



# HPE NFV Director

On-Boarding Guide Operations:     Scale In of a VNF

Release 4.1

Second Edition

# Notices

---

## Legal notice

© Copyright 2016 Hewlett Packard Enterprise Development LP

Confidential computer software. Valid license from HPE required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for HPE products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HPE shall not be liable for technical or editorial errors or omissions contained herein.

Printed in the US

## Trademarks

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries. Oracle and Java are registered trademarks of Oracle and/or its affiliates.

Adobe®, Acrobat® and PostScript® are trademarks of Adobe Systems Incorporated.

Microsoft®, Internet Explorer, Windows®, Windows Server 2007®, Windows XP®, and Windows 7® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Firefox® is a registered trademark of the Mozilla Foundation.

Google Chrome® is a trademark of Google Inc.

EnterpriseDB® is a registered trademark of EnterpriseDB.

Postgres Plus® Advanced Server is a registered U.S. trademark of EnterpriseDB.

UNIX® is a registered trademark of The Open Group.

X/Open® is a registered trademark, and the X device is a trademark of X/Open Company Ltd. in the UK and other countries.

Red Hat® is a registered trademark of the Red Hat Company.

Apache CouchDB, CouchDB, and the project logo are trademarks of The Apache Software Foundation.

Node.js project. Joyent® and Joyent's logo are registered trademarks of Joyent, Inc.

Neo4j is a trademark of Neo Technology.

# Contents

Notices .....	2
Preface .....	6
About this guide.....	6
Audience .....	6
Document history.....	6
Chapter 1 Scale In of a VNF. ....	7
Chapter 2 Specific Elements of the TLD Scale In VNF.....	9
2.1 TLD QUOTA ASSIGNMENT: Quota Assignment Task. ....	9
2.2 SCALE IN PRE TLD: SCALE IN PRE. ....	10
2.3 Undeploy VNF: STOP_MONITOR .....	11
2.4 Deactivate VM : Deactivate_VM .....	12
2.5 SCALE IN POST TLD: SCALE IN POST .....	13
2.6 Delete vPort: DEACTIVATE_PORT_GROUP .....	14
2.1 Delete vPort: Delete_vPort .....	15
2.1 Undeploy Monitor: UNDEPLOY_MONITOR .....	16
2.2 Undeploy Monitor: UNDEPLOY_MONITOR .....	17
2.1 . Delete Inventory: DELETE_INVENTORY_COMPONENT_LEVEL.....	18
2.2 . TLD VNF STATUS CHANGE: VNF_STATUS_CHANGE. ....	19

# List of tables

Table 1: Document history.....	6
--------------------------------	---

# List of figures

Figure: 1: Selection of the components for the Scale In. ....	9
Figure: 2 Deployment of Pre-Processing policies. ....	10
Figure: 3: Stopping monitor. ....	11
Figure: 4 Deactivation of a VM. ....	12
Figure: 5: Deploying Post-Processing policies. ....	13
Figure: 6: Deactivation of the Port Group associated. ....	14
Figure: 7: Deletion of a vPort. ....	15
Figure: 8: Deletion of the volume. ....	16
Figure: 8: Undeployment of a monitor. ....	17
Figure: 10: Deletion at component level. ....	18
Figure: 11: Delete Virtual Machine from scaled out VNF. ....	19

# Preface

---

## About this guide

---

This Guide is intended to explain and guide the user through the Scale In of a VNF.

## Audience

---

This document is targeting any user level of NFV Director: Domain users, Organization Users, Tenant Users, Group Users and Datacenter users.

## Document history

---

Table 1: Document history

Edition	Date	Description
1.0	1 September 2016	First Edition

# Chapter 1 Scale In of a VNF.

From now on, and to make easier the understanding of the TLDs, we are going to explain the functionality of each set of TASK\_LIST\_DEFINITION:GENERIC, and the number of TASK\_DEFINITION:GENERIC children of the previously mentioned TASK\_LIST\_DEFINITION:GENERIC.

Basically, the TASK\_LIST\_DEFINITION:GENERIC connect what we can consider “units of execution”, those are the TASK\_DEFINITION:GENERIC, that have a WORKFLOW assigned to be executed when the execution of the TLD reach them.

If you like to have a more deep knowledge about the workflows mentioned through this document please refer to the specific document.



If in the category FIND, the attribute Path is present, the attribute FIND.ArtifactType will be the starting artifact for the Path, but the FIND.Status attribute refers to the last artifact on the Path.

**FIND.ArtifactType ==** **VIRTUAL\_MACHINE.**  
**FIND.Status==** **INSTANTIATED.**  
**FIND.Path==**

**VIRTUAL\_MACHINE>VIRTUAL\_CORE<CORE<CPU<SERVER<AVAILABILITY\_ZONE<REGION**  
**>COMPUTE>FLAVOR**

**In this example, we are looking for a FLAVOR in status INSTANTIATED, we do not expect to get a VIRTUAL\_MACHINE, in status INSTANTIATED.**

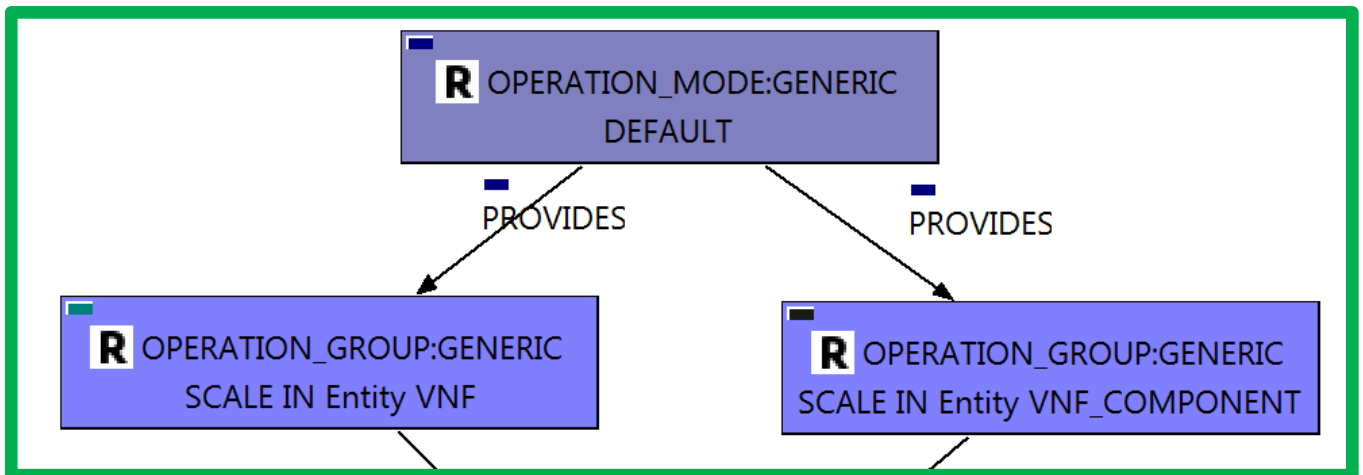


If during the use of the TLDs, the “Regenerate UUIDs” option is used, the user should check the Id of the tree that brings all the elements of the TLD, this “id” is specific and it will be the same for all the tree groups in all the TLDs.



The two modes available are “Default” and “Simulated”, the second one is only available if it is configured previously, by defect, and the mode that will be used is “Default”.

The Scale In TLD has some specifications that should be explained, when we open the TLD of the operation, the first we will see is the following:



This TLD starts with one Operation\_Mode (as usual), but it has two Operation\_Group, each one of the elements is “RootArtifact” of its own tree, this division of groups is justified in order to cover all the possible levels of the escalation policy, one group responds to the policies that has been set between the VNF and the VNF Component, and the second one respond to the policies set it in the level between the VNF Component and the Virtual Machine.



## Chapter 2 Specific Elements of the TLD Scale In VNF.

In this chapter the different elements of the specific TLD will be explained conscientiously.

### 2.1 TLD QUOTA ASSIGNMENT: Quota Assignment Task.

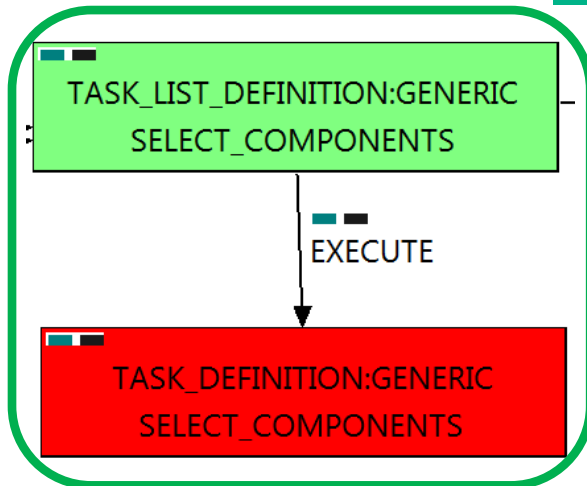


Figure: 1: Selection of the components for the Scale In.

The Scale In over a VNF is a process that delete a number of Virtual Machines that were escalated previously over a specific VNF, this means that a Scale In is only applicable to a VNF that suffered a Scale Out before.

The WF present in this TD will check for all the artifacts and relationship created for the Virtual Machine that was scaled out also will manage to leave the components that are connected with the ones that are going to be deleted in an adequate state to be used.

Targets of the TASK DEFINITION:  
ENABLED

STATUS of the TD:

GENERAL.Name ==	Select Components
FIND.MainArtifact ==	
<b>VNF@status=ACTIVE,VNF_COMPONENT&lt;VNF@status=ACTIVE</b>	
SET.Running_Status ==	ACTIVE.
SET.Status ==	ACTIVE.
EXECUTE.Workflow ==	
	<b>"WF_TS_SCALE_IN_SELECTED"</b>
EXECUTE.Inactive==	false
ROLLBACK.Behaviour_on_error ==	STOP
ROLLBACK.Rollback_Status ==	ACTIVE.
ROLLBACK.Number_of_retries ==	0
DATA.Lock ==	false

The Workflow present in EXECUTE.Workflow it is going to seek a VNF in Status ACTIVE in the DDBB, when the WF find it, it will start. This workflow assign all the resources needed by the VNF to get a successful Deploy, it will check the available resources and decide which one should be assigned.

In case of error during the execution, the workflow jump to the ROLLBACK category, if the "Behaviour\_on\_error" attribute its set on "ROLLBACK" the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, the attribute "number\_of\_retries" set the number of rollback attempts. In this case, the behavior is set as "STOP" also the TLD has not have a rollback workflow assigned, in case of error the operation will stop without rollback.

Due to that the value of the attribute DATA.Lock is false, once the TD has finished, no artifact or element will be set as "locked".

## 2.2 SCALE IN PRE TLD: SCALE IN PRE.

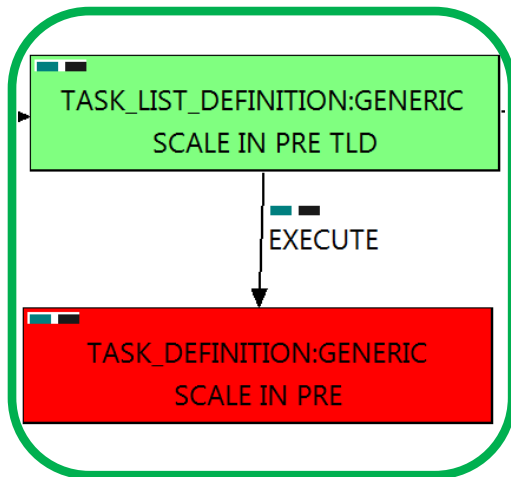


Figure: 2 Deployment of Pre-Processing policies.

This TD is responsible of the provision in the right order of the artifacts referenced by the PreProcessing policies, these policies allow the user to set a number of elements that should be taken in consideration in a certain order, in other case, the execution will fail depending on the event occurred.

Targets of the TASK DEFINITION:  
ENABLED

STATUS of the TD:

FIND.MainArtifact ==	POLICY:POSTPRE_PROCESSING
FIND.Condition ==	
<b>PROCESSING_JOB.Job_type==constant:PRE&amp;&amp;</b>	
<b>PROCESSING_JOB.Operation==constant:SCALEIN</b>	
EXECUTE.OrderBy ==	PROCESSING_JOB.OrderBy
ROLLBACK.Number_of_retries ==	0
DATA.Lock ==	false

In this TD there is not a workflow to be executed, the target of this TD is process in the correct order the PreProcessing policies present in the VNF, these policies should be executed in a specific order to make the changes or configurations properly, in other case an error will be launched

If the TD ends successfully, the Pre-Processing policies will have been processed adequately.

In case of error during the execution, the TD will jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK.

Due to that the value of the attribute DATA.Lock is false, when the Task Definition has finished the artifact that was used in the workflow executed will remain unlocked.

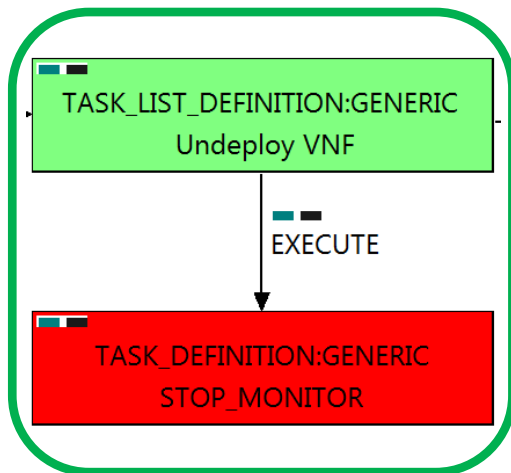


Figure: 3: Stopping monitor.

### 2.3 Undeploy VNF: STOP\_MONITOR

The TDs that have present in their names “Stop”, are Task Definitions responsible of the stopping of a specific artifact or element, in this case of the MONITORs, the TLD it is going to stop the MONITOR element given.

Once finished, our VNF should have the MONITOR given in status DEPLOYED, having taken in consideration all the rules for the stopping.

Targets of the TASK DEFINITION:  
ENABLED

STATUS of the TD:

GENERAL.Name ==	Stop Monitor
FIND.MainArtifact==	<b>MONITOR</b>
FIND.Condition==	status==constant:TO_BE_STOPPED
SET.Running_Status ==	TO_BE_STOPPED
Set.Status ==	TO_BE_UNDEPLOYED
EXECUTE.Workflow ==	<b>“WF_TS_MONITOR_STOP”</b>
EXECUTE.Inactive==	false
ROLLBACK.Behaviour_on_error ==	STOP
ROLLBACK.Number_of_retries ==	0
DATA.Lock ==	false

The Workflow present in EXECUTE.Workflow it is going to seek a MONITOR in Status STARTED in the DDBB, when the WF find it, it will start. This workflow stop the given MONITOR needed by the VNF to get a successful Undeploy.

In case of error during the execution, the TD will jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, but in this case, we have a “STOP” as value set for behavior, so no Rollback it is going to be initiated, the execution will stop.

Due to that the value of the attribute DATA.Lock is false, when the Task Definition has finished the artifact that was used in the workflow executed will remain unlocked.

## 2.4 Deactivate VM : Deactivate\_VM

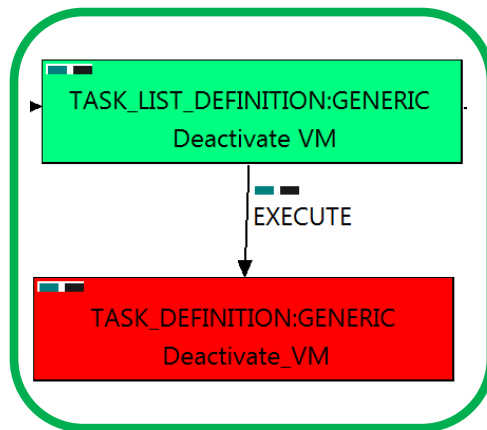


Figure: 4 Deactivation of a VM.

The TDs that have present in the their names “Deactivate”, are Task Definitions responsible of the deactivation in the platform targeted and the updating of the status in the platform and the DDBB, in this case, the artifact that is going to be deactivated is a “VIRTUAL\_MACHINE with status “STOPPED”, this means, when this workflow finish, we will have the VIRTUAL\_MACHINE given with a new status INSTANTIATED.

Targets of the TASK DEFINITION:  
STATUS of the TD: ENABLED  
Categories:

```

FIND.MainArtifact ==
VNF>VNF_COMPONENT>
VIRTUAL_MACHINE@status=TO_BE_DEACTIVATED,
VNF_COMPONENT>
VIRTUAL_MACHINE@status=TO_BE_DEACTIVATED
SET.Running_Status == TO_BE_DEACTIVATED.
SET.Status == TO_BE_DELETED.
SET.OperationalStatus == TO_BE_DELETED
EXECUTE.Workflow == “WF_TS_DEACTIVATE_VM”
EXECUTE.Inactive== false
ROLLBACK.Behaviour_on_error == STOP
ROLLBACK.Number_of_retries == 0
DATA.Lock == false

```

The Workflow present in EXECUTE.Workflow attribute it is going to deactivate a “VIRTUAL\_MACHINE” in Status TO\_BE\_DELETED in the DDBB . Once found , the WF will start the deactivation, if the deactivation it is successful we set the status of the artifact as the SET.Status attribute dictates. The attribute SET.Running\_Status concern about the temporal status that the artifact it is going to maintain until the final change of status that comes from SET.Status.

This TD could launch different workflows depending on the type of the VM that it is going to be deactivated, the main kinds of our VIRTUAL\_MACHINES are HELION, and thereupon two of the WFs that could be used in this deactivation are: “WF\_NFVD\_DEACTIVATE\_VM\_HELION” and “WF\_NFVD\_DEACTIVATE\_VM\_OPENSTACK”.

In case of error during the execution, the workflow jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, in this case it will be “WF\_TS\_ACTIVATE\_VM”, but in this case, we have a “STOP” set as behavior, so no Rollback it is going to be initiated, so the execution it is going to end here in case of error.

Due to that the value of the attribute DATA.Lock is false, when the Task Definition has finished the artifact that was used in the workflow executed will remain unlocked.

## 2.5 SCALE IN POST TLD: SCALE IN POST

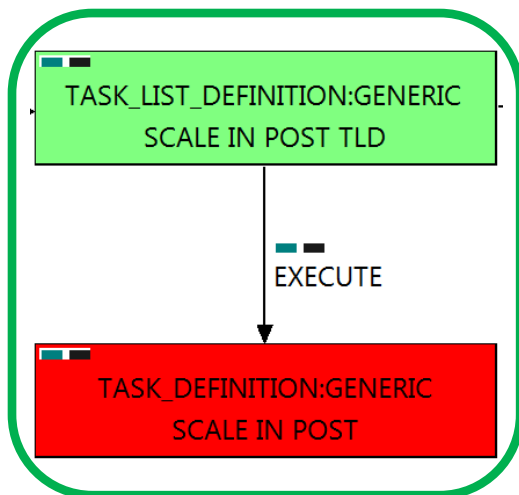


Figure: 5: Deploying Post-Processing policies.

This TD is responsible of the provision in the right order of the artifacts referenced by the Post-Processing policies, these policies allow the user to treat a number of elements that should be taken in consideration after the execution of some TD in a specific order, in other case, the execution will fail depending on the event occurred.

Targets of the TASK DEFINITION:  
ENABLED

STATUS of the TD:

FIND.MainArtifact ==	POLICY:POSTPRE_PROCESSING
FIND.Condition ==	
<b>PROCESSING_JOB.Job_type==constant:POST&amp;&amp;</b>	
<b>PROCESSING_JOB.Operation==constant:SCALEIN</b>	
EXECUTE.OrderBy ==	PROCESSING_JOB.OrderBy
EXECUTE.Inactive==	false
ROLLBACK.Behaviour_on_error ==	STOP
ROLLBACK.Number_of_retries ==	0
DATA.Lock ==	false

In this TD there is not a workflow to be executed, the target of this TD is process in the correct order the PostProcessing policies present in the VNF, these policies should be executed in a specific order to make the changes or configurations properly, in other case an error will be launched

If the TD ends successfully, the Post-Processing policies will have been applied adequately.

In case of error during the execution, the TD will jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, but in this case, we have a “STOP” as value set for behavior, so no Rollback it is going to be initiated, the execution will stop.

Due to that the value of the attribute DATA.Lock is false, when the Task Definition has finished the artifact that was used in the workflow executed will remain unlocked.

## 2.6 Delete vPort: DEACTIVATE\_PORT\_GROUP

The TDs that have present in the their names “Deactivate”, are Task Definitions responsible of the deactivation in the platform targeted and the updating of the status in the platform and the DDBB, in this case, the artifact that is going to be deactivated is a “PORT\_GROUP:VCENTER”, this means, when this workflow finish, we will have the artifact PORT\_GROUP with status INSTANTIATED.

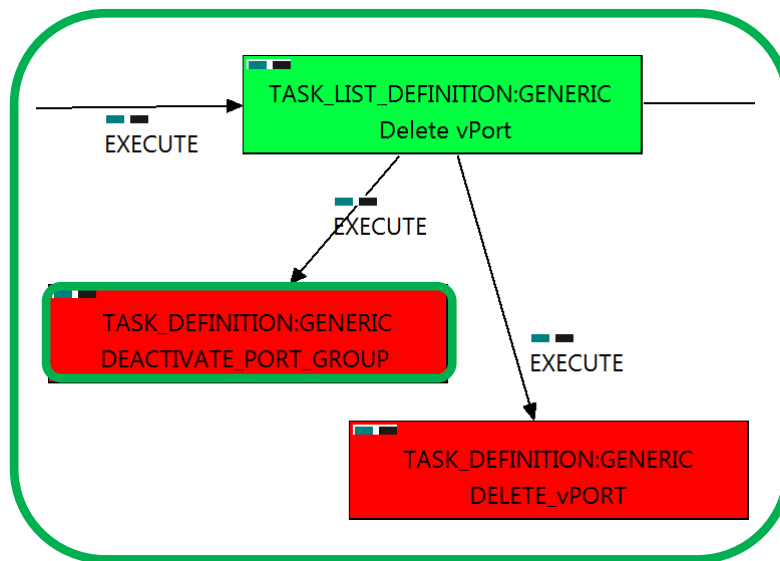


Figure: 6: Deactivation of the Port Group associated.

Targets of the TASK:DEFINITION: STATUS of the TD: ENABLED  
Categories:

```

FIND.MainArtifact==
VNF>VNF_COMPONENT>VIRTUAL_MACHINE>VIRTUAL_PORT<PORT_GROUP,
VNF_COMPONENT>VIRTUAL_MACHINE>VIRTUAL_PORT<PORT_GROUP
EXECUTE.Workflow ==          "WF_TS_DEACTIVATE_PORT_GROUP_VCENTER"
EXECUTE.Inactive==           false
ROLLBACK.Behaviour_on_error == STOP
ROLLBACK.Number_of_retries == 0
DATA.Lock ==                  true
  
```

In case of error during the execution, the TD will jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, but in this case, we have a “STOP” as value set for behavior, so no Rollback it is going to be initiated, the execution will stop.

Due to that the value of the attribute DATA.Lock is true, when the Task Definition has finished the artifact that was used in the workflow executed will be set as “Locked”.

## 2.1 Delete vPort: Delete\_vPort

The TDs that have present in their names “Delete”, are Task Definitions that delete an artifact or element from the DDBB or from the platforms targeted, in this case, the artifact that it is going to be deleted is the VIRTUAL\_PORT given.

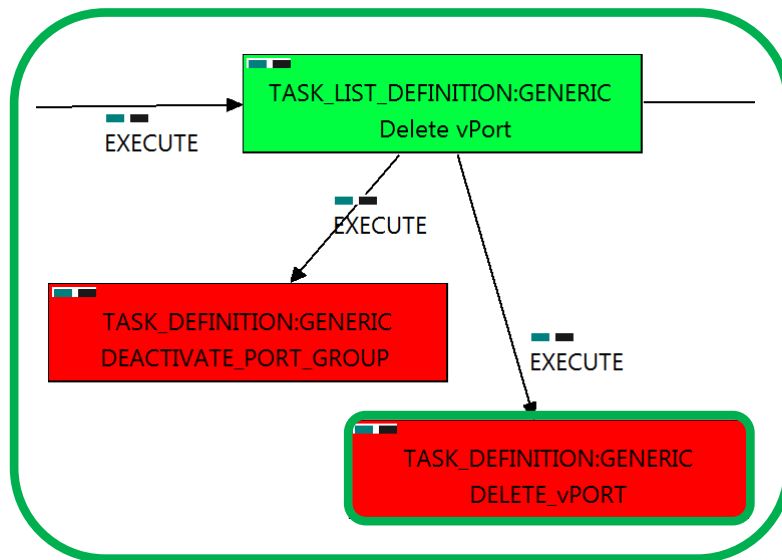


Figure: 7: Deletion of a vPort.

Targets of the TASK:DEFINITION: STATUS of the TD: ENABLED  
Categories:

```

FIND.MainArtifact==
VNF>VNF_COMPONENT>VIRTUAL_MACHINE>VIRTUAL_PORT@status=TO_BE_DELETED,V
NF_COMPONENT>VIRTUAL_MACHINE>VIRTUAL_PORT@status=TO_BE_DELETED
EXECUTE.Workflow == "WF_TS_DEACTIVATE_PORT"
EXECUTE.Inactive== false
ROLLBACK.Behaviour_on_error == STOP
ROLLBACK.Number_of_retries == 0
DATA.Lock == true
  
```

The Workflow present in EXECUTE.Workflow it is going to seek a VIRTUAL\_PORT in Status ACTIVE in the DDBB, when the WF find it, it will start. This workflow will start one more, this last one, it is going to be named after the VIRTUAL\_PORT that the TD it is trying to delete.

In case of error during the execution, the workflow jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, but in this case, we have a “STOP” set as behavior, so no Rollback it is going to be initiated, so the execution it is going to end here in case of error.

Due to that the value of the attribute DATA.Lock is true, when the Task Definition has finished the artifact that was used in the workflow executed will be set as “Locked”.

## 2.1 Undeploy Monitor: UNDEPLOY\_MONITOR

The TDs that have present in the their names “Delete”, are Task Definitions responsible of the deletion of an artifact in the platform targeted and the updating of the status in the platform and the DDBB, in this case, the artifact that is going to be deleted is a “VOLUME”, this means, when this workflow finish, we are going to have a volume less.

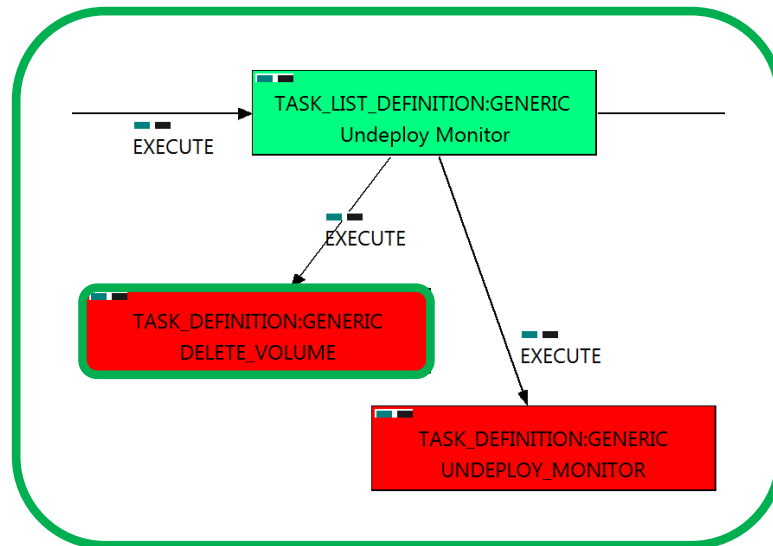


Figure: 8: Deletion of the volume.

Targets of the TASK:DEFINITION:  
Categories:

STATUS of the TD: ENABLED

GENERAL.Name==	Delete Volume
FIND.mainArtifact ==	
VNF>VNF_COMPONENT>VIRTUAL_MACHINE>VIRTUAL_LUN@status=TO_BE_DELETED,	
VNF_COMPONENT>VIRTUAL_MACHINE>VIRTUAL_LUN@status=TO_BE_DELETED.	
SET.Running_Status ==	TO_BE_DELETED.
SET.Status ==	TO_BE_DELETED
EXECUTE.Workflow ==	“WF_TS_DELETE_VOLUME”
ROLLBACK.Behaviour_on_error ==	STOP
ROLLBACK.Number_of_retries ==	0
DATA.Lock ==	false

The Workflow present in EXECUTE.Workflow attribute it is going to seek a VIRTUAL\_LUN in the DDBB . Once found , the WF will start the deleting, if the deletion it is successful we set the status of the artifact as the SET.Status attribute dictates. The attribute SET.Running\_Status concern about the temporal status that the artifact it is going to maintain until the final change of status that comes from SET.Status.

In case of error during the execution, the workflow jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, in this case it will be “WF\_TS\_CREATE\_VOLUME”. But in this case the behavior is set as CONTINUE, this means, that the execution is going to continue no matter which error could be.



The attribute DATA.Lock is set with a value of “false”, this means no element will be locked at the of the TD’s execution.

## 2.2 Undeploy Monitor: UNDEPLOY\_MONITOR

The TDs that have present in the their names “Undeploy” are Task Definitions responsible of the undeployment in the platform targeted and the updating of the status in the platform and the DDBB , these deployments are slightly different to the ones we launch for our entities, as a rule, they are small components as the MONITORS. In this case, the artifact that is going to be deployed is a “MONITOR”, this means, when this workflow finish, we will have a MONITOR deployed with status DEPLOYED.

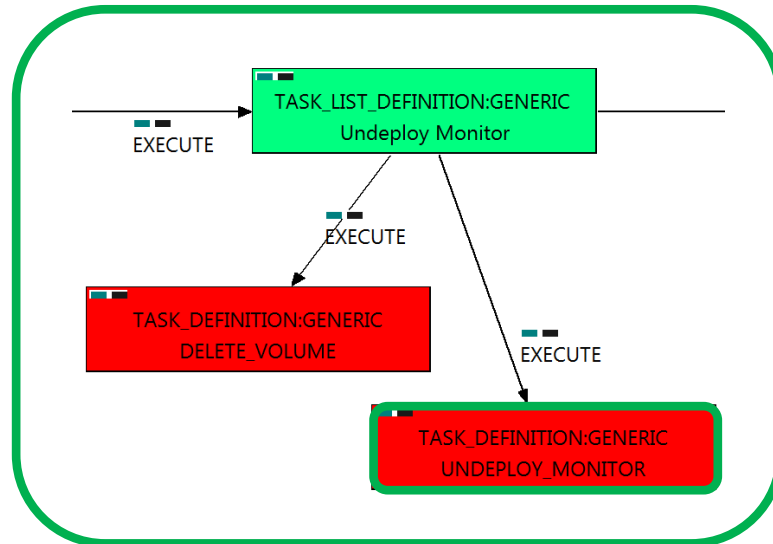


Figure: 9: Undeployment of a monitor.

Targets of the TASK:DEFINITION:  
Categories:

STATUS of the TD: ENABLED

GENERAL.Name==	UNDEPLOY MONITOR
FIND.ArtifactType ==	MONITOR.
FIND.Condition ==	<b>status==constant:TO_BE_UNDEPLOYED</b>
SET.Running_Status ==	TO_BE_UNDEPLOYED.
SET.Status ==	TO_BE_DELETED
EXECUTE.Workflow ==	<b>“WF_TS_MONITOR_UNDEPLOY”</b>
ROLLBACK.Behaviour_on_error ==	STOP
ROLLBACK.Number_of_retries ==	0
DATA.Lock ==	false

The Workflow present in EXECUTE.Workflow attribute it is going to seek a MONITOR in Status DEPLOYED in the DDBB . Once found, the WF will start the deployment, if the deployment is successful we set the status of the artifact as the SET.Status attribute dictates. The attribute SET.Running\_Status concern about the temporal status that the artifact it is going to maintain until the final change of status that comes from SET.Status.

In case of error during the execution, the workflow jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, but in this case, we have a “STOP” set as behavior, so no Rollback it is going to be initiated, so the execution it is going to end here in case of error.

Due to that the value of the attribute DATA.Lock is false, when the Task Definition has finished the artifact that was used in the workflow executed will remain unlocked.

## 2.1 . Delete Inventory: DELETE\_INVENTORY\_COMPONENT\_LEVEL.

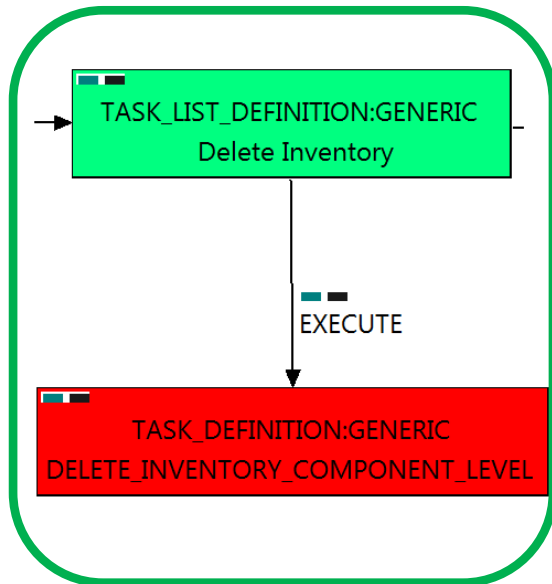


Figure: 10: Deletion at component level.

The TDs that have present in their names “Status Change”, are Task Definitions responsible of the change in the status of the entity associated, in this case responsible of the deletion of a specific entity by ID, these entities are the components related to the Virtual Machine that is going to be deleted during the Scale In process of the VNF. When the WF has finished we will have an VNF with status ACTIVE in case of successful execution, or status ERROR in case of error, or simply not any change in the status because a ROLLBACK during the execution.

Targets of the TASK DEFINITION:  
ENABLED

STATUS of the TD:

GENERAL.Name ==	DELETE VNF
FIND.MainArtifact ==	
<b>VNF&gt;POLICY:ENTITY_RANGE&gt;</b>	
<b>VNF_COMPONENT@status=TO_BE_DELETED</b>	
EXECUTE.Inactive==	false
EXECUTE.Workflow ==	
“WF_TS_DELETE_INSTANCE_TREE_BY_ARTIFACT_ID”	
ROLLBACK.Behaviour_on_error ==	STOP
ROLLBACK.Number_of_retries ==	0
DATA.Lock ==	false

The TASK\_DEFINITION do not execute any workflow, with the attributes present in the categories it is enough to change the status of the entity.

In case of error during the execution, the workflow jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, but in this case, we have a “STOP” set as behavior, so no Rollback it is going to be initiated, so the execution it is going to end here in case of error.

The TLD will finish correctly once the execution reaches this point, the VNF will change its status to “ACTIVE”.

## 2.2 . TLD VNF STATUS CHANGE: VNF\_STATUS\_CHANGE.

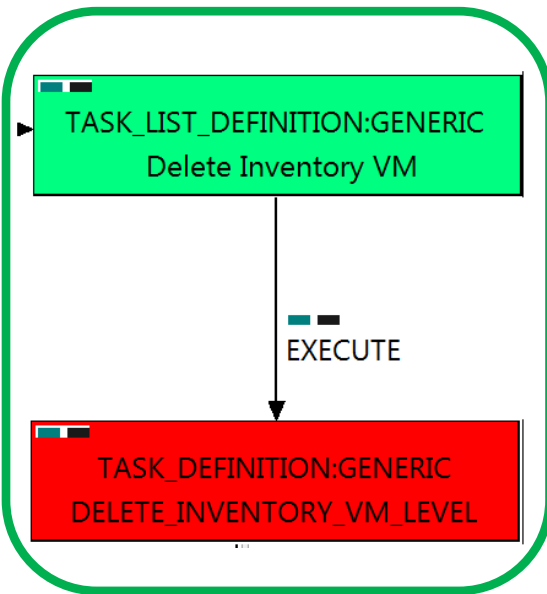


Figure: 11: Delete Virtual Machine from scaled out VNF.

The TDs that have present in their names “Status Change”, are Task Definitions responsible of the change in the status of the entity associated, in this case the Virtual Machine related to the VNF that is execution the Scale In. When the WF has finished we will have an VNF with status ACTIVE in case of successful execution, or status ERROR in case of error, or simply not any change in the status because a ROLLBACK during the execution.

Targets of the TASK DEFINITION:  
ENABLED

STATUS of the TD:

```

GENERAL.Name ==                                DELETE_INVENTORY_VM
FIND.MainArtifact ==
VNF>VNF_COMPONENT>POLICY:ENTITY_RANGE>
VIRTUAL_MACHINE@status=TO_BE_DELETED,
VNF_COMPONENT>POLICY:ENTITY_RANGE>
VIRTUAL_MACHINE@status=TO_BE_DELETED
FIND.Condition ==                                status==constant:INSTANTIATED
SET.Running_Status ==                                INSTANTIATED.
SET.Status ==                                ACTIVE.
EXECUTE.Workflow==
    “WF_TS_DELETE_INSTANCE_TREE_BY_ARTIFACT_ID”
EXECUTE.Inactive==                                false
ROLLBACK.Behaviour_on_error ==                                STOP
ROLLBACK.Number_of_retries ==                                0
DATA.Lock ==                                true
  
```

The TASK\_DEFINITION do not execute any workflow, with the attributes present in the categories it is enough to change the status of the entity.

In case of error during the execution, the workflow jump to the ROLLBACK category, If the “Behaviour\_on\_error” attribute its set on “ROLLBACK” the WF will start the execution of the Workflow present in the attribute with the same name in the category ROLLBACK, but in this case, we have a “STOP” set as behavior, so no Rollback it is going to be initiated, so the execution it is going to end here in case of error.

The TLD will finish correctly once the execution reaches this point, the VNF will change its status to “ACTIVE”.