VNFM Driver API

V0.1

Catalog

1. Scope

The scope of the present document is to describe the VNFM Driver exposed API specification.

2. Terms, Definitions and Abbreviations

For the purposes of the present document, the following abbreviations apply:

Abbreviation	
NFVO	Network Functions Virtualization Orchestrator
VNFM	Virtual Network Function Manager
VNF	Virtual Network Function

Table 2-1 abbreviations

3. Interfaces provided by VNFM Driver

Interfaces use RESTful API and the format is as follows:

http(s)://[hostname][:port]/api/{vnfmtype}/v1/{vnfm_id}/[……]

R1 vnfmtype:

zte-vnfm

fw-vnfm

juju

3.1 Instantiate VNF

If Definition	Description
URI	http(s)://[hostname][:port]/api/{vnfmtype}/v1/{vnfmid}/vnfs
Operation	POST
Direction	NSLCM->VNFMDriver

3.1.1 Request

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceName	M	1	String	Human-readable name of the VNF instance to be created.
vnfPackageId	M	1	String	VNF packageId
vnfDescriptorId	M	1	String	Information sufficient to identify the VNF Descriptor which defines the VNF to be created.
flavourId	M	0..1	String	Reserved
vnfInstanceDescription	M	0..1	String	Human-readable description of the VNF instance to be created.
extVirtualLink	M	0..N	ExtVirtualLinkData	References to external virtual links to connect the VNF to.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the NFVO as input to the instantiation process, specific to the VNF being instantiated.

ExtVirtualLinkData:

Attribute	Qualifier	Cardinality	Content	Description
vlInstanceId	M	0..1	String	Identifier of the VL instance.
vim	CM	0..1	VimInfo	Information about the VIM that manages this resource. This attribute shall be supported and present if VNF-related resource management in direct mode is applicable.
networkId	M	1	String	The network UUID of VIM
cpdId	M	0..1	String	Identifier of the external CPD in VNFD

VimInfo:

Attribute	Qualifier	Cardinality	Content	Description
vimInfoId	M	1	Identifier	The identifier of this VimInfo instance, for the purpose of referencing it from other information elements.
vimId	M	1	Identifier	The identifier of the VIM.
interfaceInfo	M	0..N	KeyValuePair	Information about the interface to the VIM, including VIM provider type, API version, and protocol type.
accessInfo	M	0..N	KeyValuePair	Authentication credentials for accessing the VIM. Examples may include those to support different authentication schemes, e.g., OAuth, Token, etc.
interfaceEndpoint	M	1	String	Information about the interface endpoint. An example is a URL.

interfaceInfo:

Attribute	Qualifier	Cardinality	Content	Description
vimType	M	1	String	vim
apiVersion	M	1	String	
protocolType	M	1	String	http https

accessInfo:

Attribute	Qualifier	Cardinality	Content	Description
tenant	M	1	String	Tenant Name of tenant
username	M	1	String	Username for login
password	M	1	String	Password of login user

```
{
    “vnfInstanceName”:”vFW”,
    “vnfPackageId”:”1”,
    “vnfDescriptorId”:”1”,
    “vnfInstanceDescription”:”vFW_1”,
    “extVirtualLinkLink”:[
        {
            ”vlInstanceId”:”1”,
            “resourceId”:”1246”,
            ” cpdId”:”11111”,
            ”vim”:
            {
                    “vimInfoId”:”1”,
                    “vimid”:”1”,
                “interfaceInfo”:{
                    "vimType":”vim”,
                    "apiVersion":”v2”,
                    "protocolType":”http”
                }
                “accessInfo”:{
                    "tenant":”tenant_vCPE”,
                    "username":”vCPE”,
                    "password":”vCPE_321”
                }
                “interfaceEndpoint”:”http://10.43.21.105:80/”
            }
        }
    ]
    “additionalParam”:{
    ……
    }
}
```

3.1.2 Response

Parameter	Qualifier	Cardinality	Content	Description
jobId	M	1	Identifier	Identifier of the VNF lifecycle operation occurrence. [lifecycleOperationOccurrenceId]
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.

```
{
    “jobId”:”1”,
    “vnfInstanceId”:”1”
}
```

3.2 Terminate VNF

IF Definition	Description
URI	http(s)://[hostname][:port]/api/{vnfmtype}/v1/{vnfmid}/vnfs/{vnfInstanceId}/terminate
Operation	POST
Direction	NSLCM->VNFMDriver

3.2.1 Request

Parameter	Qualifier	Cardinality	Content	Description
terminationType	M	1	Enum	Signals whether forceful or graceful termination is requested. In case of forceful termination, the VNF is shut down immediately, and resources are released. Note that if the VNF is still in service, this may adversely impact network service, and therefore, operator policies apply to determine if forceful termination is allowed in the particular situation. In case of graceful termination, the VNFM first arranges to take the VNF out of service (by means out of scope of the present specification, e.g. involving interaction with EM, if required). Once this was successful, or after a timeout, the VNFM shuts down the VNF and releases the resources.
gracefulTerminationTimeout	M	0..1	TimeDuration	The time interval (second) to wait for the VNF to be taken out of service during graceful termination, before shutting down the VNF and releasing the resources. If not given, it is expected that the VNFM waits for the successful taking out of service of the VNF, no matter how long it takes, before shutting down the VNF and releasing the resources (see note). Minimum timeout or timeout range are specified by the VNF

				Provider (e.g. defined in the VNFD or communicated by other means). Not relevant in case of forceful termination.

```
{
   “vnfInstanceId”:”1”,
   “terminationType”:”graceful”,
   “gracefulTerminationTimeout”:”60”
}
```

3.2.2 Response

Parameter	Qualifier	Cardinality	Content	Description
jobId	M	1	Identifier	Identifier of the VNF lifecycle operation occurrence. [lifecycleOperationOccurrenceId**]**

```
{
   “jobId”:”1”
}
```

3.3 Query VNF

IF Definition	Description
URI	http(s)://[hostname][:port]/api/{vnfmtype}/v1/{vnfmid}/vnfs/{vnfInstanceId}
Operation	GET
Direction	NSLCM->VNFMDriver

3.3.1 Request

VNF filter: vnfInstanceId via url [R1]

3.3.2 Response

Parameter	Qualifier	Cardinality	Content	Description
vnfInfo	M	0..N	VnfInfo	The information items about the selected VNF instance(s) that are returned. If attributeSelector is present, only the attributes listed in attributeSelector will be returned for

				the selected VNF instance(s). See note.

VnfInfo Table

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	String	VNF instance identifier.
vnfInstanceName	M	0..1	String	VNF instance name. See note.
vnfInstanceDescription	M	0..1	String	Human-readable description of the VNF instance.
vnfdId	M	1	String	Identifier of the VNFD on which the VNF instance is based.
vnfPackageId	M	0..1	String	Identifier of the VNF Package used to manage the lifecycle of the VNF instance. See note. Shall be present for an instantiated VNF instance.
version	M	1	String	Version of the VNF.
vnfProvider	M	1	String	Name of the person or company providing the VNF.
vnfType	M	1	String	VNF Application Type
vnfStatus	M	1	Enum	The instantiation state of the VNF. Possible values: INACTIVE (Vnf is terminated or not instantiated), ACTIVE (Vnf is instantiated). [instantiationState]

```
{
  “vnfInfo”:
  {
       "nfInstanceId":”1”,
       "vnfInstanceName":”vFW”,
       "vnfInstanceDescription":”vFW in Nanjing TIC Edge”,
       "vnfdId":”1”,
       "vnfPackageId":”1”,
       "version":”V1.1”,
       "vnfProvider":”ZTE”,
       "vnfType":”vFW”,
       "vnfStatus":” ACTIVE”,
  }
```

}

3.4 Get operation status

IF Definition	Description

URI	http(s)://[hostname][:port]/api/{vnfmtype}
	/v1/{vnfmid}/jobs/{jobid}&responseId={ responseId }

Operation	GET
Direction	NSLCM->VNFMDriver

3.4.1 Request

None

3.4.2 Response

Parameter	Qualifier	Cardinality	Content	Description
jobId	M	1	String	Job ID
responseDescriptor	M	1	-	Including: vnfStatus， statusDescription， errorCode， progress、 responseHistoryList、responseId
status	M	1	String	JOB status started processing finished error
progress	M	1	Integer	progress （1-100）
statusDescription	M	1	String	Progress Description
errorCode	M	1	Integer	Errorcode
responseId	M	1	Integer	Response Identifier
responseHistoryList	M	0..n	ArrayList<>	History Response Messages from the requested responseId to lastest one. Including fields: vnfStatus, statusDescription, errorCode, progress,

				responseId

```
{
    ″jobId″ : ″1234566″,
    ″responseDescriptor″ : {
        ″progress″ : ″40″,
        ″status″ : ″proccessing″,
        ″statusDescription″ : ″OMC VMs are decommissioned in VIM″,
        ″errorCode″ : null,
        ″responseId″ : ″42″,
        ″responseHistoryList″ : [{
                ″progress″ : ″40″,
                ″status″ : ″proccessing″,
                ″statusDescription″ : ″OMC VMs are decommissioned in VIM″,
                ″errorCode″ : null,
                ″responseId″ : ″1″
            }, {
                ″progress″ : ″41″,
                ″status″ : ″proccessing″,
                ″statusDescription″ : ″OMC VMs are decommissioned in VIM″,
                ″errorCode″ : null,
                ″responseId″ : ″2″
            }
        ]
    }
}
```

3.5 Scale VNF

If Definition	Description
URI	http(s)://[hostname][:port]/api/{vnfmtype}/v1/{vnfmid}/vnfs/{vnfInstanceId}/scale
Operation	POST
Direction	NSLCM->VNFMDriver

3.5.1 Request

Parameter	Qualifier	Cardinality	Content	Description
type	M	1	Enum	Defines the type of the scale operation requested (scale out, scale in). The set of types actually supported depends on the capabilities of the VNF being managed, as declared in the VNFD. See note 1.

aspectId	M	1	Identifier	Identifies the aspect of the VNF that is requested to be scaled, as declared in the
numberOfSteps	M	0..1	Integer	Number of scaling steps to be executed as part of this ScaleVnf operation. It shall be a positive number. Defaults to 1. The VNF Provider defines in the VNFD whether or not a particular VNF supports performing more than one step at a time. Such a property in the VNFD applies for all
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the NFVO as input to the scaling process, specific to the VNF being scaled. **Reserved**
NOTE 1: ETSI GS NFV-IFA 010 [2] specifies that the lifecycle management operations that expand or contract a VNF instance include scale in, scale out, scale up and scale down. Vertical scaling (scale up, scale down) is not supported in the present document. SCALE_IN designates scaling in. SCALE_OUT 1 designates scaling out. NOTE 2: A scaling step is the smallest unit by which a VNF can be scaled w.r.t a particular scaling aspect.				

```
{
   “vnfInstanceId”:”5”,
   “type”:” SCALE_OUT”,
   “aspectId”:”101”,
   “numberOfSteps”:”1”,
   “additionalParam”:{
   ……
   }
}
```

3.5.2 Response

Parameter	Qualifier	Cardinality	Content	Description
jobId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

```
{
   “jobId”:”1”
}
```

3.6 Heal VNF

If Definition	Description
URI	http(s)://[hostname][:port]/api/{vnfmtype}/v1/{vnfmid}/vnfs/{vnfInstanceId}/heal

Operation	POST
Direction	NSLCM->VNFMDriver

3.6.1Request

Parameter	Qualifier	Cardinality	Content	Description
action	M	1	String	Indicates the action to be done upon the given virtual machine. Only ‘vmReset’ is supported currently.
affectedvm	M	1	AffectedVm	Defines the information of affected virtual machines.
~~isgrace~~	~~M~~	~~1~~	~~String~~	~~Defines whether this operation is forced or done gracefully. Valid value includes:~~ ~~'force'~~

AffectedVm

Parameter	Qualifier	Cardinality	Content	Description
vimid	M	1	String	Defines the UUID of virtual machine.
vduid	M	1	String	Defines the id of vdu.
vmname	M	1	String	Defines the name of virtual machine..

```
{
  “action”: “vmReset”,
  “affectedvm”:
    {
      “vmid”: “804cca71-9ae9-4511-8e30-d1387718caff”,
      “vduid”: “vdu_100”,
      “vmname”: “ZTE_SSS_111_PP_2_L”
    },
}
```

3.6.2Response

Parameter	Qualifier	Cardinality	Content	Description
jobId	M	1	Identifier	The identifier of the VNF healing operation occurrence.

```
{
    “jobId”:”1”
}
```