*       | string | Prefix value in case the type is prefix and reference to the network instance in case the type is ip-pool |

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.
For this descriptor, either instance or a combination of type and value should be configured.

VNFC Connection Point Aggregation Descriptor Parameters

This connection point aggregation is used to provide interface bonding for link aggregation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bond</td>
<td>M</td>
<td>string</td>
<td>Unique identifier for this Link Aggregation or Interface bonding.</td>
</tr>
<tr>
<td>aggregate-connection-point</td>
<td>M</td>
<td>array of references</td>
<td>List of connection points used to aggregate for this specific bonding.</td>
</tr>
</tbody>
</table>
VNFC Connection Point Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connection-point-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier for connection point such as eth0, eth1 etc.</td>
</tr>
<tr>
<td>virtual-link descriptor</td>
<td>M</td>
<td>virtual-link descriptor</td>
<td>Virtual link descriptor to associate this connection point within VNFC.</td>
</tr>
<tr>
<td>fixed-ip</td>
<td>O</td>
<td>array of inet:ip-address</td>
<td>If you want to use fixed IP address to this connection point, you can supply a set of fixed IPs to be used for the instances of this VNFC. You should supply number of fixed IPs equal to the number of instances defined for this VNFC.</td>
</tr>
<tr>
<td>allowed-address-pair</td>
<td>O</td>
<td>array of allowed address pair</td>
<td>List of allowed address pair. See the VNFC Connection Point Allowed Address Pair Descriptor Parameters section for details.</td>
</tr>
<tr>
<td>vlan</td>
<td>O</td>
<td>array of vlan descriptor</td>
<td>List of VLAN interfaces created on this interface. See the VNFC Connection Point VLAN Descriptor Parameters section for more details.</td>
</tr>
</tbody>
</table>

VNFC Connection Point VLAN Descriptor Parameters

VNF creates logical interfaces on a given connection point. For example, if you have SRIOV-FLAT port, VPC gateway will use the same port but different kind of traffic using a logical interface aka VLAN interface. The same is true for Gi-LAN usage. This allows UAS to pass IP address along with VRF to VNF for auto-creation of internal VLAN interfaces. This avoids a need for user to log in to VNF and create these interfaces manually.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vlan-id</td>
<td>M</td>
<td>uint16 (1..4096)</td>
<td>Unique VLAN identifier for this interface.</td>
</tr>
<tr>
<td>ipv4</td>
<td>M*</td>
<td>inet:ip-prefix</td>
<td>IPv4 subnet used for this VLAN Interface. One of IPv4 or IPv6 is mandatory.</td>
</tr>
<tr>
<td>ipv6</td>
<td>M*</td>
<td>inet:ip-prefix</td>
<td>IPv6 subnet used for this VLAN Interface. One of IPv4 or IPv6 is mandatory.</td>
</tr>
<tr>
<td>vrf</td>
<td>O</td>
<td>string</td>
<td>VRF name if applicable.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

VNFC Connection Point Allowed Address Pair Descriptor Parameters

The allowed address pair extension extends the port attribute to enable you to specify arbitrary mac_address/ip_address(cidr) pairs that are allowed to pass through a port regardless of the subnet associated with the network.

Please note UAS automatically adds allowed address pair to take care for virtual IP.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>inet:ip-prefix</td>
<td>IP prefix (cidr)</td>
</tr>
<tr>
<td>mac-address</td>
<td>O</td>
<td>inetyang:mac-address</td>
<td>MAC address</td>
</tr>
</tbody>
</table>
VNF Descriptor Configuration

### VNFC Virtual Link Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| virtual-link-type | M        | choice-string | Type of virtual Link. It can be one of the following types:  
|                 |          |         | • internal: Choose virtual link from VNF internal VLDs.  
|                 |          |         | • service-vl: Choose virtual link from service level VLDs.                  |
| internal-vl     | M*       | reference | Reference to one of the VLDs from internal VLDs. It is required only when virtual link type is “internal”. |
| service-vl      | M*       | reference | Reference to one of the VLDs defined at service level.                      |

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

### VNFC Volume Instance Group Parameters

VNFC volume instance group is a set of pre-created volumes which can be used in VNF deployment. This is the way pre-created volumes can be used from VIM instead of using volume descriptors from VDU. The volume group is used to supply the pre-created volumes for a VNFC hence a specific group should have enough volume instances to associate with all VNFC instances for a given VNFC. Number of VNFC instances depends on “number of instances” defined for a specific VNFC in a given VNF deployment.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>boot</td>
<td>O</td>
<td>reference</td>
<td>Boot volume. Reference to volume instance descriptor.</td>
</tr>
<tr>
<td>storage</td>
<td>M</td>
<td>array of references</td>
<td>List of pre-created volumes for storage volumes. See the following section for more details.</td>
</tr>
</tbody>
</table>

### Volume Instance Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>volume-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier for volume as it exists on VIM. You can use either volume name or UUID but there should be only one volume instance by that identity.</td>
</tr>
<tr>
<td>type</td>
<td>M</td>
<td>bool</td>
<td>Volume type as exists on VIM.</td>
</tr>
<tr>
<td>bootable</td>
<td>O</td>
<td>bool</td>
<td>Flag to indicate if it is bootable flag. By default its set to “false”. You should have bootable volume to use as boot disk within VNFC.</td>
</tr>
</tbody>
</table>

### VNFC- VDU Descriptor Parameters

VNFC VDU describes the properties of VDU used to create this VNFC instance and has the following parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vdu-id</td>
<td>M</td>
<td>reference</td>
<td>Reference to VDU descriptor used to create this VNFC. See the following section for more details.</td>
</tr>
<tr>
<td>image</td>
<td>O</td>
<td>string</td>
<td>By default VDU image is used from the descriptor. This parameter can be used to override the image for this deployment. For example, you can specify pre-created image from VIM.</td>
</tr>
<tr>
<td>flavor</td>
<td>O</td>
<td>string</td>
<td>Similar to image, by default the UAS creates flavor based on flavor information defined in VDU. This parameter can be used to override VDU flavor for this deployment.</td>
</tr>
</tbody>
</table>
## VNFC Syslog Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>server-type</td>
<td>M</td>
<td>choice string</td>
<td>Server type. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• uas-proxy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• server</td>
</tr>
<tr>
<td>uas-proxy</td>
<td>M*</td>
<td>null</td>
<td>UAS proxy for capturing the syslog of VNFC in UAS. In this case, it is AutoVNF.</td>
</tr>
<tr>
<td>severity</td>
<td>O*</td>
<td>uint8</td>
<td>Severity of syslog messages. This parameter is present if the server type is uas-proxy. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 0: emerg, panic</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 1: alert</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 2: crit</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 3: err.error</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 4: warning, warn</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 5: notice</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 6: info</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 7: debug</td>
</tr>
<tr>
<td>server</td>
<td>M*</td>
<td>Array of server descriptor</td>
<td>List of external servers for capturing the syslogs for the configured VNFC. See the Server Descriptor Parameters for more details.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

## Server Descriptor Parameters

The server is configured to capture the syslog messages of VNFC(s) in the external syslog server(s).

**Note:** The external server(s) should be reachable from the configured VNFC.
### VNF Descriptor Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>ip-address</td>
<td>External server IP address</td>
</tr>
<tr>
<td>port</td>
<td>O</td>
<td>uint16</td>
<td>Port number to be used for sending the syslog. By default, it is set to 514.</td>
</tr>
</tbody>
</table>
| severity   | O        | uint8 | Severity of syslog messages. It can be one of the following:  
  - 0: emerg, panic  
  - 1: alert  
  - 2: crit  
  - 3: err.error  
  - 4: warning, warn  
  - 5: notice  
  - 6: info  
  - 7: debug  
  By default, this value is set to 6. |

### VDU Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vdu-id</td>
<td>M</td>
<td>string (1..16)</td>
<td>Unique ID for this VDU descriptor.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------</td>
<td>--------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>vdu-type</td>
<td>M</td>
<td>choice string</td>
<td>VDU type. It can be one of the following: • automation-service: Cisco Ultra Automation Service • cisco-esc: Cisco Elastic Service Controller • element-manager: Cisco Ultra VNF Element Manager • control function: Cisco UGP Control Function Node • session-function: Cisco UGP Session Function Node • network-function: Cisco UGP Network Function Node • application-function: Cisco UGP Application Function Node • juniper-srx: Juniper vSRX node • cisco-asav: Cisco Virtual ASA (ASAv) • sandvine-pts: Sandvine Policy Traffic Switch (PTS) • sandvine-sbp: Sandvine Subscriber Policy Broker (SPB) • sandvine-sde: Sandvine Service Deliver Engine (SDE) • integra-oam: Openwave Integra OAM • integra-iptm: Openwave Integra IPTM • integra-tcp: Openwave DynaBoost • cps-master: Cisco UCPS Master Node • cps-control: Cisco UCPS Control Node • cps-policy: Cisco UCPS Policy Node • cps-session: Cisco UCPS Session Node • cps-pcrf: Cisco UCPS PCRF Node</td>
</tr>
<tr>
<td>flavor</td>
<td>M</td>
<td>flavor descriptor</td>
<td>VDU Flavor descriptor. See the VDU Flavor Descriptor Parameters section for details.</td>
</tr>
<tr>
<td>upp</td>
<td>O</td>
<td>upp descriptor</td>
<td>Ultra-Packet Processor Attributes. See the UPP Descriptor Parameters section for more details.</td>
</tr>
<tr>
<td>ned</td>
<td>O</td>
<td>array of ned descriptor</td>
<td>List of Network Element Drivers (NED) applicable for this VDU, used for UGP CF VDU. See the NED Descriptor Parameters section for more details.</td>
</tr>
<tr>
<td>login-credential</td>
<td>O*</td>
<td>secure-token</td>
<td>Login credential for the VDU. It is required for UAS modules, ESC, EM, and CF. It is optional for UP.</td>
</tr>
<tr>
<td>netconf-credential</td>
<td>O</td>
<td>secure-token</td>
<td>NETCONF credential to authenticate over API. It is required only for ESC.</td>
</tr>
<tr>
<td>scm</td>
<td>O*</td>
<td>secure-token</td>
<td>Service Configuration Manager Secure token. It is required for VNF-EM and UAS.</td>
</tr>
<tr>
<td>authorized-key</td>
<td>O</td>
<td>secure-token</td>
<td>SSH keys used to login to this VDU.</td>
</tr>
<tr>
<td>configuration</td>
<td>O</td>
<td>array of configuration descriptor</td>
<td>Array of configuration descriptor. See the VDU Configuration Descriptor Parameters section for details.</td>
</tr>
<tr>
<td>image</td>
<td>M</td>
<td>image descriptor</td>
<td>VDU image parameters. See the following section for more details.</td>
</tr>
</tbody>
</table>
### VNF Descriptor Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnf-rack</td>
<td>O</td>
<td>reference</td>
<td>VNF Rack reference. This is used to inherit host-aggregate for placement on VIM.</td>
</tr>
<tr>
<td>vnf-package</td>
<td>O</td>
<td>reference</td>
<td>VNF Package reference to provide VDU image and configuration. See the VDU VNF- Package Parameters section for more details.</td>
</tr>
<tr>
<td>volume</td>
<td>O</td>
<td>array of volume descriptor</td>
<td>List of volume descriptors to be used at VDU level. All the VNFC instances inheriting this VDU will have these set of descriptors. See the VDU Volume Descriptor Parameters for details.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

### VDU Image Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image-type</td>
<td>M</td>
<td>choice string</td>
<td>Image source type. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• url: Image source via network URL.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• package: Image is used from VNF package.</td>
</tr>
<tr>
<td>url</td>
<td>M*</td>
<td>url</td>
<td>External URL to get the image. This is required when image-type is URL.</td>
</tr>
<tr>
<td>vnf-package</td>
<td>M*</td>
<td>empty</td>
<td></td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

### VDU VNF- Package Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>primary</td>
<td>M</td>
<td>image descriptor</td>
<td>Primary image to be used for deployment.</td>
</tr>
<tr>
<td>secondary</td>
<td>O</td>
<td>image descriptor</td>
<td>Secondary image used in the case of any failure while deploying primary. This field is specifically used while updating the image with a “primary” package.</td>
</tr>
</tbody>
</table>

### VDU Volume Descriptor Parameters

These are a set of volume descriptors defined at VDU level. These volumes are used to provide boot disk as well one or more storage disks. You can define volume descriptors at VDU level and any VNFC using this VDU descriptor gets those volumes inherited. The UAS automatically creates these volumes based on descriptors and passes to VNFC instances. Alternatively, you can also use pre-created volumes those are defined at VNFD level.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>volume-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID assigned for this volume descriptor.</td>
</tr>
<tr>
<td>type</td>
<td>M</td>
<td>string</td>
<td>The existing type of volume on VIM.</td>
</tr>
<tr>
<td>size</td>
<td>M</td>
<td>int32</td>
<td>Size of volume in GB.</td>
</tr>
<tr>
<td>bus</td>
<td>O</td>
<td>choice</td>
<td>PCI bus interface that is to be used while attaching the OS disk to the domain. Currently supported bus types are IDE (&quot;ide&quot;), SCSI (&quot;scsi&quot;) and VIRTIO (&quot;virtio&quot;). Default value is set to &quot;ide&quot;.</td>
</tr>
<tr>
<td>bootable</td>
<td>O</td>
<td>bool</td>
<td>Flag to indicate if it is a bootable volume. Default is &quot;false&quot;. This must be true for boot volume. The UAS uses the VDU image to create bootable volume.</td>
</tr>
<tr>
<td>preserve-on-upgrade</td>
<td>O</td>
<td>bool</td>
<td>During upgrade, there is a requirement where volumes need to be preserved as the storage volumes might have data that requires to be preserved during upgrade. If this flag is set to true, the UAS does not destroy the volume during VNF deactivation, instead it preserves it and re-attaches to the VDU after redeployment. That way, storage data is not lost during upgrade. This is specifically useful for VDUs like UGP CF to preserve the Call Data Records (CDRs).</td>
</tr>
</tbody>
</table>

### VDU Configuration Descriptor Parameters

These are a list of day-zero metadata passed to VDU. These day-zero and day-one files are treated as templates, that is, these files can have $ defined variables. The UAS and VNF-EM populate with the values defined at VDU level or VNFD level.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>destination-path</td>
<td>M</td>
<td>string</td>
<td>Destination path. This is the mounted location of the file within VNFC instance inheriting this VDU.</td>
</tr>
<tr>
<td>initialization-variable</td>
<td>O</td>
<td>array of type:value</td>
<td>List of day-zero variable initializations that are common across all instances for this specific configuration file.</td>
</tr>
<tr>
<td>apply-at</td>
<td>O</td>
<td>choice string</td>
<td>Shows how VNF-EM applies this configuration.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• day-zero: Configuration is passed to VDU at day-0 using configuration drive.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• day-one: Configuration is applied at day-one using NED. This is useful when a VDU does not support configuration drive.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>By default, configuration is passed as configuration drive.</td>
</tr>
<tr>
<td>source</td>
<td>M</td>
<td>choice string</td>
<td>Source type. The options are as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• url: UAS receives this file over a network based URL such as http, ftp, etc.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• package: configuration file from the package.</td>
</tr>
<tr>
<td>source-url</td>
<td>M*</td>
<td>url</td>
<td>Source URL for the configuration file. This is required only when source choice is &quot;url&quot;.</td>
</tr>
</tbody>
</table>
### VNF Descriptor Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>package</td>
<td>M*</td>
<td>reference</td>
<td>Reference to one of the configuration files from the VNF Package descriptor. This is only required when source type is “package”.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

### NED Descriptor Parameters

Network Element Driver (NED) is used by VNF-EM to connect to device in order to manage. VNF-EM uses this channel to push configuration to the device, pulls operational state and KPIs from the device.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| ned-type    | O        | choice string| Supported types of Network Element Driver. It can be one of the following:  
- netconf: NETCONF based on network element driver  
- cli: CLI Based on network element driver  
- snmp: SNMP Based network element driver                                                                                                         |
| ned-id      | M        | string       | Unique identifier for NED. It is mandatory only when a NED is associated to the VDU. This NED ID is predefined and VNF-EM uses this ID to identify the device.                                                                                                                   |
| port-number | M        | int16        | Port number used in VNF-EM to mount the device.                                                                                                                                                                                                                               |
| authentication | M     | secure-token | Reference to secure token used to authenticate device.                                                                                                                                                                                                                       |

### UPP Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cores</td>
<td>O</td>
<td>percent</td>
<td>Percentage of virtual CPUs assigned for DPDK task in Ultra Packet Processor. Default value is 30%.</td>
</tr>
<tr>
<td>crypto-cores</td>
<td>O</td>
<td>percent</td>
<td>Percentage of virtual CPUs assigned for DPDK task to handle software crypto. Default value is 0.</td>
</tr>
</tbody>
</table>
| service-mode | O      | choice string | Service type for Ultra- Packet-Gateway (UGP). It can be one of the following:  
- vpc: UPP Packat core for gateway deployments  
- usf: Ultra-Services Framework (Gi-LAN) deployment.  
- epdg: EPDG Gateway  
Default value is “vpc”.                                                                 |
| disable-mcdma | O    | bool   | Disable MCDMA threads. It is disabled on NF/AF cards by default.                                                                                                                                                 |
| enable-vnpu  | O        | bool   | Run DPDK task in IO/VNPU mode for optimum performance. By default, it is “false”.                                                                                                                             |
| disable-numa | O        | bool   | Disable NUMA optimizations forcefully even when multiple sockets are presented to the VM. Default is “false”.                                                                                                    |

### VDU Flavor Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vcpus</td>
<td>M</td>
<td>uint16</td>
<td>Number of virtual CPUs to use.</td>
</tr>
<tr>
<td>ram</td>
<td>M</td>
<td>uint32</td>
<td>Amount of RAM to use (in megabytes).</td>
</tr>
<tr>
<td>Parameter</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>---------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>root- disk</td>
<td>M</td>
<td>uint16</td>
<td>Amount of disk space (in gigabytes) to use for the root (/) partition.</td>
</tr>
<tr>
<td>ephemeral- disk</td>
<td>O</td>
<td>uint16</td>
<td>Amount of disk space (in gigabytes) to use for the ephemeral partition.</td>
</tr>
<tr>
<td>swap- disk</td>
<td>O</td>
<td>uint32</td>
<td>Amount of swap space (in megabytes) to use.</td>
</tr>
<tr>
<td>huge- page- size</td>
<td>O</td>
<td>choice</td>
<td>Huge page size. It can be one of the following options:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>string</td>
<td>• small: The smallest page size is used, for example, 4 KB on x86.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• large: Use large page sizes only for guest RAM, for example, either 2MB, 1GB on x86.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• any: It is left to the compute driver to decide. In this case, the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>libvirt driver might try to find large pages, but fall back to small</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>pages. Other drivers may choose alternate policies if any.</td>
</tr>
<tr>
<td>cpu-policy</td>
<td>O</td>
<td>choice</td>
<td>CPU pinning policy. For the libvirt driver, you can pin the virtual CPUs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>string</td>
<td>(vCPUs) of instances to the hosts physical CPU cores (pCPUs) using</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>properties. You can further refine this by stating how hardware CPU</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>threads in a simultaneous multithreading-based (SMT) architecture be</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>used. These configurations will result in improved per-instance determin-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ism and performance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>It can be any one from the following policies:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• shared: The guest vCPUs will be allowed to freely float across host</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>pCPUs, albeit potentially constrained by NUMA policy.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• dedicated: The guest vCPUs will be strictly pinned to a set of host pCPUs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>In the absence of an explicit vCPU topology request, the drivers typically</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>expose all vCPUs as sockets with one core and one thread. When strict CPU</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>pinning is in effect the guest CPU topology will be setup to match the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>topology of the CPUs to which it is pinned. This option implies an</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>overcommit ratio of 1.0. For example, if a two vCPU guest is pinned to a</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>single host core with two threads, then the guest will get a topology of</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>one socket, one core, and threads.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Default policy is &quot;shared&quot;.</td>
</tr>
<tr>
<td>cpu-thread-pol-</td>
<td>O</td>
<td>choice</td>
<td>CPU thread scheduling policy for hype-threaded CPUs. It can be any one</td>
</tr>
<tr>
<td>icy</td>
<td></td>
<td>string</td>
<td>from the following policies:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• prefer: The host may or may not have an SMT architecture. Where an SMT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>architecture is present, thread siblings are preferred.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• isolate: The host must not have an SMT architecture or must emulate a</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>non-SMT architecture. If the host does not have an SMT architecture, each</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>vCPU is placed on a different core as expected. If the host does have an</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SMT architecture - that is, one or more cores have thread siblings - then</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>each vCPU is placed on a different physical core. No vCPUs from other</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>guests are placed on the same core. All but one thread sibling on each</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>utilized core is therefore guaranteed to be unusable.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• require: The host must have an SMT architecture. Each vCPU is allocated</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>on thread siblings. If the host does not have an SMT architecture, then it</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>is not used. If the host has an SMT architecture, but not enough cores with</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>free thread siblings are available, then scheduling fails.</td>
</tr>
</tbody>
</table>
**VNF Descriptor Configuration**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numa- nodes</td>
<td>O</td>
<td>array of numa descriptor</td>
<td>List of Numa Node Descriptor. See the following section for more details.</td>
</tr>
<tr>
<td>cpu-topology</td>
<td>O</td>
<td>cpu-topology</td>
<td>CPU Topology descriptor. See the following section for more details.</td>
</tr>
<tr>
<td>host-aggregate</td>
<td>O</td>
<td>reference</td>
<td>Reference to host-aggregate from VNF Rack.</td>
</tr>
</tbody>
</table>

**NUMA Descriptor Parameters**

Specifies the number of host NUMA nodes to restrict execution of instance vCPU threads. If not specified, the vCPU threads can run on any number of the available host NUMA nodes.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numa-node-id</td>
<td>M</td>
<td>string</td>
<td>NUMA node ID.</td>
</tr>
<tr>
<td>flavor-cores</td>
<td>O</td>
<td>array of int32</td>
<td>A list of instance vCPUs to map to instance NUMA node are evenly divided among available NUMA nodes. If not, it is divided among available NUMA nodes.</td>
</tr>
<tr>
<td>flavor-memory</td>
<td>O</td>
<td>int32</td>
<td>The number of MB of instance memory to map to instance NUMA node N. If not specified, memory is evenly divided among available NUMA nodes.</td>
</tr>
</tbody>
</table>

**CPU Topology Descriptor Parameters**

For the libvirt driver, you can define the topology of the processors in the virtual machine using properties. Then, select the properties with max limit using the image properties.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>flavor-sockets</td>
<td>O</td>
<td>uint32</td>
<td>The number of sockets for the guest VM. By default, this is set to the number of vCPUs requested.</td>
</tr>
<tr>
<td>flavor-cores</td>
<td>O</td>
<td>uint32</td>
<td>The number of cores per socket for the guest VM. By default, this is set to 1.</td>
</tr>
<tr>
<td>flavor-threads</td>
<td>O</td>
<td>uint32</td>
<td>The number of threads per core for the guest VM. By default, this is set to 1.</td>
</tr>
</tbody>
</table>
**USF Descriptor Parameters**

- **Endpoint set** (Host, VNFC, connection-point)
- **VDU Path** (P1, P2, P3, P4, P5, P6, P7) set of endpoints
- **Networking Forwarding Path (NFP)** set of VDU paths.
  - NFP1: P1 -> P5 -> P6, NFP2: P2 -> P3 -> P6, NFP3: P2 -> P3 -> P7

**Service Function Chain**, set of Network Forwarding Path.
SFC1: NFP1, NFP2, NFP3  SFC2: NFVP1, SFC3: NFP1, NFP3

USF is implemented as a separate element group and depending on service chaining, there can be one or more element groups to provide service chaining. There are two kinds of deployment:

- **One Element group per host** - In this mode, all VDUs of the element group are placed on the same host. Multiple instances of the same element group are deployed to meet required scale and to provide redundancy. For more details, see the Ultra Services Framework Deployment Guide.

- **One Element group spawned across multiple hosts** - In this mode, VDUs pertaining to one element group are placed across multiple hosts.

The decision whether to use single host element group or multi-host element group depends on several factors including:

- **Subscriber stickiness** - When multiple instances of element group are deployed to meet the required scale, there are several paths called Network Forwarding Path (NFP) to provide the same service. The selection of the NFP path depends on factors like stickiness where it is required to make sure flows pertaining to a given subscriber use the same forwarding path. The other criteria might be based on load at a given time.
The availability of resources to deploy a given service - There might be situations where single server cannot provide enough resources to deploy a network service hence you should design service element group across multiple hosts.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>service-element-group-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID assigned to service element group.</td>
</tr>
<tr>
<td>number-of-instances</td>
<td>O</td>
<td>int32</td>
<td>Number of instances for this element group. By default, the value is set to 1.</td>
</tr>
<tr>
<td>host</td>
<td>M</td>
<td>array of host descriptor</td>
<td>List of one or more host descriptors to layout element group. See the following section for more details.</td>
</tr>
</tbody>
</table>

**USF Host Descriptor Parameters**

This is a logical host mapping to overlay USF element group.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host-id</td>
<td>M</td>
<td>string</td>
<td>Logical host ID.</td>
</tr>
<tr>
<td>vnfcb</td>
<td>M</td>
<td>array of VNFC</td>
<td>List of VNFC used to form the service element group.</td>
</tr>
<tr>
<td>vdu-path</td>
<td>M</td>
<td>array of vdu-path descriptor</td>
<td>List of VDU paths. See the following section for more details.</td>
</tr>
<tr>
<td>network-forwarding-path</td>
<td>M</td>
<td>array of network-forwarding path descriptor</td>
<td>List of network forwarding path where each Network Forwarding Path is a set of VDU path.</td>
</tr>
<tr>
<td>service-function-chain</td>
<td>M</td>
<td>array of service function chain descriptor</td>
<td>List of Service function chain descriptors where each Service Function Chain (SFC) descriptor is a set of Network Forwarding Paths (NFP).</td>
</tr>
</tbody>
</table>
### USD Network Forwarding Path (NFP) Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nfp-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID assigned to Network Forwarding Path (NFP).</td>
</tr>
<tr>
<td>ref-type</td>
<td>M</td>
<td>choice</td>
<td>Reference type. This NFP either refers to another NFP “vdu-path-refs-from-nfp” or new NFP created with the set of VDU Paths “vdu-path-refs”.</td>
</tr>
<tr>
<td>vdu-path-refs-from-nfp</td>
<td>M*</td>
<td>reference</td>
<td>Inherit network-forwarding-path from the existing NFP. This is a required configuration when ref-type is set to “vdu-path-refs-from-nfp”.</td>
</tr>
<tr>
<td>vdu-path-ref</td>
<td>M*</td>
<td>array of reference</td>
<td>Create new NFP based on set of VDU path.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

### USF VDU Path Descriptor Parameters

This is used to define a set of endpoints. An endpoint is named tuple to refer to a connection point on a specific VNFC placed to a specific host.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vdu-path-id</td>
<td>M</td>
<td>string</td>
<td>Unique VDU path ID.</td>
</tr>
<tr>
<td>endpoint</td>
<td>M</td>
<td>array of endpoint descriptor</td>
<td>One or more set of endpoints to create this logic VDU path. See the following section for more details.</td>
</tr>
</tbody>
</table>

### USF VDU Path Endpoint Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host-ref</td>
<td>M</td>
<td>reference</td>
<td>Reference to logical host.</td>
</tr>
<tr>
<td>vdu</td>
<td>M</td>
<td>reference</td>
<td>Reference to VNFC within that logical host.</td>
</tr>
<tr>
<td>if-ref</td>
<td>M</td>
<td>reference</td>
<td>Reference to connection point from the VNFC within this endpoint domain.</td>
</tr>
</tbody>
</table>

### USF Service Function Chain Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sfc-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID assigned to this Service Function Chain.</td>
</tr>
<tr>
<td>network-forwarding-path-ref</td>
<td>M</td>
<td>array of reference</td>
<td>List of Network Forwarding Path (NFP) reference to form the Service Function Chain (SFC).</td>
</tr>
</tbody>
</table>
Fault Management Descriptor Configuration

Fault management system defines a set of descriptors to monitor the different aspects of overall system. Following are the set of functional domains what UAS is targeted to monitor.

- VIM services
- Physical servers
- Router / switches
- VNFs including UAS

Monitoring Services

The UAS provides a set of functions to monitor different components including:

- Health of OpenStack services
- Health of UCS servers
- Health of VNFC instances
- HA state of VNF cluster

Log Collection and Aggregation

Syslog configuration for UCS and Openstack is part of FMD which is pushed through FMD activation. The UAS aggregates these logs in AutoIT and sends them to external remote syslog server. Also, UCS and Openstack can be configured to push the syslog to external remote syslog server if the external syslog server is reachable from UCS and Openstack.

Syslog configuration is part of VNF which is automatically pushed to VDUs. Syslog is a transport used to pull all the logs from different modules. The UAS aggregates these logs at centralized location as part of AutoIT and send these logs to external remote syslog server. Also, the VDUs can directly push the syslog to external remote syslog server if the external syslog server is reachable from the configured VDU.

This is a sample SNMP notification generated by fault management process for CEPH fault.

```
--SNMP logs collected at NMS--
Nov 20 15:51:36 localhost snmptrapd[1563]: 2017-11-20 15:51:36 <UNKNOWN> [UDP: (10.105.248.224):55633->[10.105.248.149]:162]:#012DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (1) 0:00:00.01011SNMPv2-MIB::snmpTrapOID.0 = OID: SNMPv2-SMI::enterprises.9.9.849.1.1.1.3.15.98.108.114.110.115.100.49.45.98.108.114.102.109.100.49.24 = STRING: "b1rnfvpop#011SNMPv2-SMI::enterprises.9.9.849.1.1.1.5.15.98.108.114.110.115.100.49.45.98.108.114.102.109.100.49.24 = INTEGER: 3#011SNMPv2-SMI::enterprises.9.9.849.1.1.1.6.15.98.108.114.110.115.100.49.45.98.108.114.102.109.100.49.24 = STRING: "blrt-est-compute-1: ceph:status"#011SNMPv2-SMI::enterprises.9.9.849.1.1.1.7.15.98.108.114.110.115.100.49.45.98.108.114.102.109.100.49.24 = INTEGER: 1#011SNMPv2-SMI::enterprises.9.9.849.1.1.1.8.15.98.108.114.110.115.100.49.45.98.108.114.102.109.100.49.24 = INTEGER:
```
Fault Management Descriptor Configuration

The UAS creates fault events as operational data in the following format:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fmr-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID assigned to Fault Management Descriptor.</td>
</tr>
<tr>
<td>snmp-engine-id</td>
<td>O</td>
<td>string</td>
<td>SNMP engine ID which is used by local SNMP agent.</td>
</tr>
<tr>
<td>domain</td>
<td>O</td>
<td>array of domain summary</td>
<td>List of number of active and clear faults for all domains.</td>
</tr>
<tr>
<td>fault</td>
<td>O</td>
<td>array of fault entries</td>
<td>List of active faults from NFVI, VIM, UAS domain.</td>
</tr>
<tr>
<td>vim-service</td>
<td>O</td>
<td>array of VIM Services status</td>
<td>List of VIM services health status descriptor.</td>
</tr>
<tr>
<td>nfvi-node</td>
<td>O</td>
<td>array of NFVI Node Status</td>
<td>List of NFVI node health status descriptor.</td>
</tr>
<tr>
<td>nsd</td>
<td>O</td>
<td>array of Network Service Status</td>
<td>List of network service health status descriptor.</td>
</tr>
</tbody>
</table>

Domain Summary Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain-id</td>
<td>M</td>
<td>choice string</td>
<td>Fault domain on which this fault is occurring. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• hardware: Hardware including servers, L2/L3 elements</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vim: VIM manager such as OpenStack</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• uas: Ultra Automation Services modules</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vnfm: VNF Manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vnf-em: Ultra VNF Element Manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vnf: Ultra VNF</td>
</tr>
<tr>
<td>active-faults</td>
<td>O</td>
<td>uint32</td>
<td>Number of active fault count for the domain.</td>
</tr>
<tr>
<td>clear-faults</td>
<td>O</td>
<td>uint32</td>
<td>Number of clear fault count for the domain.</td>
</tr>
</tbody>
</table>
## Fault Entry Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>index</td>
<td>M</td>
<td>uint32</td>
<td>Uniquely identifies a specific instance of a fault at network service level.</td>
</tr>
<tr>
<td>nfv-id</td>
<td>M</td>
<td>string (1..512)</td>
<td>Uniquely identifies the Ultra-M PoD on which this fault is occurring.</td>
</tr>
</tbody>
</table>
| domain    | M        | choice string | Fault domain on which this fault is occurring. It can be one of the following:  
  - hardware: Hardware including servers, L2/L3 elements  
  - vim: VIM manager such as OpenStack  
  - uas: Ultra Automation Services modules  
  - vnfm: VNF Manager  
  - vnf-em: Ultra VNF Element Manager  
  - vnf: Ultra VNF |
| source    | M        | string (1..512) | Uniquely identifies the resource with the fault domain where this fault is occurring. |
| time      | M        | yang:date-and-time | The date and time when the fault was occurred. |
| severity  | M        | choice string | Perceived severity of the fault. It can be one of the following:  
  - emergency: System level FAULT impact  
  - critical: Critical fault specific to VNF/service.  
  - major: Component level failure within VNF/service.  
  - alert: Warning condition for a service/VNF, may eventually impact service.  
  - informational: Informational only, does not impact service. |
| code      | M        | choice string | Code which uniquely identifies the fault class. The options are as follows:  
  - other: Other events.  
  - network-connectivity: Network Connectivity failure events.  
  - resource-usage: Resource Usage Exhausted event.  
  - hardware-failure: Hardware Failure Events.  
  - configuration: Config error events.  
  - service-failure: Process/Service failures. |
| description | M         | string (1..2048) | Human-readable message providing details about the fault. |
### VIM Service Status Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>service-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier of VIM service.</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Service health status. The following are the status indicators:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• :-) – Healthy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• XXX – Faulty</td>
</tr>
<tr>
<td>last-event</td>
<td>M</td>
<td>string</td>
<td>Last fault event occurred on this service.</td>
</tr>
</tbody>
</table>

### NFVI Node Status Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>node-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier of NFVI node.</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Service health status. The following are the status indicators:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• :-) – Healthy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• XXX – Faulty</td>
</tr>
<tr>
<td>last-event</td>
<td>M</td>
<td>string</td>
<td>Last fault event occurred on this service.</td>
</tr>
</tbody>
</table>

### Network Service Status Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsd-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier of network service.</td>
</tr>
<tr>
<td>vnfd</td>
<td>O</td>
<td>array of VNFD health status</td>
<td>List of VNFD health status.</td>
</tr>
<tr>
<td>vnfc</td>
<td>O</td>
<td>array of VNFC health status</td>
<td>List of VNFC health status.</td>
</tr>
</tbody>
</table>
Fault Management Descriptor Configuration

**VNFD Status Descriptor Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnfd-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier of VNFD.</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Service health status. The following are the status indicators:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• :-) – Healthy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• XXX – Faulty</td>
</tr>
<tr>
<td>last-event</td>
<td>M</td>
<td>string</td>
<td>Last fault event occurred on this service.</td>
</tr>
</tbody>
</table>

**VNFC Status Descriptor Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnfc-id</td>
<td>M</td>
<td>string</td>
<td>Unique identifier of VNFC.</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Service health status. The following are the status indicators:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• :-) – Healthy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• XXX – Faulty</td>
</tr>
<tr>
<td>last-event</td>
<td>M</td>
<td>string</td>
<td>Last fault event occurred on this service.</td>
</tr>
</tbody>
</table>

**Fault Notification Descriptor Parameters**

The UAS can send the service faults over NETCONF notification.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fmr-id</td>
<td>M</td>
<td>string</td>
<td>Unique ID assigned to Fault Management Descriptor.</td>
</tr>
<tr>
<td>index</td>
<td>M</td>
<td>uint32</td>
<td>Uniquely identifies a specific instance of a fault at network service level.</td>
</tr>
<tr>
<td>nfv-id</td>
<td>M</td>
<td>string</td>
<td>(1..512) Uniquely identifies the Ultra-M PoD on which this fault is occurring.</td>
</tr>
<tr>
<td>domain</td>
<td>M</td>
<td>choice string</td>
<td>Fault domain on which this fault is occurring. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• hardware: Hardware including servers, L2/L3 elements</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vim: VIM manager such as OpenStack</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• uas: Ultra Automation Services modules</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vnfm: VNF Manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vnf-em: Ultra VNF Element Manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• vnf: Ultra VNF</td>
</tr>
<tr>
<td>Parameter</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>source</td>
<td>M</td>
<td>string (1..512)</td>
<td>Uniquely identifies the resource with the fault domain where this fault is occurring.</td>
</tr>
<tr>
<td>time</td>
<td>M</td>
<td>yang:date-and-time</td>
<td>The date and time when the fault was occurred.</td>
</tr>
<tr>
<td>severity</td>
<td>choice</td>
<td>string M</td>
<td>Perceived severity of the fault. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• emergency: System level FAULT impact</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• critical: Critical fault specific to VNF/service.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• major: Component level failure within VNF/service.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• alert: Warning condition for a service/VNF, may eventually impact service.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• informational: Informational only, does not impact service.</td>
</tr>
<tr>
<td>code</td>
<td>choice</td>
<td>string M</td>
<td>Code which uniquely identifies the fault class. The options are as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• other: Other events.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• network-connectivity: Network Connectivity failure events.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• resource-usage: Resource Usage Exhausted event.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• resource-threshold: Resource Threshold crossing alarms.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• hardware-failure: Hardware Failure Events.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• security-violation: Security violation alerts.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• configuration: Config error events.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• service-failure: Process/Service failures.</td>
</tr>
<tr>
<td>description</td>
<td>M</td>
<td>string (1..2048)</td>
<td>Human-readable message providing details about the fault.</td>
</tr>
</tbody>
</table>

This is a sample NETCONF notification generated by fault management process for CEPH fault.

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-29T12:43:01.432732+00:00</eventTime>
  <fault-management-event xmlns="http://www.cisco.com/usp/nfv/usp-fmr">
    <fmr-id>blrnsd1-blrfmd1</fmr-id>
    <index>24</index>
    <nfv-id>blrnfvipop1</nfv-id>
    <domain>vim</domain>
    <source>blrnsv-compute-1:ceph:status</source>
    <time>2017-11-20T20:51:35.016748-00:00</time>
    <severity>emergency</severity>
    <code>service-failure</code>
    <description>CEPH Status not healthy</description>
  </fault-management-event>
</notification>
```
Fault Management Descriptor Configuration

Operations

Configuring FMD

**Operation:** edit-config

**Namespace:** fmd xmlns="http://www.cisco.com/usf/usf-fmd"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fmd-id</td>
<td>M</td>
<td>string (1..16)</td>
<td>Unique ID assigned to Fault Management Descriptor.</td>
</tr>
<tr>
<td>domain</td>
<td>O</td>
<td>array of domain descriptor</td>
<td>List of domain configuration used to monitor faults and configure syslog.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>See the Domain Descriptor Parameters for more details. By default, all the domains are enabled for monitoring faults.</td>
</tr>
<tr>
<td>severity</td>
<td>O</td>
<td>choice string</td>
<td>Severity level to control the fault reporting. By default, it is set to “informational”.</td>
</tr>
<tr>
<td>vim-type</td>
<td>O</td>
<td>choice string</td>
<td>Type of virtualized infrastructure manager deployment. Allowed values are the following.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• OSP10 – Redhat based Openstack Platform 10 VIM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• CVIM – Cisco Virtualised Infrastructure Manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• VCD – VMWare Cloud Director VIM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>By default, it is set to “OSP10”.</td>
</tr>
<tr>
<td>snmp</td>
<td>O</td>
<td>descriptor</td>
<td>SNMP parameters configuration. See the SNMP Descriptor Parameters for more details.</td>
</tr>
<tr>
<td>vim-param</td>
<td>O</td>
<td>vim-param descriptor</td>
<td>VIM level fault monitoring configuration. See the VIM Descriptor Parameters for more details.</td>
</tr>
</tbody>
</table>

Domain Descriptor Parameters

The UAS monitors the faults and configures remote syslog for a specific domain.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>M</td>
<td>string</td>
<td>Fault domain to enable monitoring faults and syslog.</td>
</tr>
<tr>
<td>monitoring</td>
<td>O</td>
<td>monitoring descriptor</td>
<td>Monitoring parameters. See the Monitor Descriptor Parameters for more details.</td>
</tr>
<tr>
<td>syslog</td>
<td>O</td>
<td>syslog descriptor</td>
<td>Remote syslog parameters. Applicable to only hardware and vim domain. See the Syslog Descriptor Parameters for more details.</td>
</tr>
</tbody>
</table>
### Monitoring Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enabled</td>
<td>O</td>
<td>boolean</td>
<td>Enables/disables monitoring the faults. By default, it is set to &quot;true&quot;.</td>
</tr>
<tr>
<td>suppress-hw-af-</td>
<td>O</td>
<td>Array of strings</td>
<td>The affected DN for which faults should not be raised. Applicable to only hardware domain.</td>
</tr>
<tr>
<td>fected-dn</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>suppress-hw-</td>
<td>O</td>
<td>Array of strings</td>
<td>The fault IDs for which faults should not be raised. Applicable to only hardware domain.</td>
</tr>
<tr>
<td>fault-id</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>suppress-uas-fault</td>
<td>O</td>
<td>Array of strings</td>
<td>This can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• overall: Applicable to all domains</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• api-endpoint: Applicable to only vnf-EM domain</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• ha-event: Applicable to uas and vnf-EM domain</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• cluster-ha: Applicable to uas and vnf-EM domain</td>
</tr>
<tr>
<td>vim-param</td>
<td>O</td>
<td>vim-param descriptor</td>
<td>VIM level fault monitoring configuration. See the <a href="#">VIM Descriptor Parameters</a> for more details.</td>
</tr>
</tbody>
</table>

### VIM Descriptor Parameters

These are the set of parameters that can be customized for UAS to monitor VIM. Most of these parameters are built around OpenStack as that’s the only VIM UAS supports currently.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>controller-count</td>
<td>O</td>
<td>uint8</td>
<td>Number of controller nodes. By default, the value is set to 3. UAS uses this configuration to monitor OpenStack cluster via PCS.</td>
</tr>
<tr>
<td>disk-threshold</td>
<td>O</td>
<td>percent</td>
<td>Threshold to generate events if disk usage goes beyond this threshold. UAS uses this threshold configuration to monitor disks via CEPH.</td>
</tr>
<tr>
<td>osd-count</td>
<td>O</td>
<td>uint16</td>
<td>Number of OSD nodes which are hosting storage. UAS uses this configuration to monitor storage overall health via CEPH.</td>
</tr>
<tr>
<td>ntp-skew-threshold</td>
<td>O</td>
<td>uint16</td>
<td>Maximum allowed NTP skew in mili-seconds. UAS monitors NTP time synchronization and it will report the issue if NTP skew exceeds than this threshold. Default value is set to 100ms.</td>
</tr>
<tr>
<td>module</td>
<td>O</td>
<td>array of strings</td>
<td>List of VIM modules to monitor the health. The UAS has the following set of modules and it monitors the health for each module by default.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• ceph</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• cinder</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• nova</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• pcs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• rabbitmqctl</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• neutron</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• ntpdc</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• systemctl</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>systemctl is used to monitor the health of different services via system-d bus.</td>
</tr>
</tbody>
</table>


List of services on different types of nodes. Following are the set of services that UAS monitors by default via system-d bus. You can add/delete services to/from the list for a given type of node. You can enable/disable the class of service on need basis.

<table>
<thead>
<tr>
<th>service</th>
<th>O</th>
<th>array of strings</th>
</tr>
</thead>
<tbody>
<tr>
<td>controller:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cinder</td>
<td></td>
<td></td>
</tr>
<tr>
<td>glance</td>
<td></td>
<td></td>
</tr>
<tr>
<td>heat</td>
<td></td>
<td></td>
</tr>
<tr>
<td>heat-api</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nova swift</td>
<td></td>
<td></td>
</tr>
<tr>
<td>swift-account</td>
<td></td>
<td></td>
</tr>
<tr>
<td>swift-container</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ntpd</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mongod</td>
<td></td>
<td></td>
</tr>
<tr>
<td>memcached</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-dhcp-agent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-l3-agent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-metadata-agent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-openvswitch-agent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-server</td>
<td></td>
<td></td>
</tr>
<tr>
<td>httpd</td>
<td></td>
<td></td>
</tr>
<tr>
<td>compute:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ceph-mon.target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ceph-radosgw.target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ceph.target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>openvswitch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-sriov-nic-agent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-openvswitch-agent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ntpd</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nova-compute</td>
<td></td>
<td></td>
</tr>
<tr>
<td>libvirtd</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OSD compute:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ceph-mon.target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ceph-radosgw.target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ceph.target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>openvswitch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-sriov-nic-agent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>neutron-openvswitch-agent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ntpd</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nova-compute</td>
<td></td>
<td></td>
</tr>
<tr>
<td>libvirtd</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Syslog Descriptor Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>server-type</td>
<td>M</td>
<td>choice string</td>
<td>Server type. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• uas-proxy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• server</td>
</tr>
<tr>
<td>uas-proxy</td>
<td>M*</td>
<td>null</td>
<td>UAS proxy for capturing the syslog of configured domain in UAS. In this case, it is AutoIT.</td>
</tr>
<tr>
<td>severity</td>
<td>O*</td>
<td>uint8</td>
<td>Severity of syslog messages. This parameter is present if the server type is uas-proxy. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 0: emerg, panic</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 1: alert</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 2: crit</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 3: err.error</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 4: warning, warn</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 5: notice</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 6: info</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 7: debug</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>By default, this value is set to 6.</td>
</tr>
<tr>
<td>server</td>
<td>M*</td>
<td>Array of server descriptor</td>
<td>List of external servers for capturing the syslogs for the configured domain. See the following table for more details.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

Server Descriptor Parameters

The server is configured to capture the syslog messages of configured domain in the external syslog server(s).

**Note:** The external server(s) should be reachable from the component(s) of configured domain.
Fault Management Descriptor Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>ip-address</td>
<td>External server IP address</td>
</tr>
<tr>
<td>port</td>
<td>O</td>
<td>uint16</td>
<td>Port number to be used for sending the syslog. By default, it is set to 514.</td>
</tr>
</tbody>
</table>
| severity  | O        | uint8    | Severity of syslog messages. It can be one of the following:  
- 0: emerg, panic  
- 1: alert  
- 2: crit  
- 3: err, error  
- 4: warning, warn  
- 5: notice  
- 6: info  
- 7: debug  
By default, this value is set to 6. |

SNMP Descriptor Parameters

The UAS sends the service faults over SNMP via SNMP traps and notifications. The SNMP CISCO-ULTRAM-MIB defines the protocol OIDs to send SNMP traps. The UAS also provides backend for SNMP agent and the remote SNMP manager fetches the faults using SNMP.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enterprise-id</td>
<td>M</td>
<td>uint16</td>
<td>IANA assigned enterprise number.</td>
</tr>
<tr>
<td>v2c</td>
<td>O</td>
<td>SNMPV2c descriptor</td>
<td>SNMP V2 parameters.</td>
</tr>
<tr>
<td>v3</td>
<td>O</td>
<td>SNMPV3 Descriptor</td>
<td>SNMP V3 parameters</td>
</tr>
</tbody>
</table>

SNMPv2 Configuration Descriptor Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enabled</td>
<td>O</td>
<td>bool</td>
<td>Enable SNMPv2 based agent.</td>
</tr>
<tr>
<td>community</td>
<td>O</td>
<td>string (1..32)</td>
<td>Community string to access SNMP agent via SNMP V2C community based authentication.</td>
</tr>
<tr>
<td>target</td>
<td>O</td>
<td>array of SNMP Targets</td>
<td>List of SNMP targets to receive SNMP traps for faults. See the following section for more details.</td>
</tr>
</tbody>
</table>

SNMPv2c Target Descriptor Parameters:
### Fault Management Descriptor Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>ip-address</td>
<td>IP address of remote SNMP NMS server.</td>
</tr>
<tr>
<td>community</td>
<td>O</td>
<td>string (1..32)</td>
<td>Community string to access authenticate trap server for SNMPv2c. By default community string is set to “public”.</td>
</tr>
<tr>
<td>port</td>
<td>O</td>
<td>uint16</td>
<td>Port number to send trap to remove end. By default port is set to standard SNMP trap port - 162.</td>
</tr>
</tbody>
</table>

### SNMPv3 Configuration Descriptor Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enabled</td>
<td>O</td>
<td>bool</td>
<td>Enable/disable SNMP V3 based agent.</td>
</tr>
<tr>
<td>user</td>
<td>M</td>
<td>reference</td>
<td>Reference to SNMP user descriptor for V3 agent. See the following section for more details.</td>
</tr>
<tr>
<td>target</td>
<td>O</td>
<td>array of SNMPv3 target descriptors</td>
<td>List of SNMPV3 descriptors to receive V3 traps.</td>
</tr>
</tbody>
</table>

### SNMPv3 User Descriptor Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>M</td>
<td>string (1..16)</td>
<td>SNMP user name.</td>
</tr>
<tr>
<td>auth</td>
<td>O</td>
<td>SNMPV3 authentication descriptor</td>
<td>SNMPV3 authentication parameter.</td>
</tr>
<tr>
<td>priv</td>
<td>O</td>
<td>SNMPV3 privacy parameters</td>
<td>SNMPV3 privacy parameters.</td>
</tr>
</tbody>
</table>

### SNMPv3 Authentication Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>protocol</td>
<td>M</td>
<td>choice string</td>
<td>Authentication protocol. It can be one of the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• none</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• md5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• sha-96</td>
</tr>
<tr>
<td>none</td>
<td>M*</td>
<td>string</td>
<td>None. Authentication mechanism is not used.</td>
</tr>
<tr>
<td>md5</td>
<td>M*</td>
<td>string</td>
<td>HMAC- MD5- 96 Digest authentication protocol key. It is only required when the selected protocol is &quot;md5&quot;.</td>
</tr>
<tr>
<td>sha-96</td>
<td>M*</td>
<td>string</td>
<td>HMAC- SHA- 96 Digest authentication protocol key. It is only required when the selected protocol is “sha-96”.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.
SNMPv3 Privacy Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>protocol</td>
<td>M</td>
<td>choice string</td>
<td>Encryption protocol. It should be one of the choices from “none”, “cbc-des”, “des3ede”, “aes-cfb-128”, “aes-cfb-192” or “aes-cfb-256”.</td>
</tr>
<tr>
<td>None</td>
<td>M*</td>
<td>string</td>
<td>None. Encryption mechanism is not used.</td>
</tr>
<tr>
<td>cbc-des</td>
<td>M*</td>
<td>string</td>
<td>Encryption key to be used with CBC-DES symmetric encryption protocol. It is only required when the selected protocol is “des3-cbc”.</td>
</tr>
<tr>
<td>des3ede</td>
<td>M*</td>
<td>string</td>
<td>Encryption key to be used with 3DES-EDÈ symmetric encryption protocol. It is only required when the selected protocol is “des3-cbc”.</td>
</tr>
<tr>
<td>aes-cfb-128</td>
<td>M*</td>
<td>string</td>
<td>Encryption key to be used with AEC CFB 128 bits mode. It is only required when the selected protocol is “aes-cfb-128”.</td>
</tr>
<tr>
<td>aes-cfb-192</td>
<td>M*</td>
<td>string</td>
<td>Encryption key to be used with AEC CFB 192 bits mode. It is only required when the selected protocol is “aes-cfb-192”.</td>
</tr>
<tr>
<td>aes-cfb-256</td>
<td>M*</td>
<td>string</td>
<td>Encryption key to be used with AEC CFB 256 bits mode. It is only required when the selected protocol is “aes-cfb-256”.</td>
</tr>
</tbody>
</table>

* - Denotes the default requirement of the attribute. However, it can be modified from (O)ptional to (M)andatory or (M)andatory to (O)ptional depending on use case scenario.

SNMPv3 Target Descriptor Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>M</td>
<td>ip-address</td>
<td>IP address of remote SNMP NMS server.</td>
</tr>
<tr>
<td>user</td>
<td>M</td>
<td>reference</td>
<td>Reference to SNMP user.</td>
</tr>
<tr>
<td>port</td>
<td>O</td>
<td>uint16</td>
<td>Port number to send trap to remove end. By default port is set to standard SNMP trap port - 162.</td>
</tr>
</tbody>
</table>

Example RPC

```xml
<nc:rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:f49e76f0-bebf-4597-8eed-7660ab9996f7"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <config>
      <nsd xmlns="http://www.cisco.com/usp/nfv/usp-nads">
        <nsd id>blrnsd1</nsd id>
        <fmd>blrfmd1</fmd>
      </nsd>
      <snmp-user xmlns="http://www.cisco.com/usp/nfv/usp-fmd">
        <name>test</name>
      </snmp-user>
    </config>
  </nc:edit-config>
</nc:rpc>
```
<auth>
  <md5>Cisco@123</md5>
</auth>
<priv>
  <cbc-des>Cisco@123cbc-des</cbc-des>
</priv>
</snmp-user>
<fmd xmlns="http://www.cisco.com/usp/nfv/usf-fmd" fmd-blr="fmd-id">
  <snmp>
    <enterprise-id>9</enterprise-id>
    <v2c>
      <enabled>true</enabled>
      <target>
        <ip-address>10.105.159.172</ip-address>
        <port>162</port>
        <community>public</community>
      </target>
    </v2c>
    <v3>
      <enabled>true</enabled>
      <user>test</user>
      <target>
        <ip-address>10.105.159.172</ip-address>
        <port>162</port>
        <user>test</user>
      </target>
    </v3>
  </snmp>
</fmd>
</config>
</nc:edit-config>
</nc:rpc>

Example Output

  <ok/>
</rpc-reply>
Fault Management Descriptor Deployment

This section describes the parameters that can be used to activate the FMD.

Operations

Activating FMD

**Operation:** nsd:activate

**Namespace:** nsd:activate xmlns:nsd="http://www.cisco.com/usp/nfv/usp-nsds"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fmd</td>
<td>0</td>
<td>reference</td>
<td>Reference to Fault Management Descriptor.</td>
</tr>
</tbody>
</table>

**Example RPC**

```xml
<nc:rpc message-id="urn:uuid:a436e985-14ef-4842-8b08-f3dc283b4c18"
        xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
        <nsd:nsd>blrnsd1</nsd:nsd>
        <nsd:fmd>blrfmd1</nsd:fmd>
    </nsd:activate>
</nc:rpc>
```

**Example Output**

```xml
    <transaction-id xmlns="http://www.cisco.com/usp/nfv/usp-nsds">1511959320-398539</transaction-id>
</rpc-reply>
```

Obtaining Notifications Generated During FMD Activation

**Operation:** get

**Namespace:** fmd-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper"

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Required</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance-id</td>
<td>M</td>
<td>string</td>
<td>A unique ID assigned to this descriptor instance. The same ID is used to</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>create operational record for this instance. Hence, you can pull</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>operational record for this instance at a given time.</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Required</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>descriptor-id</td>
<td>M</td>
<td>string</td>
<td>Configuration descriptor ID. This is the descriptor ID, UAS is acting on and this is being provided as part of the request. This descriptor ID has the associated configuration.</td>
</tr>
<tr>
<td>transaction-id</td>
<td>M</td>
<td>string</td>
<td>Unique transaction ID assigned to this instantiation job. You can use this transaction ID to pull details about this request.</td>
</tr>
<tr>
<td>operation-type</td>
<td>M</td>
<td>choice string</td>
<td>Type of operation (same as type of transaction)</td>
</tr>
<tr>
<td>status</td>
<td>M</td>
<td>choice string</td>
<td>Current deployment status (start, in-progress, error, or success).</td>
</tr>
<tr>
<td>steps-total</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps required to be performed.</td>
</tr>
<tr>
<td>steps-completed</td>
<td>O</td>
<td>uint16</td>
<td>Total number of steps completed.</td>
</tr>
<tr>
<td>version</td>
<td>O</td>
<td>string</td>
<td>Deployment version.</td>
</tr>
</tbody>
</table>

**Example Notification**

```xml
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-29T12:42:00.477126+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>blrnsd1-instance</instance-id>
    <descriptor-id>blrnsd1</descriptor-id>
    <transaction-id>1511959320-398539</transaction-id>
    <operation-type>activate-ns-deployment</operation-type>
    <status>requested</status>
  </ns-deployment-event>
</notification>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-29T12:42:00.737925+00:00</eventTime>
  <fmd-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>blrnsd1-blrfmd1</instance-id>
    <descriptor-id>blrfmd1</descriptor-id>
    <transaction-id>1511959320-642693</transaction-id>
    <operation-type>activate-fmd</operation-type>
    <status>requested</status>
  </fmd-deployment-event>
</notification>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-29T12:42:00.863248+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>blrnsd1-instance</instance-id>
    <descriptor-id>blrnsd1</descriptor-id>
    <transaction-id>1511959320-398539</transaction-id>
    <operation-type>activate-ns-deployment</operation-type>
    <status>in-progress</status>
  </ns-deployment-event>
</notification>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-29T12:42:01.139044+00:00</eventTime>
  <fmd-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>blrnsd1-blrfmd1</instance-id>
    <descriptor-id>blrfmd1</descriptor-id>
    <transaction-id>1511959320-642693</transaction-id>
    <operation-type>activate-fmd</operation-type>
    <status>in-progress</status>
  </fmd-deployment-event>
</notification>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-29T12:42:01.289693+00:00</eventTime>
</notification>
```
<fmd-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
  <instance-id>blrnsd1-blrfmd1</instance-id>
  <descriptor-id>blrfmd1</descriptor-id>
  <transaction-id>1511959320-642693</transaction-id>
  <operation-type>activate-fmd</operation-type>
  <status>success</status>
</fmd-deployment-event>
</notification>

<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2017-11-29T12:42:01.432732+00:00</eventTime>
  <ns-deployment-event xmlns="http://www.cisco.com/usp/nfv/usp-uas-common-oper">
    <instance-id>blrnsd1-instance</instance-id>
    <descriptor-id>blrnsd1</descriptor-id>
    <transaction-id>1511959320-398539</transaction-id>
    <operation-type>activate-ns-deployment</operation-type>
    <status>success</status>
  </ns-deployment-event>
</notification>