# VNF Manager North Bound Interface Summary

## Introduction

This technical work paper gives an overview of the requirements for the VNF Manager northbound interface. The northbound interface can be REST, CLI or NETCONF API

## Use Cases

The APIs for VNF Northbound interfaces are for the following functionality

* VM and VNF Management
* Policy and Health Management
* NFVO Grant Related Interfaces

## Priority and Customer Requests

High. Base Functionality for Customer commitments

# Technical Specifications

## VM and VNF Management Northbound APIs

The following APIs are to be supported for VNF and VM management through the VNF Manager. The APIs are REST, CLI and NETCONF based.

* Image Management
  + Upload Image: uploadImage
  + List Images: listImages
  + Delete Image: deleteImage
* Network Service
  + Upload Network Service: upload NSD
  + Instantiate Network Service operation i.e. Deploy Network Service: deployNSD
  + List Network Service: listNetworkService
  + Terminate Network Service: terminateNSD
  + Delete Network Service from Catalog: deleteNSD
* VNF/ VM Management
  + Upload VNF: uploadVNFD
  + Instantiate VNF operation i.e. Deploy VNF: DeployVNFD
  + List VNFs (Query VNF operation): listVNF
  + Terminate VNF operation: terminateVNF
  + Terminate VM (as defined in a single Virtual Descriptor Unit): terminateVM
  + Delete VNF from Catalog: deleteVNFD
  + Configure VNF (Operate VNF operation): configureVNF
  + Manual Scale VNF operation: scaleVNF
  + Modify VNFd (Modify VNF information operation): modifyVNFD
  + Heal VNF operation (Restart VNF as of now): healVNF
* Volume Management
  + Create Volume: createVolume
  + Add Volume to VM: addVolume
  + Detach Volume from VM: detachVolume
  + Delete Volume: deleteVolume
  + Retrieve Volume from NFVI and store in disk format: getVolume
* DNS Service Management
  + Add DNS service to Network Service: addDNS
  + Delete DNS from Network Service: deleteDNS
* Load Balancing Service Management
  + Add Load Balancer to Network Service: addLB
  + Delete Load Balancer from Network Service: deleteLB
* Virtual Network Interface Management
  + Add Connection Point (virtual NIC) to VNF: addConnectionPoint
  + Delete Connection Point (Virtual NIC) from VNF: deleteConnectionPoint
* Snapshot and VM Image Management
  + Create VM Snapshot: vmSnapshot
  + Create VM Image: createVMImage
  + Delete VM Image from Repository: deleteVMImage
* VNF Forwarding Graph Management
  + Upload VNF Forwarding Graph: VNFFGOnboard
  + Deploy VNF Forwarding Graph: VNFFGDeploy
  + Remove Forwarding Rules from Network Service: VNFFGTerminate
  + Delete VNF Forwarding Graph: VNFFGDelete
* Software Deployment on VM
  + Deploy Software on VM: deploySW
  + Remove Software from VM: removes
* Software Management
  + Update VNF Software: updateVNF

## Policy and Health Management Northbound Interfaces

The following APIs are to be supported for VNF and VM Monitoring through the VNF Manager. The APIs are REST, CLI and NETCONF based.

Policy Management Northbound Interfaces are

* Create a Policy: createPolicy
* Update Policy: updatePolicy
* Delete Policy: deletePolicy
* Query Policy: queryPolicy
* Activate Policy: activatePolicy
* De-Activate Policy: deactivatePolicy

Measurement and Monitoring APIs on the Northbound Interface

* Subscribe to Filtered Alarm Set from VNFM : subscribeRequest / subscribeResponse
* Notify Alarms over Northbound interface: notify
* Query active alarms from VNFM: getAlarmListRequest / getAlarmListResponse
* Manually cancel alarm: cancelAlarmRequest / cancelAlarmResponse

## NFVO Interface for Grant Functions

TBD

## Data Model Impact

List any impact on the Descriptor Data Models (NSd, VLd, VDU, VNFd) and VNF Manager configuration

## UI Impact

List any UI Impact

## Dependencies

List any dependencies

## Testing

List Test cases

## Documentation

List the documentation needs

## Schedule

List the APIs and their release schedule.

Nomenclature. Rel-A: July 2016, Rel-B: Sep-2016, Rel-C: Dec-2016, Rel-D: Apr-2017

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Author** | **Date** | **Remarks** |
| **1.0** | Shamik |  | Initial version created |
| **2.0** | Shamik |  | Updated after Anuj’s input |