

# Dell EMC VMAX Storage

## 1. VMAX Connection Management

### 1.1. VMAX Connection Management

#### 1.1.1. Add Unisphere Server instance

Register a Unisphere for VMAX server instance

#### 1.1.1.1. Versions

Version	Date	User	Comment
0.0.1	Jun 27, 2017	admin	
1.0.0	Jan 23, 2018	admin	

Current version:1.0.0

#### 1.1.1.2. Inputs

Name	Type	Description
connName*	string	Name of the connection
server*	string	FQDN or IP address of the Unisphere server instance
port*	string	HTTPS Port number of the Unisphere server instance
ignoreCertificateWarnings	boolean	If set to Yes, the Unisphere Server instance certificate is accepted silently and the certificate is added to the trusted store
username*	string	User name of the Unisphere server instance
password*	SecureString	Password for the Unisphere server instance

\*Parameter is mandatory

#### 1.1.1.3. Outputs

Name	Type	Description
newConnection	VMAXPlugin:Connection	New Unisphere connection Instance

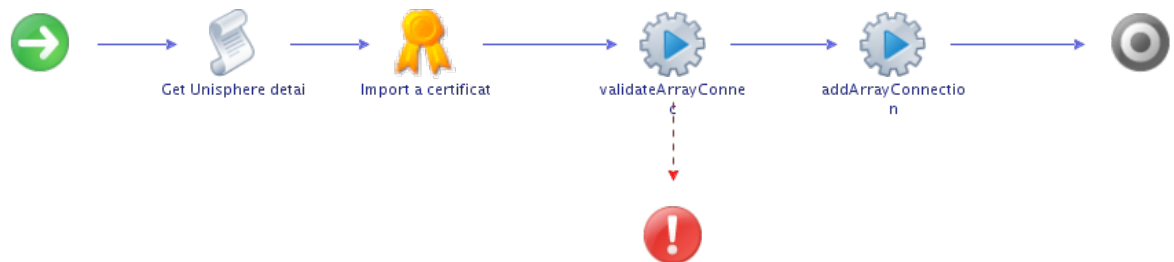
#### 1.1.1.4. Attributes

Name	Type	Description
errorCode	string	
hostAndPort	string	

#### 1.1.1.5. Parameter presentation

Parameter	Description	Parameter Info
connName	Name of the connection	Mandatory input: true
server	FQDN or IP address of the Unisphere server instance	Mandatory input: true
port	HTTPS Port number of the Unisphere server	Default value: 8443 Mandatory input: true
ignoreCertificateWarnings	Do you want to ignore certificate warnings? If you select Yes, the Unisphere Server instance certificate is accepted silently and the certificate is added to the trusted store	
username	User name of the Unisphere server instance	Mandatory input: true
password	Password for the Unisphere server instance	Mandatory input: true

### 1.1.1.6. Workflow schema



### 1.1.1.7. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
Get Unisphere details	WorkflowItemTask	Import a certificate from URL	
Import a certificate from URL	WorkflowItemLink	validateArrayConnection	Imports an SSL certificate into the server trust store.
addArrayConnection	WorkflowItemTask		This action adds a new U4V connection to inventory
validateArrayConnection	WorkflowItemTask	addArrayConnection	
	WorkflowItemEnd		

### 1.1.1.8. Source code for the used actions

#### 1.1.1.8.1. Get Unisphere details

```

var u = new URL();
hostAndPort = u.escapeHost(server) + ":" + port;

```

#### 1.1.1.8.2. addArrayConnection

```

//Auto generated script, cannot be modified !
actionResult =

```

## List the Unisphere Server Instances

---

```
System.getModule("com.dell.storage.vmax.configuration").addArrayConnection(connName,server,
,port,username,password) ;
```

### 1.1.1.8.3. validateArrayConnection

```
//Auto generated script, cannot be modified !
```

```
actionResult =
```

```
System.getModule("com.dell.storage.vmax.configuration").validateArrayConnection(server,por
t,user,password) ;
```

## 1.1.2. List the Unisphere Server Instances

List all known Unisphere for VMAX instances

### 1.1.2.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.1.2.2. Outputs

Name	Type	Description
connList	Array/VMAXPlugin:Co nnection	

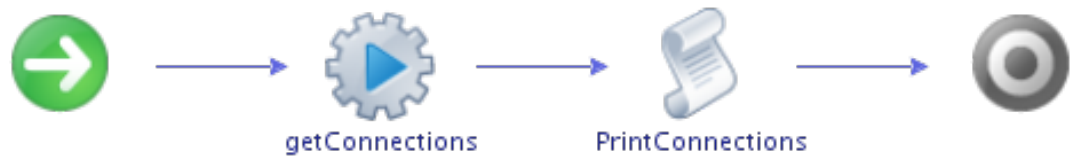
### 1.1.2.3. Attributes

Name	Type	Description
actionResult	Array/VMAXPlugin:Co nnection	

### 1.1.2.4. Parameter presentation

Parameter	Description	Parameter Info
-----------	-------------	----------------

### 1.1.2.5. Workflow schema



### 1.1.2.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
getConnections	WorkflowItemTask	PrintConnections	
PrintConnections	WorkflowItemTask		

### 1.1.2.7. Source code for the used actions

#### 1.1.2.7.1. getConnections

```
//Auto generated script, cannot be modified !  
actionResult = System.getModule("com.dell.storage.vmax.configuration").getConnections() ;
```

#### 1.1.2.7.2. PrintConnections

```
var connList = actionResult;  
  
if ( !connList )  
{  
    throw("There are no Unisphere server instances registered");  
}  
  
for each (var conn in connList)  
{  
    System.log(conn.displayName);  
}
```

## 1.1.3. Remove Unisphere Server instance

Unregister a Unisphere for VMAX from the vRO server

### 1.1.3.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.1.3.2. Inputs

Name	Type	Description
connection*	VMAXPlugin:Connection	Select the connection to remove

\*Parameter is mandatory

### 1.1.3.3. Parameter presentation

Parameter	Description	Parameter Info
connection	Select the connection to remove	Mandatory input: true

### 1.1.3.4. Workflow schema



### 1.1.3.5. Workflow items

Name	Type	Next item	Description
------	------	-----------	-------------

Name	Type	Next item	Description
	WorkflowItemEnd		
removeArrayConnection	WorkflowItemTask		

### 1.1.3.6. Source code for the used actions

#### 1.1.3.6.1. removeArrayConnection

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.configuration").removeArrayConnection(connection)
;
```

### 1.1.4. Update Unisphere Server instance

Update a registered Unisphere for VMAX instance

#### 1.1.4.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.1.4.2. Inputs

Name	Type	Description
connection*	VMAXPlugin:Connection	Selection connection to be updated
newConnectionName	string	New connection name
newServer	string	FQDN/IP address of unisphere server instance
newPort	number	Port number
newUser	string	Unisphere user name
newPassword	SecureString	User password

\*Parameter is mandatory

### 1.1.4.3. Outputs

Name	Type	Description
newConnection	VMAXPlugin:Connection	

### 1.1.4.4. Attributes

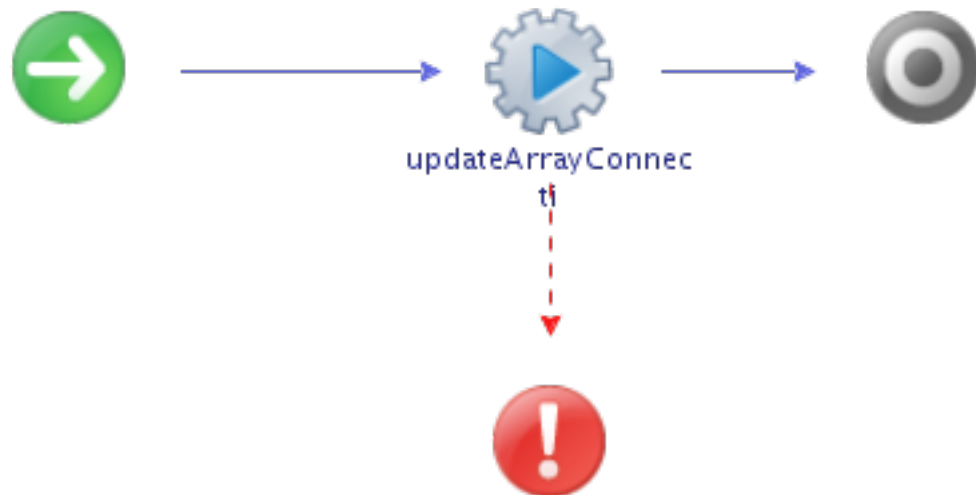
Name	Type	Description
errorCode	string	

### 1.1.4.5. Parameter presentation

Parameter	Description	Parameter Info
connection	Select connection to be updated	Mandatory input: true
newConnectionName	New connection name	Default value: GetAction("com.dell.storage.vmax.configuration", "getConnectionProperties").call(#connection).name
newServer	FQDN/IP address of unisphere server instance	Default value: GetAction("com.dell.storage.vmax.configuration", "getConnectionProperties").call(#connection).server
newPort	Port number	Default value: GetAction("com.dell.storage.vmax.configuration", "getConnectionProperties").call(#connection).port
newUser	Unisphere user name	Default value: GetAction("com.dell.storage.vmax.configuration", "getConnectionProperties").call(#connection).user
newPassword	User password	



### 1.1.4.6. Workflow schema



### 1.1.4.7. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
updateArrayC onnection	WorkflowItemTask		
	WorkflowItemEnd		

### 1.1.4.8. Source code for the used actions

#### 1.1.4.8.1. updateArrayConnection

```
//Auto generated script, cannot be modified !
actionResult =
System.getModule("com.dell.storage.vmax.configuration").updateArrayConnection(connection,n
ewConnectionName,newServer,newPort,newUser,newPassword) ;
```

## 1.2. VMAX Host Management

## 1.2.1. Create Masking View on a VMAX array

Create a Masking View on a VMAX array

### 1.2.1.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.1.2. Inputs

Name	Type	Description
name*	string	Enter name for the new masking view
vmaxArray*	VMAXPlugin:VMAXArray	Select VMAX array
storageGroup*	VMAXPlugin:StorageGroup	Select storage group
portGroup*	VMAXPlugin:PortGroup	Select port group
initiatorGroup*	VMAXPlugin:InitiatorGroup	Select initiator group

\*Parameter is mandatory

### 1.2.1.3. Attributes

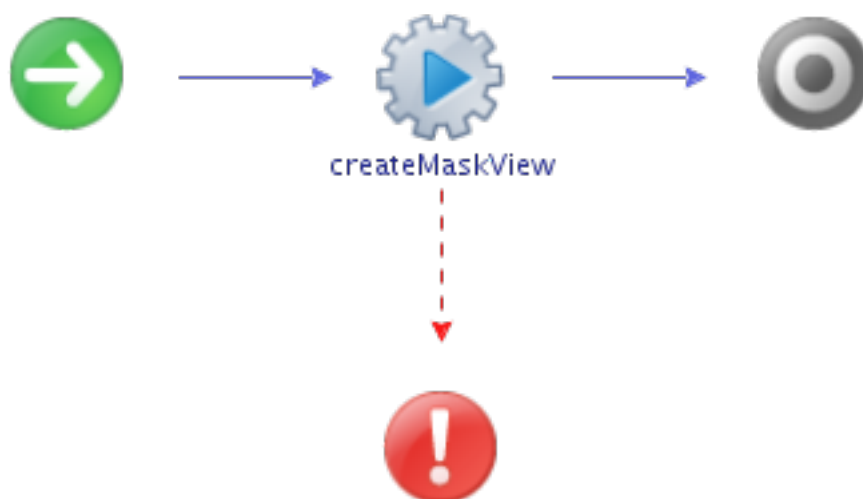
Name	Type	Description
errorCode	string	

### 1.2.1.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	Select VMAX array	Mandatory input: true
name	Enter name for the new masking view	Mandatory input: true

Parameter	Description	Parameter Info
storageGroup	Select storage group	Mandatory input: true
portGroup	Select port group	Mandatory input: true
initiatorGroup	Select initiator group	Mandatory input: true

### 1.2.1.5. Workflow schema



### 1.2.1.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
createMaskView	WorkflowItemTask		
	WorkflowItemEnd		

## 1.2.1.7. Source code for the used actions

### 1.2.1.7.1. createMaskView

```
//Auto generated script, cannot be modified !  
System.getModule("com.dell.storage.vmax.provisioning").createMaskView(array,maskViewName,initiatorGroup,portGroup,storageGroup) ;
```

## 1.2.2. Create Port Group on a VMAX Array

Create Port Group on a VMAX Array

### 1.2.2.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.2.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	The array on which to create the port group
portGroupName*	string	The name of the port group
portsToAdd*	Array/VMAXPlugin:Port	The ports to be added to the port group
director	VMAXPlugin:Director	The director to filter ports on
showOnlinePortsOnly	boolean	Only display ports which are online

\*Parameter is mandatory

### 1.2.2.3. Attributes

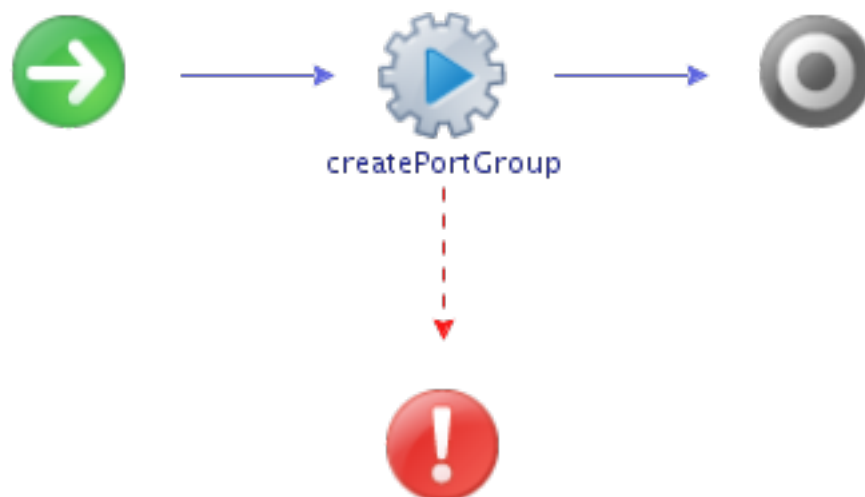
Name	Type	Description
errorCode	string	

### 1.2.2.4. Parameter presentation

Parameter	Description	Parameter Info
-----------	-------------	----------------

Parameter	Description	Parameter Info
vmaxArray	The array on which to create the port group	Mandatory input: true
portGroupName	The name of the port group	Mandatory input: true
portsToAdd	The ports to be added to the port group	Mandatory input: true
director	The director to filter ports on	
showOnlinePortsOnly	Only display ports which are online	Default value: true

### 1.2.2.5. Workflow schema



### 1.2.2.6. Workflow items

Name	Type	Next item	Description

Name	Type	Next item	Description
	WorkflowItemEnd		
createPortGroup	WorkflowItemTask		
	WorkflowItemEnd		

## 1.2.2.7. Source code for the used actions

### 1.2.2.7.1. createPortGroup

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.provisioning").createPortGroup(vmaxArray,portGroupName,portsToAdd) ;
```

## 1.2.3. Create VMAX Host

Create a host on a VMAX array

### 1.2.3.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.3.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	Select the VMAX array on which the host is to be created
hostName*	string	Enter the name for the host
initiators*	Array/VMAXPlugin:WWN	Select the set of initiators to add to the host
volumeSetAddressingEnabled	boolean	Enable Volume Set Addressing
avoidResetBroadcastEnabled	boolean	Enable Avoid Reset Broadcast
environSetEnabled	boolean	Enable Environ Set

Name	Type	Description
disableQResetOnUA	boolean	Disable Queue Reset on UA
scsi3Enabled	boolean	Enable SCSI3
scsiSupport1Enabled	boolean	Enable SCSI Support1
spc2ProtocolVersionEnabled	boolean	Enable SPC2 Protocol Version
openvmsEnabled	boolean	Enable OpenVMS
consistentLunEnabled	boolean	Enable Consistent Luns
setHostFlags	boolean	Advanced Settings

\*Parameter is mandatory

### 1.2.3.3. Attributes

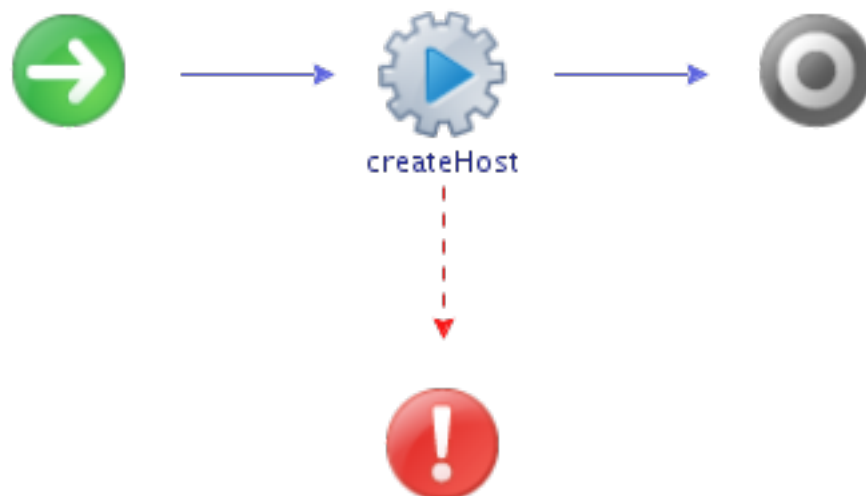
Name	Type	Description
errorCode	string	

### 1.2.3.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	Select the VMAX array on which the host is to be created	Mandatory input: true
initiators	Select the set of initiators to add to the host	Mandatory input: true
hostName	Enter the name for the host	Mandatory input: true
setHostFlags	Advanced Settings	Default value: false
volumeSetAddressingEnabled	Enable Volume Set Addressing	

Parameter	Description	Parameter Info
spc2ProtocolVersionEnabled	Enable SPC2 Protocol Version	
consistentLunEnabled	Enable Consistent Luns	
openvmsEnabled	Enable OpenVMS	
scsiSupport1Enabled	Enable SCSI Support1	
scsi3Enabled	Enable SCSI3	
disableQResetOnUA	Disable Queue Reset on UA	
avoidResetBroadcastEnabled	Enable Avoid Reset Broadcast	
environSetEnabled	Enable Environ Set	

### 1.2.3.5. Workflow schema





### 1.2.3.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
createHost	WorkflowItemTask		
	WorkflowItemEnd		

### 1.2.3.7. Source code for the used actions

#### 1.2.3.7.1. createHost

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.host").createHost(vmaxArray,hostName,initiators,volumeSetAddressingEnabled,avoidResetBroadcastEnabled,enviromSetEnabled,disableQResetOnUA,scsi3Enabled,scsiSupport1Enabled,spc2ProtocolVersionEnabled,openvmsEnabled,consistentLunEnabled,setHostFlags) ;
```

### 1.2.4. Create VMAX Host Group

Create a Host Group on a VMAX array

#### 1.2.4.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.2.4.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	Select the VMAX array to create the host on
hostGroupName*	string	Enter the name for the host
hosts*	Array/VMAXPlugin:UnisphereHost	Enter the list of hosts to add to the host group
volumeSetAddressingEnabled	boolean	Enable Volume Set Addressing

Name	Type	Description
avoidResetBroadcastEnabled	boolean	Enable Avoid Reset Broadcast
environSetEnabled	boolean	Enable Environ Set
disableQResetOnUA	boolean	Disable Queue Reset on UA
scsi3Enabled	boolean	Enable SCSI3
scsiSupport1Enabled	boolean	Enable SCSI Support1
spc2ProtocolVersionEnabled	boolean	Enable SPC2 Protocol Version
openvmsEnabled	boolean	Enable OpenVMS
consistentLunEnabled	boolean	Enable Consistent Luns
setHostGroupFlags	boolean	Set host group flags?

\*Parameter is mandatory

### 1.2.4.3. Attributes

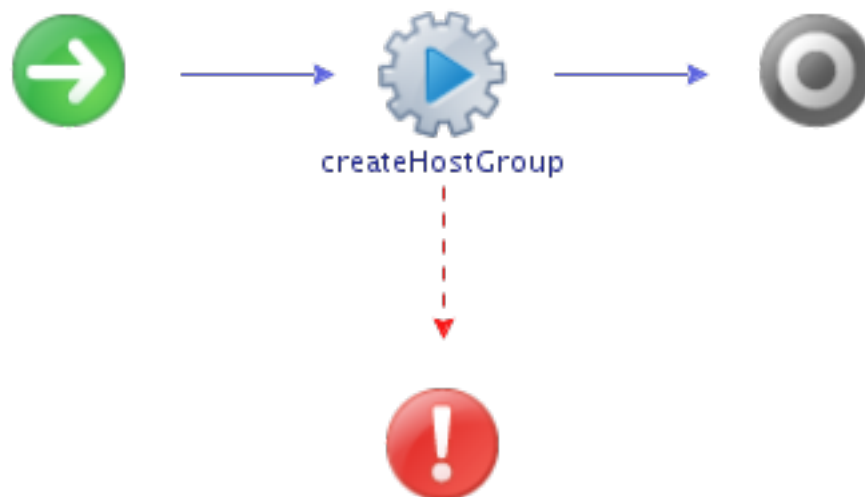
Name	Type	Description
errorCode	string	

### 1.2.4.4. Parameter presentation

Parameter	Description	Parameter Info
volumeSetAddressingEnabled	Enable Volume Set Addressing	
avoidResetBroadcastEnabled	Enable Avoid Reset Broadcast	
environSetEnabled	Enable Environ Set	
disableQResetOnUA	Disable Queue Reset on UA	
scsi3Enabled	Enable SCSI3	

Parameter	Description	Parameter Info
scsiSupport1Enabled	Enable SCSI Support1	
spc2ProtocolVersionEnabled	Enable SPC2 Protocol Version	
openvmsEnabled	Enable OpenVMS	
consistentLunEnabled	Enable Consistent Luns	
vmaxArray	Select the VMAX array to create the host group on	Mandatory input: true
hosts	Enter the list of hosts to add to the host group	Mandatory input: true
hostGroupName	Enter the name for the host group [AlpahNumeric with "_" allowed]	Mandatory input: true
setHostGroupFlags	Advanced Settings	

#### 1.2.4.5. Workflow schema



#### 1.2.4.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
createHostGroup	WorkflowItemTask		
	WorkflowItemEnd		

#### 1.2.4.7. Source code for the used actions

##### 1.2.4.7.1. createHostGroup

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.host").createHostGroup(vmaxArray,hostGroupName,hosts,volumeSetAddressingEnabled,avoidResetBroadcastEnabled,enviromSetEnabled,disableQResetOnUA,scsi3Enabled,scsiSupport1Enabled,spc2ProtocolVersionEnabled,openvmsEnabled,consistentLunEnabled,setHostGroupFlags) ;
```

#### 1.2.5. Delete Masking View on VMAX

Delete Masking View on a VMAX array

### 1.2.5.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.5.2. Inputs

Name	Type	Description
maskingView*	VMAXPlugin:MaskView	The mask view to delete
vmaxArray*	VMAXPlugin:VMAXArray	

\*Parameter is mandatory

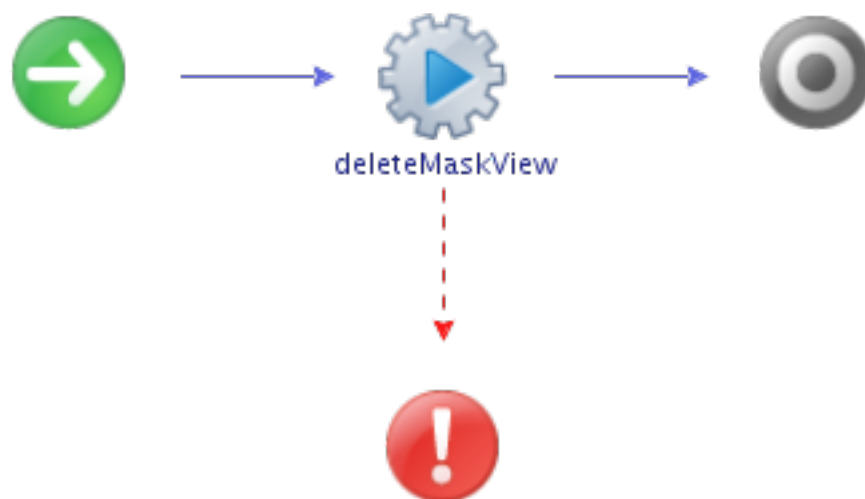
### 1.2.5.3. Attributes

Name	Type	Description
errorCode	string	

### 1.2.5.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
maskingView	The mask view to delete	Mandatory input: true

### 1.2.5.5. Workflow schema



### 1.2.5.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
deleteMaskView	WorkflowItemTask		
	WorkflowItemEnd		

### 1.2.5.7. Source code for the used actions

#### 1.2.5.7.1. deleteMaskView

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.provisioning").deleteMaskView(maskingView) ;
```

## 1.2.6. Delete Port Group on a VMAX array

Delete Port Group on a VMAX array

### 1.2.6.1. Versions

Version	Date	User	Comment
---------	------	------	---------

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

## 1.2.6.2. Inputs

Name	Type	Description
vmaxArray	VMAXPlugin:VMAXArray	The VMAX Array the port group belongs to
portGroup*	VMAXPlugin:PortGroup	The port group to delete

\*Parameter is mandatory

## 1.2.6.3. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	The VMAX Array the port group belongs to	
portGroup	The port group to delete	Mandatory input: true

## 1.2.6.4. Workflow schema



## 1.2.6.5. Workflow items

Name	Type	Next item	Description

Name	Type	Next item	Description
	WorkflowItemEnd		
deletePortGroup	WorkflowItemTask		

## 1.2.6.6. Source code for the used actions

### 1.2.6.6.1. deletePortGroup

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.provisioning").deletePortGroup(portGroup) ;
```

## 1.2.7. Delete VMAX Host

Delete a given Host on a VMAX Array

### 1.2.7.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.7.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
host*	VMAXPlugin:UnisphereHost	The host to delete on the VMAX

\*Parameter is mandatory

### 1.2.7.3. Attributes

Name	Type	Description
errorCode	string	

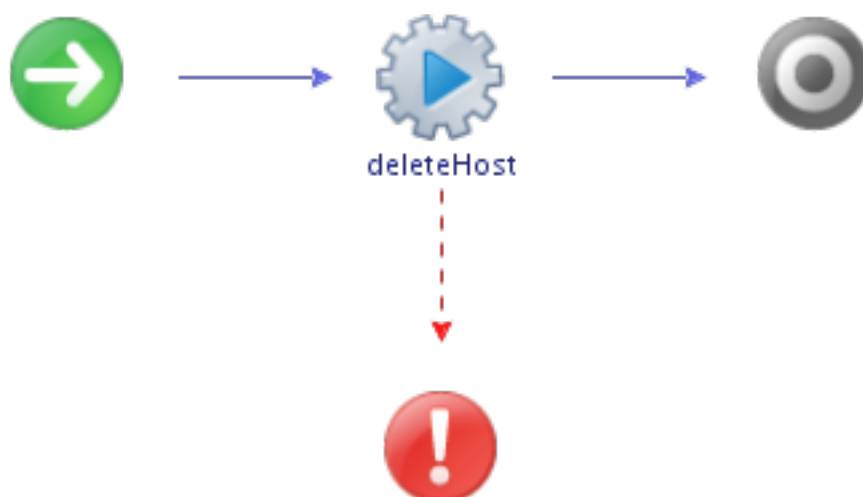
### 1.2.7.4. Parameter presentation

Parameter	Description	Parameter Info
-----------	-------------	----------------



Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
host	The host to delete on the VMAX	Mandatory input: true

### 1.2.7.5. Workflow schema



### 1.2.7.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
deleteHost	WorkflowItemTask		
	WorkflowItemEnd		

### 1.2.7.7. Source code for the used actions

#### 1.2.7.7.1. deleteHost

//Auto generated script, cannot be modified !

```
System.getModule("com.dell.storage.vmax.host").deleteHost(host) ;
```

### 1.2.8. Delete VMAX Host Group

Delete a given Host Group on a VMAX Array

#### 1.2.8.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.2.8.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
hostGroup*	VMAXPlugin:UnisphereHostGroup	The Host Group to delete

\*Parameter is mandatory

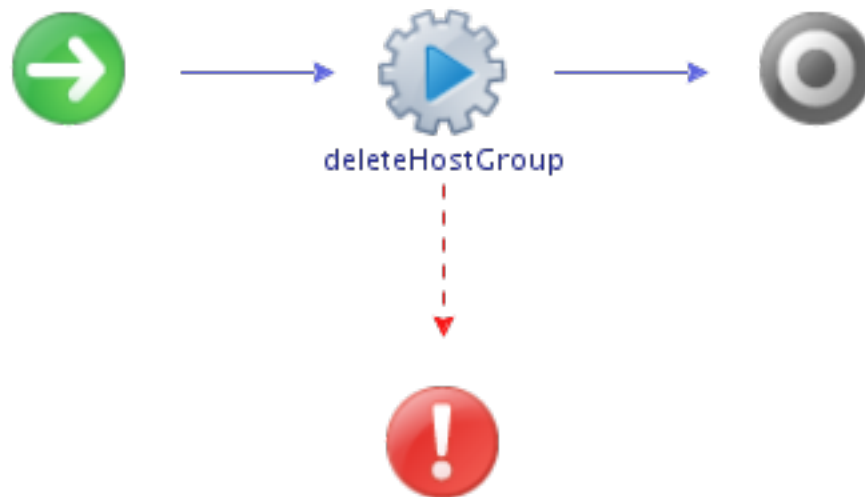
#### 1.2.8.3. Attributes

Name	Type	Description
errorCode	string	

#### 1.2.8.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
hostGroup	The Host Group to delete	Mandatory input: true

#### 1.2.8.5. Workflow schema



### 1.2.8.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
deleteHostGroup	WorkflowItemTask		
	WorkflowItemEnd		

### 1.2.8.7. Source code for the used actions

#### 1.2.8.7.1. deleteHostGroup

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.host").deleteHostGroup(hostGroup) ;
```

### 1.2.9. Edit Port Group on a VMAX array

Modify a Port Group on a VMAX array

#### 1.2.9.1. Versions

Version	Date	User	Comment
---------	------	------	---------

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.9.2. Inputs

Name	Type	Description
vmaxArray	VMAXPlugin:VMAXArray	The VMAX Array the port group belongs to
portGroupToEdit*	VMAXPlugin:PortGroup	The port group to edit
newPortGroupName	string	The new name for the port group
newPortsList	Array/VMAXPlugin:Port	The new list of ports for the port group
director	VMAXPlugin:Director	Filter ports by director
showOnlinePortsOnly	boolean	Show online ports only

\*Parameter is mandatory

### 1.2.9.3. Attributes

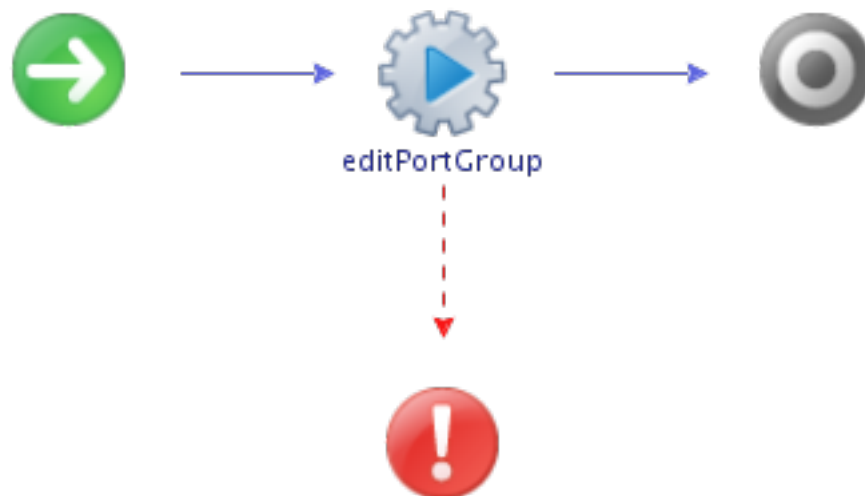
Name	Type	Description
errorCode	string	

### 1.2.9.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	The VMAX Array the port group belongs to	
portGroupToEdit	The port group to edit	Mandatory input: true
newPortGroupName	The new name for the port group	

Parameter	Description	Parameter Info
newPortsList	The new list of ports for the port group	Default value: GetAction("com.dell.storage.vmax.provisioning","getPortsInPortGroup").call( #portGroupToEdit )
director	Filter ports by director	
showOnlinePortsOnly	Show online ports only	Default value: true

### 1.2.9.5. Workflow schema



### 1.2.9.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
editPortGroup	WorkflowItemTask		
	WorkflowItemEnd		

## 1.2.9.7. Source code for the used actions

### 1.2.9.7.1. editPortGroup

```
//Auto generated script, cannot be modified !  
actionResult =  
System.getModule("com.dell.storage.vmax.provisioning").editPortGroup(portGroupToEdit,newPortGroupName,newPortsList) ;
```

## 1.2.10. Edit VMAX Host

Modify a VMAX host

### 1.2.10.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.10.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
host*	VMAXPlugin:UnisphereHost	The host to be edited
newHostName	string	The new name for the host
initiators	Array/VMAXPlugin:WWN	Enter the list of initiators to add to the host
editHostFlags	boolean	Change host flags?
volumeSetAddressingEnabled	boolean	Enable Volume Set Addressing
avoidResetBroadcastEnabled	boolean	Enable Avoid Reset Broadcast
environSetEnabled	boolean	Enable Environ Set
disableQResetOnUA	boolean	Disable Queue Reset on UA

Name	Type	Description
scsi3Enabled	boolean	Enable SCSI3
scsiSupport1Enabled	boolean	Enable SCSI Support1
spc2ProtocolVersionEnabled	boolean	Enable SPC2 Protocol Version
openvmsEnabled	boolean	Enable OpenVMS
consistentLunEnabled	boolean	Enable Consistent Luns

\*Parameter is mandatory

### 1.2.10.3. Attributes

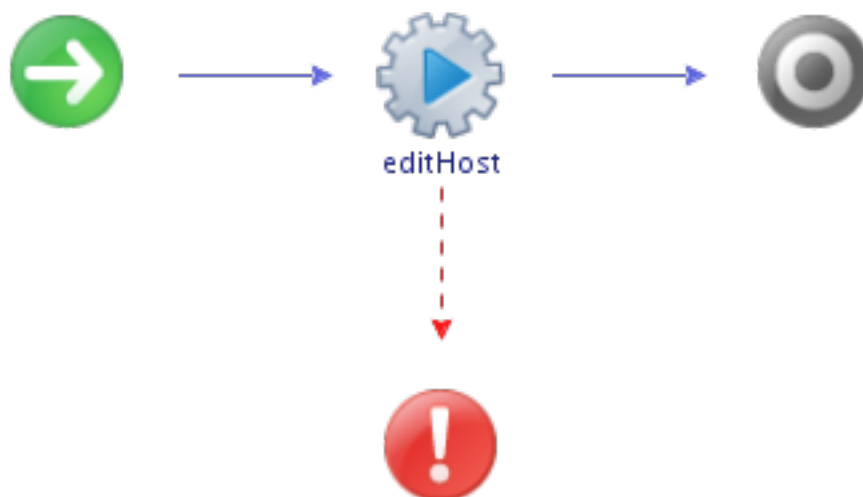
Name	Type	Description
errorCode	string	

### 1.2.10.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
host	The host to be edited	Mandatory input: true
newHostName	The new name for the host	
initiators	Enter the list of initiators to add to the host	Default value: GetAction("com.dell.storage.vmax.host", "getInitiatorsInHost").call(#host )
editHostFlags	Advanced Settings	Default value: false
volumeSetAddressingEnabled	Enable Volume Set Addressing	
avoidResetBroadcastEnabled	Enable Avoid Reset Broadcast	
environSetEnabled	Enable Environ Set	

Parameter	Description	Parameter Info
disableQResetOnUA	Disable Queue Reset on UA	
scsi3Enabled	Enable SCSI3	
scsiSupport1Enabled	Enable SCSI Support1	
spc2ProtocolVersionEnabled	Enable SPC2 Protocol Version	
openvmsEnabled	Enable OpenVMS	
consistentLunEnabled	Enable Consistent Luns	

### 1.2.10.5. Workflow schema



### 1.2.10.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		



Name	Type	Next item	Description
editHost	WorkflowItemTask		
	WorkflowItemEnd		

## 1.2.10.7. Source code for the used actions

### 1.2.10.7.1. editHost

//Auto generated script, cannot be modified !

```
System.getModule("com.dell.storage.vmax.host").editHost(host,newHostName,initiators,editHostFlags,volumeSetAddressingEnabled,avoidResetBroadcastEnabled,enviromSetEnabled,disableQResetOnUA,scsi3Enabled,scsiSupport1Enabled,spc2ProtocolVersionEnabled,openvmsEnabled,consistentLunEnabled) ;
```

## 1.2.11. Edit VMAX Host Group

Modify a VMAX Host Group

### 1.2.11.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.11.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
hostGroup*	VMAXPlugin:UnisphereHostGroup	The host group to be edited
newHostGroupName	string	New name for the host group
hosts	Array/VMAXPlugin:UnisphereHost	Enter the list of hosts to add to the host group
changeHostGroupFlags	boolean	Change host group flags?
volumeSetAddressingEnabled	boolean	Enable Volume Set Addressing

Name	Type	Description
avoidResetBroadcastEnabled	boolean	Enable Avoid Reset Broadcast
environSetEnabled	boolean	Enable Environ Set
disableQResetOnUA	boolean	Disable Queue Reset on UA
scsi3Enabled	boolean	Enable SCSI3
scsiSupport1Enabled	boolean	Enable SCSI Support1
spc2ProtocolVersionEnabled	boolean	Enable SPC2 Protocol Version
openvmsEnabled	boolean	Enable OpenVMS
consistentLunEnabled	boolean	Enable Consistent Luns

\*Parameter is mandatory

### 1.2.11.3. Attributes

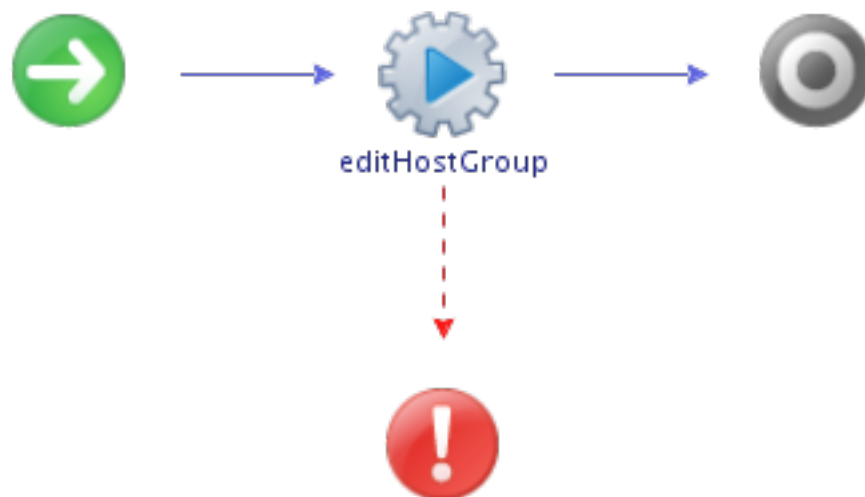
Name	Type	Description
errorCode	string	

### 1.2.11.4. Parameter presentation

Parameter	Description	Parameter Info
volumeSetAddressingEnabled	Enable Volume Set Addressing	
avoidResetBroadcastEnabled	Enable Avoid Reset Broadcast	
environSetEnabled	Enable Environ Set	
disableQResetOnUA	Disable Queue Reset on UA	
scsi3Enabled	Enable SCSI3	
scsiSupport1Enabled	Enable SCSI Support1	

Parameter	Description	Parameter Info
spc2ProtocolVersionEnabled	Enable SPC2 Protocol Version	
openvmsEnabled	Enable OpenVMS	
consistentLunEnabled	Enable Consistent Luns	
vmaxArray	vmaxArray	Mandatory input: true
hostGroup	The host group to be edited	Mandatory input: true
newHostGroupName	New name for the host group	
hosts	Enter the list of hosts to add to the host group	Default value: GetAction("com.dell.storage.vmax.host","getHostsInHostGroup").call(#hostGroup )
changeHostGroupFlags	Advanced Settings	Default value: false

### 1.2.11.5. Workflow schema



## 1.2.11.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
editHostGroup	WorkflowItemTask		
	WorkflowItemEnd		

## 1.2.11.7. Source code for the used actions

### 1.2.11.7.1. editHostGroup

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.host").editHostGroup(hostGroup,newHostGroupName,hosts,changeHostGroupFlags,volumeSetAddressingEnabled,avoidResetBroadcastEnabled,enviro
nSetEnabled,disableQResetOnUA,scsi3Enabled,scsiSupport1Enabled,spc2ProtocolVersionEnabled,openvmsEnabled,consistentLunEnabled) ;
```

## 1.2.12. Rename Masking View on VMAX

Rename Masking View on a VMAX array

### 1.2.12.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.2.12.2. Inputs

Name	Type	Description
maskView*	VMAXPlugin:MaskView	Select the mask view to edit
newName*	string	New name for the mask view
vmaxArray*	VMAXPlugin:VMAXArray	The array which contains the mask view

\*Parameter is mandatory

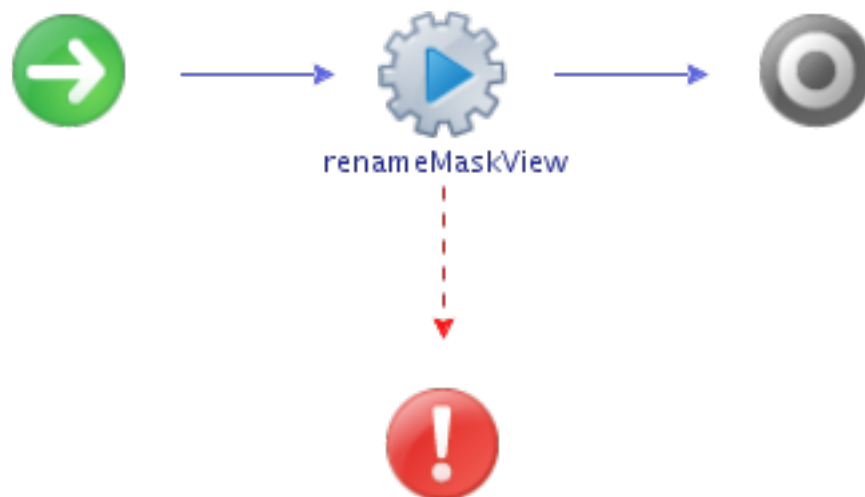
### 1.2.12.3. Attributes

Name	Type	Description
errorCode	string	

### 1.2.12.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	The array which contains the mask view	Mandatory input: true
maskView	Select the mask view to edit	Mandatory input: true
newName	New name for the mask view	Mandatory input: true

### 1.2.12.5. Workflow schema



### 1.2.12.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
renameMaskView	WorkflowItemTask		
	WorkflowItemEnd		

### 1.2.12.7. Source code for the used actions

#### 1.2.12.7.1. renameMaskView

//Auto generated script, cannot be modified !

```
System.getModule("com.dell.storage.vmax.provisioning").renameMaskView(maskView,newName) ;
```

## VMAX Snapshot management

### 1.3. Create VMAX Storage Device Snapshot

Create a snapshot of a VMAX storage volume

#### 1.3.1.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.3.1.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
searchByVolumetag	boolean	
volumeTag	string	
storageGroup	VMAXPlugin:StorageGroup	
vmaxDevice*	VMAXPlugin:VmaxDevice	
nameTheSnapshot	boolean	
snapName	string	

\*Parameter is mandatory

### 1.3.1.3. Outputs

Name	Type	Description
result	string	

### 1.3.1.4. Attributes

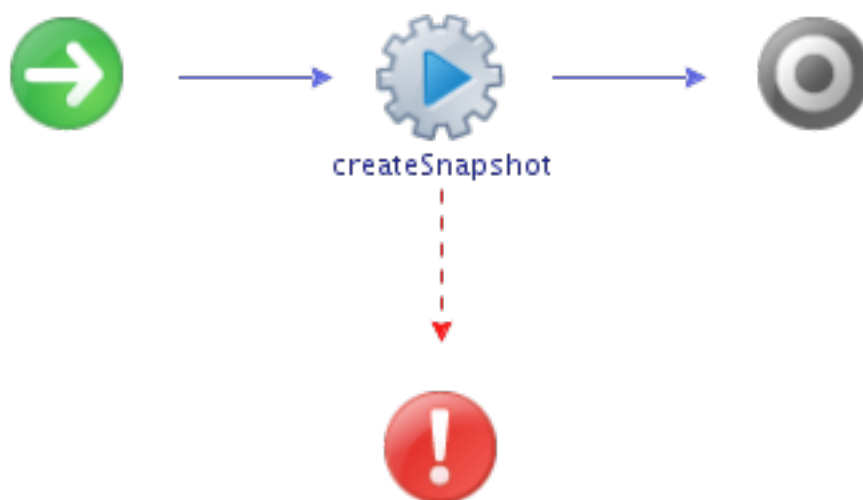
Name	Type	Description
errorCode	string	

### 1.3.1.5. Parameter presentation

Parameter	Description	Parameter Info

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
searchByVolumetag	Search Vmax Device By Volume Tag	Default value: true
volumeTag	volumeTag	
storageGroup	storageGroup	
vmaxDevice	vmaxDevice	Mandatory input: true
nameTheSnapshot	nameTheSnapshot	Default value: true
snapName	snapName	

### 1.3.1.6. Workflow schema



### 1.3.1.7. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		



## Delete VMAX Storage Device Snapshot

Name	Type	Next item	Description
createSnapshot	WorkflowItemTask		
	WorkflowItemEnd		

### 1.3.1.8. Source code for the used actions

#### 1.3.1.8.1. createSnapshot

```
//Auto generated script, cannot be modified !
actionResult =
System.getModule("com.dell.storage.vmax.snapshot").createSnapshot(vmaxDevice,vmaxArray,snapshotName) ;
```

### 1.3.2. Delete VMAX Storage Device Snapshot

Delete a given storage device snapshot on a VMAX array

#### 1.3.2.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.3.2.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
VolumeTagFlag	boolean	
volumeTag	string	
vmaxStorageGroup	VMAXPlugin:StorageGroup	
vmaxDevice*	VMAXPlugin:VmaxDevice	
snapshot*	string	

\*Parameter is mandatory

### 1.3.2.3. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
VolumeTagFlag	VolumeTagFlag	Default value: true
volumeTag	volumeTag	
vmaxStorageGroup	vmaxStorageGroup	
vmaxDevice	vmaxDevice	Mandatory input: true
snapshot	snapshot	Mandatory input: true

### 1.3.2.4. Workflow schema



### 1.3.2.5. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
deleteSnapshot	WorkflowItemTask		

### 1.3.2.6. Source code for the used actions

#### 1.3.2.6.1. deleteSnapshot

//Auto generated script, cannot be modified !

```
System.getModule("com.dell.storage.vmax.snapshot").deleteSnapshot(vmaxArray,vmaxDevice,snapshot) ;
```

### 1.3.3. Link VMAX Device Snapshot

Link a VMAX Device Snapshot to a target

#### 1.3.3.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.3.3.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	Array on which Snapshot is Taken
volumeTagFlag	boolean	Search device by Volume Tag?
volumeTag	string	
vmaxDevice*	VMAXPlugin:VmaxDevice	Device Having the Snapshot
snapshot*	string	

\*Parameter is mandatory

#### 1.3.3.3. Outputs

Name	Type	Description
actionResult	string	

#### 1.3.3.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	Array on which Snapshot is Taken	Mandatory input: true

Parameter	Description	Parameter Info
volumeTagFlag	Search device by Volume Tag?	Default value: true
volumeTag	volumeTag	
vmaxDevice	Device Having the Snapshot	Mandatory input: true
snapshot	snapshot	Mandatory input: true

### 1.3.3.5. Workflow schema



### 1.3.3.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
linkSnapshot	WorkflowItemTask		

### 1.3.3.7. Source code for the used actions

#### 1.3.3.7.1. linkSnapshot

```
//Auto generated script, cannot be modified !
actionResult =
System.getModule("com.dell.storage.vmax.snapshot").linkSnapshot(vmaxArray,vmaxDevice,snapshot) ;
```

### 1.3.4. Rename VMAX Storage Device Snapshot

Rename a device snapshot on a VMAX array

### 1.3.4.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.3.4.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
VolumeTagFlag	boolean	
VolumeTag	string	
StorageGroup	VMAXPlugin:StorageGroup	
vmaxDevice*	VMAXPlugin:VmaxDevice	
snapshot*	string	
newSnapshotName*	string	

\*Parameter is mandatory

### 1.3.4.3. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
VolumeTagFlag	VolumeTagFlag	Default value: true
VolumeTag	VolumeTag	
StorageGroup	StorageGroup	
vmaxDevice	vmaxDevice	Mandatory input: true
snapshot	snapshotName	Mandatory input: true

Parameter	Description	Parameter Info
newSnapshotName	newSnapshotName	Mandatory input: true

#### 1.3.4.4. Workflow schema



#### 1.3.4.5. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
renameSnapshot	WorkflowItemTask		

#### 1.3.4.6. Source code for the used actions

##### 1.3.4.6.1. renameSnapshot

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.snapshot").renameSnapshot(vmaxArray,vmaxDevice,snapshotName,newSnapshotName) ;
```

#### 1.3.5. Restore VMAX Storage device from a Snapshot

Restore / Revert a VMAX device to a snapshot

##### 1.3.5.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.3.5.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	The VMAX array to which the device belongs
vmaxDevice*	VMAXPlugin:VmaxDevice	The source VMAX device
snapshotName*	string	The snapshot name to revert the source device from
searchDeviceByTag	boolean	Search for source device by volume tag?
volumeTag	string	Volume Tag

\*Parameter is mandatory

### 1.3.5.3. Attributes

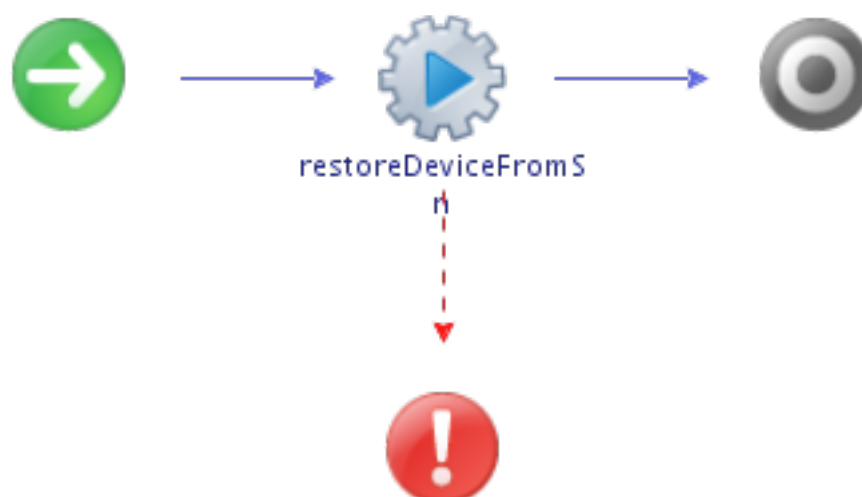
Name	Type	Description
errorCode	string	

### 1.3.5.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	The VMAX array to which the device belongs	Mandatory input: true
searchDeviceByTag	Search for source device by volume tag?	Default value: true
volumeTag	Volume Tag	
vmaxDevice	The source VMAX device	Mandatory input: true

Parameter	Description	Parameter Info
snapshotName	The snapshot name to revert the source device from	Mandatory input: true

### 1.3.5.5. Workflow schema



### 1.3.5.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
restoreDeviceFromSnapshot	WorkflowItemTask		
	WorkflowItemEnd		

### 1.3.5.7. Source code for the used actions

#### 1.3.5.7.1. restoreDeviceFromSnapshot



```
//Auto generated script, cannot be modified !
```

```
System.getModule("com.dell.storage.vmax.snapshot").restoreDeviceFromSnapshot(vmaxDevice,snapshotName);
```

### 1.3.6. Unlink a target device from VMAX Device Snapshot

Unlink a target device from the snapshot

#### 1.3.6.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.3.6.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
volumeTagFlag	boolean	
volumeTag	string	
vmaxDevice*	VMAXPlugin:VmaxDevice	
snapshot*	string	
LinkedTarget*	VMAXPlugin:VmaxDevice	
deleteTargetDevice	boolean	

\*Parameter is mandatory

#### 1.3.6.3. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
volumeTagFlag	volumeTagFlag	Default value: true

Parameter	Description	Parameter Info
volumeTag	volumeTag	
vmaxDevice	vmaxDevice	Mandatory input: true
snapshot	snapshot	Mandatory input: true
LinkedTarget	LinkedTarget	Mandatory input: true
deleteTargetDevice	deleteTargetDevice	Default value: __NULL__

### 1.3.6.4. Workflow schema



### 1.3.6.5. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
unlinkSnapshot	WorkflowItemTask		

### 1.3.6.6. Source code for the used actions

#### 1.3.6.6.1. unlinkSnapshot

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.snapshot").unlinkSnapshot(vmaxArray,snapshot,srcDevice,tgtDevice,deleteTarget) ;
```

## 1.4. VMAX Storage Management

### 1.4.1. Add device to VMAX Storage Group

Add an existing device to a Storage Group on a VMAX array

### 1.4.1.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.4.1.2. Inputs

Name	Type	Description
array*	VMAXPlugin:VMAXArray	VMAX Array
device*	VMAXPlugin:VmaxDevice	VMAX Storage Device
storageGroup*	VMAXPlugin:StorageGroup	VMAX Storage Group to which the device is to be added
volumeTag	string	The Volume Tag
searchByVolumeTag	boolean	Search for Volume By Tag?

\*Parameter is mandatory

### 1.4.1.3. Attributes

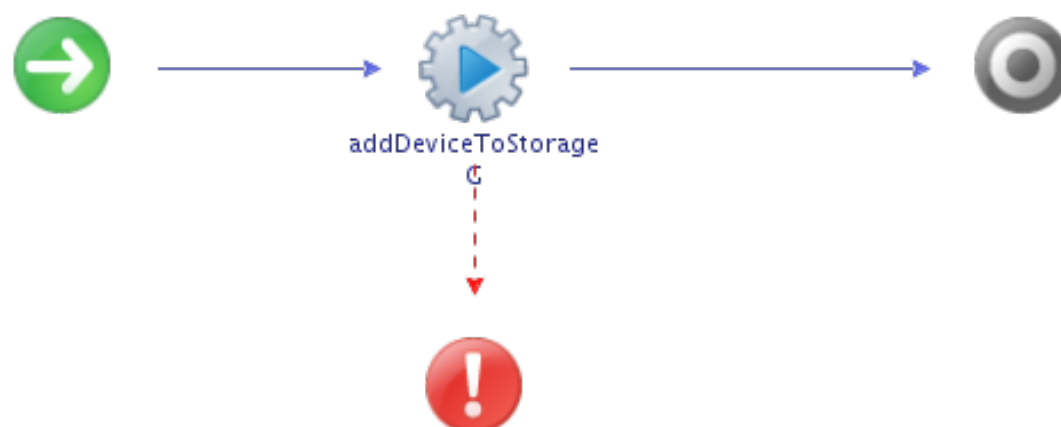
Name	Type	Description
errorCode	string	

### 1.4.1.4. Parameter presentation

Parameter	Description	Parameter Info
array	VMAX Array	Mandatory input: true
searchByVolumeTag	Search for Volume By Tag?	Default value: __NULL__
volumeTag	The Volume Tag	
device	VMAX Storage Device	Mandatory input: true

Parameter	Description	Parameter Info
storageGroup	VMAX Storage Group	Mandatory input: true

#### 1.4.1.5. Workflow schema



#### 1.4.1.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
addDeviceToStorageGroup	WorkflowItemTask		
	WorkflowItemEnd		

#### 1.4.1.7. Source code for the used actions

##### 1.4.1.7.1. addDeviceToStorageGroup

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.provisioning").addDeviceToStorageGroup(device, storageGroup) ;
```

#### 1.4.2. Create Storage Device

### 1.4.2.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.4.2.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	Select VMAX Array from where storage needs to be provisioned
storageGroup*	VMAXPlugin:StorageGroup	Select Storage group
SizeUnit	string	Device Size Unit(MB or GB)
sizeOfDevice*	number	Enter Device size
VolumeNameFlag	boolean	Attach Volume Tag to the device
VolumeName	string	Volume Tag that needs to be attached to Device
CreateMultipleDevices	boolean	
NumOfDevices	number	

\*Parameter is mandatory

### 1.4.2.3. Outputs

Name	Type	Description
deviceName	Array/VMAXPlugin:VmaxDevice	

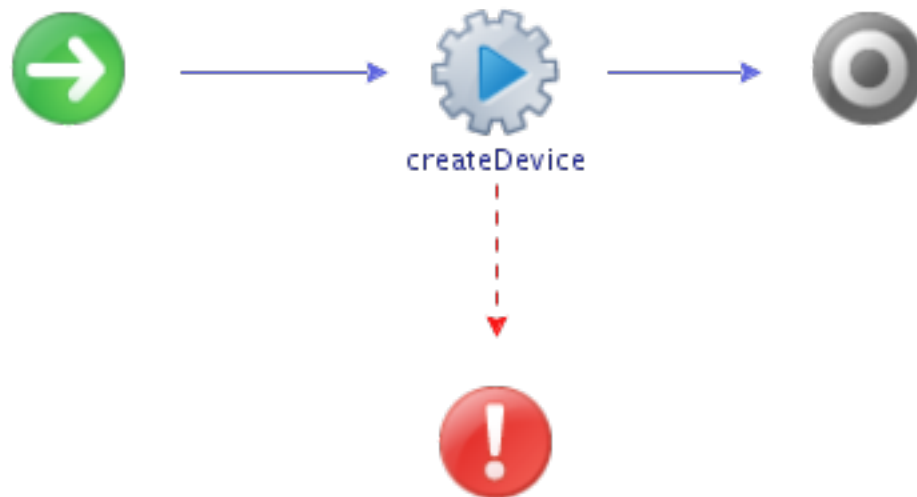
### 1.4.2.4. Attributes

Name	Type	Description
errorCode	string	

### 1.4.2.5. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	Select VMAX Array from where storage needs to be provisioned	Mandatory input: true
storageGroup	Select Storage group	Mandatory input: true
sizeOfDevice	Enter size	Mandatory input: true
SizeUnit	Device Size Unit(MB or GB)	Default value: GB
CreateMultipleDevices	CreateMultipleDevices	Default value: __NULL__
NumOfDevices	NumOfDevices	Default value: 1.0
VolumeNameFlag	Attach Volume Tag to the device	Default value: __NULL__
VolumeName	Volume tag that will be associated with the device (In case of multiple device creation, this tag is used as the base name)	

### 1.4.2.6. Workflow schema



### 1.4.2.7. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
createDevice	WorkflowItemTask		
	WorkflowItemEnd		

### 1.4.2.8. Source code for the used actions

#### 1.4.2.8.1. createDevice

```
//Auto generated script, cannot be modified !  
actionResult =  
System.getModule("com.dell.storage.vmax.provisioning").createDevice(vmaxArray,storageGroup  
,sizeUnit,deviceSize,volumeTag,numberOfdevices) ;
```

### 1.4.3. Create VMAX Storage Group

Create a Storage Group on a VMAX array

#### 1.4.3.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.4.3.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	The VMAX Array to create the storage group on
storageGroupName*	string	The name for the Storage Group

\*Parameter is mandatory

### 1.4.3.3. Attributes

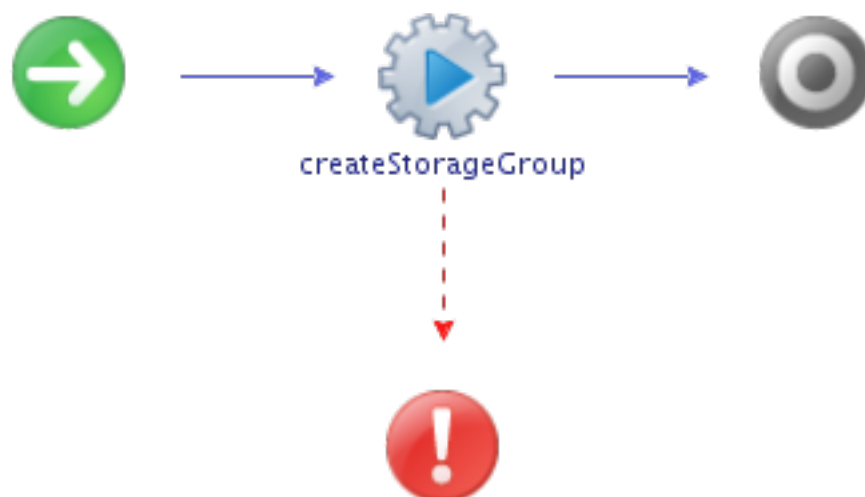
Name	Type	Description
errorCode	string	

### 1.4.3.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	The VMAX Array to create the storage group on	Mandatory input: true
storageGroupName	The name for the Storage Group	Mandatory input: true

### 1.4.3.5. Workflow schema





### 1.4.3.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
createStorageGroup	WorkflowItemTask		
	WorkflowItemEnd		

### 1.4.3.7. Source code for the used actions

#### 1.4.3.7.1. createStorageGroup

```
//Auto generated script, cannot be modified !  
System.getModule("com.dell.storage.vmax.provisioning").createStorageGroup(vmaxArray,storageGroupNames);
```

### 1.4.4. Delete VMAX Storage Device

Delete a device on a VMAX array (remove from storage group)

#### 1.4.4.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.4.4.2. Inputs

Name	Type	Description
VMAXArray*	VMAXPlugin:VMAXArray	
VolumeTagFlag	boolean	
storageGroup	VMAXPlugin:StorageGroup	
VolumeTag	string	
vmaxDevice*	VMAXPlugin:VmaxDevice	

\*Parameter is mandatory

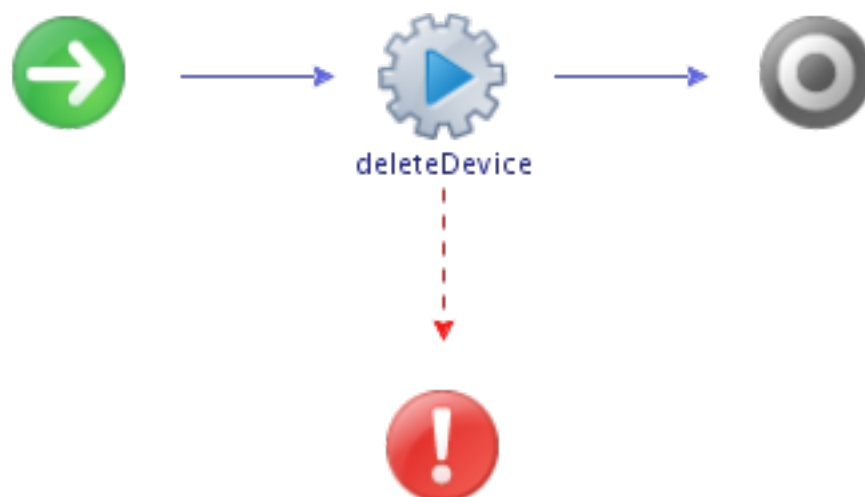
### 1.4.4.3. Attributes

Name	Type	Description
errorCode	string	

### 1.4.4.4. Parameter presentation

Parameter	Description	Parameter Info
VMAXArray	VMAXArray	Mandatory input: true
VolumeTagFlag	VolumeTagFlag	Default value: true
VolumeTag	VolumeTag	
storageGroup	storageGroup	
vmaxDevice	vmaxDevice	Mandatory input: true

### 1.4.4.5. Workflow schema



### 1.4.4.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
deleteDevice	WorkflowItemTask		
	WorkflowItemEnd		

### 1.4.4.7. Source code for the used actions

#### 1.4.4.7.1. deleteDevice

```
//Auto generated script, cannot be modified !  
System.getModule("com.dell.storage.vmax.provisioning").deleteDevice(vmaxArray,vmaxDevice)  
;
```

## 1.4.5. Delete VMAX Storage Group

Delete a given Storage Group on a VMAX array

### 1.4.5.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.4.5.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	The VMAX Array to which the storage group belongs
storageGroup*	VMAXPlugin:StorageGroup	The Storage Group to delete

\*Parameter is mandatory

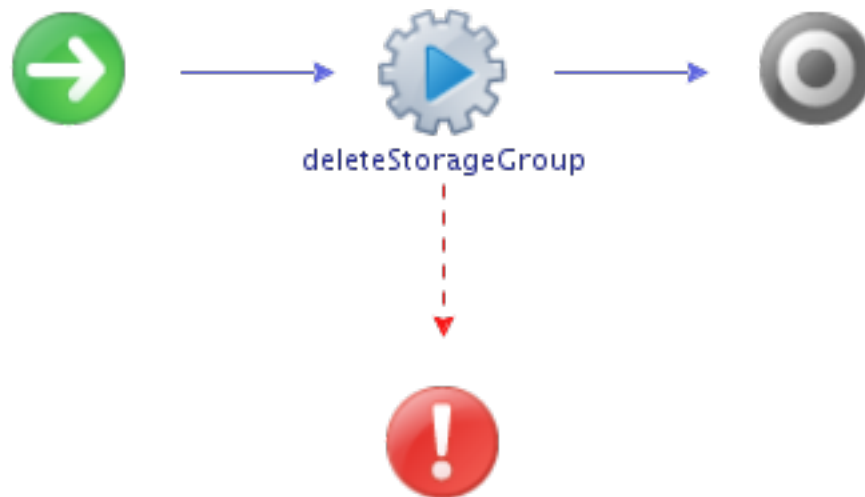
### 1.4.5.3. Attributes

Name	Type	Description
errorCode	string	

### 1.4.5.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	The VMAX Array to which the storage group belongs	Mandatory input: true
storageGroup	The Storage Group to delete	Mandatory input: true

### 1.4.5.5. Workflow schema



### 1.4.5.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
deleteStorageGroup	WorkflowItemTask		
	WorkflowItemEnd		

### 1.4.5.7. Source code for the used actions

#### 1.4.5.7.1. deleteStorageGroup

```
//Auto generated script, cannot be modified !  
System.getModule("com.dell.storage.vmax.provisioning").deleteStorageGroup(vmaxArray, storageGroup);
```

### 1.4.6. Remove device from VMAX storage group

Remove a VMAX storage device from a Storage Group that it belongs to

#### 1.4.6.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.4.6.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
vmaxDevice*	VMAXPlugin:VmaxDevice	

\*Parameter is mandatory

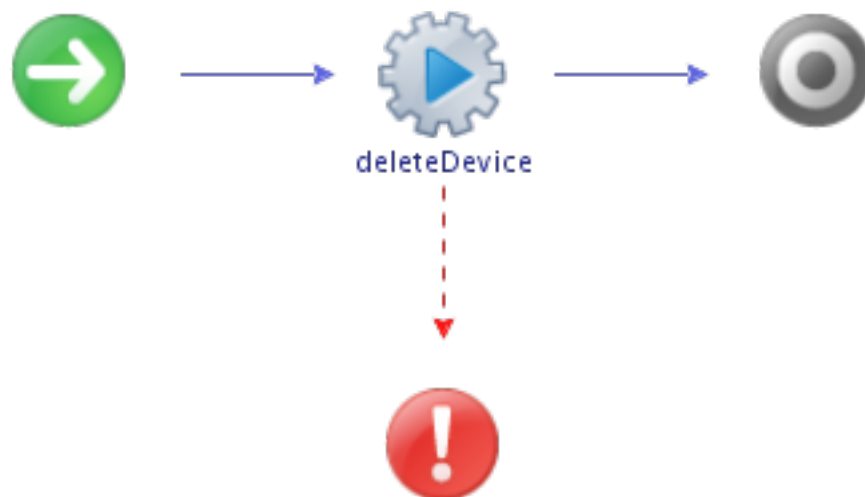
### 1.4.6.3. Attributes

Name	Type	Description
errorCode	string	

### 1.4.6.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
vmaxDevice	vmaxDevice	Mandatory input: true

### 1.4.6.5. Workflow schema



## 1.4.6.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
deleteDevice	WorkflowItemTask		
	WorkflowItemEnd		

## 1.4.6.7. Source code for the used actions

### 1.4.6.7.1. deleteDevice

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.provisioning").deleteDevice(vmaxArray,vmaxDevice)
;
```

## 1.4.7. Resize VMAX Storage Device

Expand a device on a VMAX

### 1.4.7.1. Versions

Version	Date	User	Comment
---------	------	------	---------

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

## 1.4.7.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
volumeTagFlag	boolean	
VolumeTag	string	
vmaxSG	VMAXPlugin:StorageGroup	
vmaxDevice*	VMAXPlugin:VmaxDevice	
DeviceSizeUnit	string	
newSize*	number	

\*Parameter is mandatory

## 1.4.7.3. Attributes

Name	Type	Description
errorCode	string	

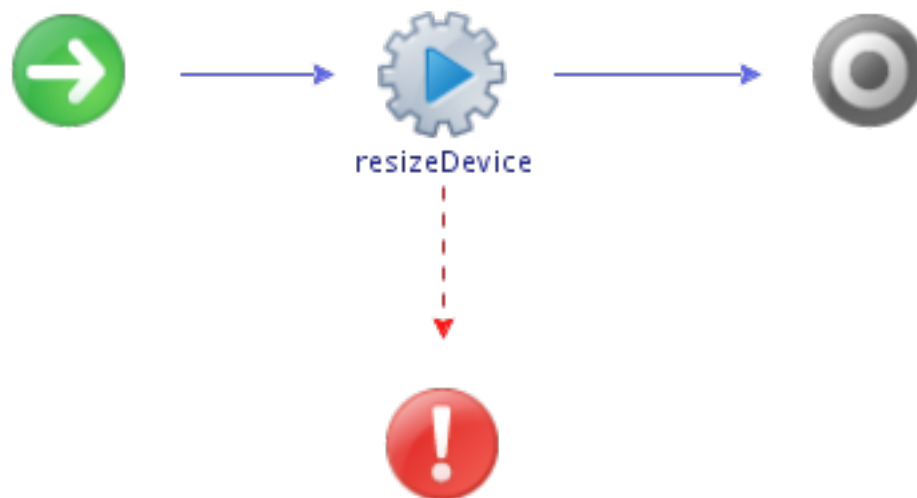
## 1.4.7.4. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
volumeTagFlag	Identify Storage Device from volumeTag or by SG	Default value: false
VolumeTag	VolumeTag	



Parameter	Description	Parameter Info
vmaxSG	vmaxSG	
vmaxDevice	vmaxDevice	Mandatory input: true
newSize	newSize of the Device	Mandatory input: true
DeviceSizeUnit	Device Size Unit in MB or GB	Default value: GB

### 1.4.7.5. Workflow schema



### 1.4.7.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
resizeDevice	WorkflowItemTask		
	WorkflowItemEnd		

## 1.4.7.7. Source code for the used actions

### 1.4.7.7.1. resizeDevice

```
//Auto generated script, cannot be modified !  
System.getModule("com.dell.storage.vmax.provisioning").resizeDevice(vmaxArray,vmaxDevice,newSizeUnit,newSize) ;
```

## 1.5. VMAX-vSphere integration workflows

### 1.5.1. Create Snapshot of a VMFS datastore hosted on VMAX device

Create a crash consistent snapshot of a VMFS datastore hosted on a VMAX device

#### 1.5.1.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.5.1.2. Inputs

Name	Type	Description
vcDatastore*	VC:Datastore	Select the VMFS datastore to create a snapshot

\*Parameter is mandatory

#### 1.5.1.3. Outputs

Name	Type	Description
snapshotName	string	

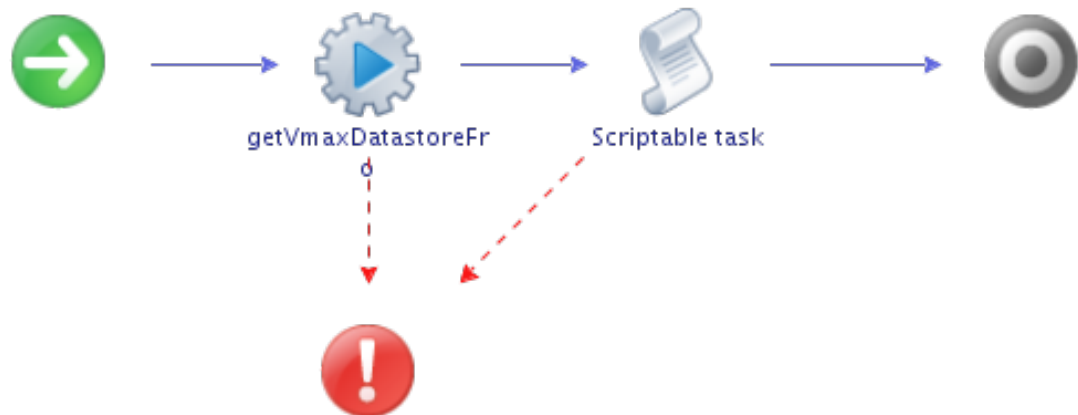
#### 1.5.1.4. Attributes

Name	Type	Description
datastore	VMAXPlugin:DataStore	
errorCode	string	

### 1.5.1.5. Parameter presentation

Parameter	Description	Parameter Info
vcDatastore	Select the VMFS datastore to create a snapshot	Mandatory input: true

### 1.5.1.6. Workflow schema



### 1.5.1.7. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
Scriptable task	WorkflowItemTask		
getVmaxDatastoreFromVcDatastore	WorkflowItemTask	Scriptable task	Convert a vCenter Datastore object to a VMAX datastore object
	WorkflowItemEnd		

### 1.5.1.8. Source code for the used actions

### 1.5.1.8.1. Scriptable task

```
System.log("Creating storage snapshot for VMFS datastore " + datastore.displayName);
try
{
    var vmaxList = System.getModule("com.dell.storage.vmax.array").getVmaxArrays();
    var vmax = null;
    for each ( var array in vmaxList )
    {
        if ( array.displayName.equals(datastore.arrayId) )
        {
            vmax = array;
            break;
        }
    }
    if ( !vmax )
        throw new VMAXPluginVmaxException("Failed to find the VMAX array: " +
datastore.arrayId);

    snapshotName = VMAXPluginVMAXManager.createSnapshot(vmax,datastore);
}
catch(e)
{
    System.error("Failed to create datastore snapshot: " + e.message);
    throw e.message;
}
```

### 1.5.1.8.2. getVmaxDatastoreFromVcDatastore

```
//Auto generated script, cannot be modified !
actionResult =
System.getModule("com.dell.storage.vmax.vmware").getVmaxDatastoreFromVcDatastore(vcDatastore);
```

## 1.5.2. Create Snapshot Schedule for VMFS datastore hosted on VMAX

Start scheduled snapshot creation for VMFS datastore hosted on a VMAX array

### 1.5.2.1. Versions

Version	Date	User	Comment

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.5.2.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	
dataStore*	VMAXPlugin:DataStore	
rpo	number	
retentionPeriod	number	
daily	boolean	
weekly	boolean	
monthly	boolean	
quarterly	boolean	

\*Parameter is mandatory

### 1.5.2.3. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	vmaxArray	Mandatory input: true
dataStore	dataStore	Mandatory input: true
rpo	Recovery point objective in minutes	
retentionPeriod	For how many hours you want to retain a snapshot	Default value: 24

Parameter	Description	Parameter Info
daily	Retain one snapshot daily for seven days	Default value: false
weekly	Retain one snapshot weekly for 5 weeks	Default value: __NULL__
monthly	Retain one snapshot monthly for a quarter	Default value: __NULL__
quarterly	Retain one snapshot quarterly for a year	Default value: __NULL__

#### 1.5.2.4. Workflow schema



#### 1.5.2.5. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
Scriptable task	WorkflowItemTask		

#### 1.5.2.6. Source code for the used actions

##### 1.5.2.6.1. Scriptable task

```
System.log("Starting to add schedule for datastore: " + datastore);
var schedule = new VMAXPluginSnapshotScheduleInfo() ;
schedule.dsUuid = datastore.uuid;
schedule.vmaxId = datastore.arrayId;
schedule.retentionPeriodHours = retentionPeriod;
schedule.snapshotIntervalMins = rpo;
schedule.retentionDaysSet = daily;
if ( daily )
{
    schedule.retentionDaysNumOfCopies = 7;
}
schedule.retentionWeeksSet = weekly;
schedule.retentionMonthsSet = monthly;
schedule.retentionQuartersSet = quarterly;
VMAXPluginSnapshotScheduleManager.add(schedule);
System.log("Successfully added snapshot schedule");
```

### 1.5.3. Create VMAX Host for an ESXi server

Create a VMAX host using the free initiators (HBAs) from the provided ESX server

#### 1.5.3.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.5.3.2. Inputs

Name	Type	Description
esx*	VC:HostSystem	ESXi server
vmaxarray*	VMAXPlugin:VMAXArray	VMAX array
name*	string	Host name

\*Parameter is mandatory

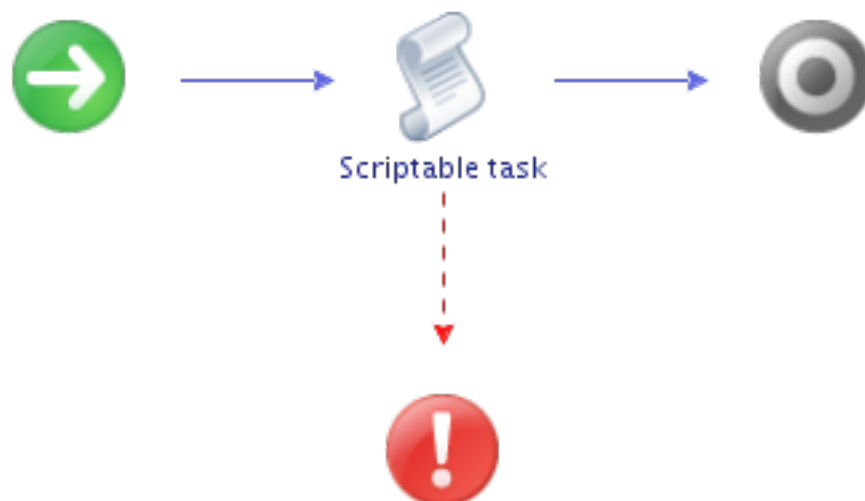
#### 1.5.3.3. Attributes

Name	Type	Description
errorCode	string	

### 1.5.3.4. Parameter presentation

Parameter	Description	Parameter Info
esx	ESXi server	Mandatory input: true
vmaxarray	VMAX array	Mandatory input: true
name	Host name	Mandatory input: true

### 1.5.3.5. Workflow schema



### 1.5.3.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
Scriptable task	WorkflowItemTask		
	WorkflowItemEnd		



#### 1.5.3.7. Source code for the used actions

#### 1.5.3.7.1. Scriptable task

[illegible]

```
function getHostForWWN(wwn, vmax)
{
    var hostlist = System.getModule("com.dell.storage.vmax.host").getAllHosts(vmax)

    for ( var i=0; i < hostlist.length; i++ )
    {
        var host = hostlist[i];
        //System.log("Checking if wwn " + wwn + " is in initiator group " + host.hostName);
        var wwns = System.getModule("com.dell.storage.vmax.host").getInitiatorsInHost(host);
        for each ( var vwwn in wwns )
        {
            if ( vwwn.wwn.equals(wwn) )
            {
                System.log("Found " + wwn + " in initiator group " + host.hostName);
                return host;
            }
        }
    }
}

function getWWNs(hostname)
{
    if ( !hostname )
        throw "host not found";

    var hostStorageSystem = VcPlugin.toManagedObject( hostname,
hostname.configManager.storageSystem );
    var wwnList = new Array();
    var i = 0;
    for each (var hba in hostStorageSystem.storageDeviceInfo.hostBusAdapter)
    {
        //System.log('Device Name: ' + hba.device);
        //System.log('Device Type: ' + hba.driver);
        if (hba instanceof VcHostFibreChannelHba)
        {
            //System.log('FC HBA: ' + hba.portWorldWideNameHex);
            //wwnList[i++] = hba.portWorldWideNameHex;
            var wwn = hba.portWorldWideNameHex;
            wwnList.push(wwn.replace(/:/g, ''));
        }
    }

    return wwnList;
}
```

## 1.5.4. Create VMAX Host Group for a vSphere cluster

Create a VMAX Host Group from a vSphere Cluster. Creates VMAX hosts for the individual ESX servers which are part of the server if necessary

### 1.5.4.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.5.4.2. Inputs

Name	Type	Description
cluster*	VC:ClusterComputeResource	vSphere cluster name
vmaxarray*	VMAXPlugin:VMAXArray	VMAX array
name*	string	Host group name

\*Parameter is mandatory

### 1.5.4.3. Parameter presentation

Parameter	Description	Parameter Info
cluster	vSphere cluster name	Mandatory input: true
vmaxarray	VMAX array	Mandatory input: true
name	Host group name	Mandatory input: true

### 1.5.4.4. Workflow schema



### 1.5.4.5. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
Scriptable task	WorkflowItemTask		

### 1.5.4.6. Source code for the used actions

#### 1.5.4.6.1. Scriptable task

```
if(!cluster) throw "vCenter Cluster parameter is null."

// Get all the ESXi hosts in the cluster
var hosts = cluster.host;
System.log("Found " + hosts.length + " ESXi hosts in the cluster");

var hostlist = new Array();
// Get WWNs of all the initiators for each ESXi host
for each (var host in hosts) {
    System.log("Get initiators for " + host.name);
    var wwns = getWWNs(host);

    if ( !wwns ) throw "Failed to get initiator list for the ESXi host " + host.name;

    var initlist = new Array();
    var initiators =
System.getModule("com.dell.storage.vmax.host").getAvailableInitiators(vmaxarray);
    // Check if there are any Unisphere hosts for this ESXi host
    for each ( var wwn in wwns )
```

```
{
  //System.log("Checking if wwn " + wwn + " already belongs to any initiator group ");
  var vhost = getHostForWWN(wwn, vmaxarray);
  if ( !vhost )
  {
    System.log("wwn " + wwn + " does not belong to any initiator group ");
    for each ( var initiator in initiators )
    {
      if ( wwn.equals(initiator.wwn) )
      {
        initlist.push(initiator);
        break;
      }
    }
  }
  else
  {
    var found = false;
    if ( hostlist.length > 0 )
    for each ( var tmphost in hostlist )
    {
      if ( vhost.hostName.equals(tmphost.hostName) )
      {
        System.log("host already present in the host group");
        found = true;
        break;
      }
    }
    if ( found === false )
    {
      System.log("Adding " + vhost.hostName + " host group");
      hostlist.push(vhost);
    }
  }
}

if ( initlist.length > 0 )
{
  var hostname = "VR0_HOST_" + host.name.substring(0, host.name.indexOf('.'));
  System.log("Creating host with name " + hostname);
  System.getModule("com.dell.storage.vmax.host").createHost(vmaxarray,hostname,initlist,false,
  e,false,false,false,false,false,false,false,false,false);
  var vhost = getHostForWWN(initlist[0].wwn, vmaxarray);
  hostlist.push(vhost);
}
```

```
}
if ( hostlist.length > 0 )
{
    System.log("Creating host group " + name + " with " + hostlist.length + " hosts");
    System.getModule("com.dell.storage.vmax.host").createHostGroup(vmaxarray,name,hostlist,false,false,false,false,false,false,false,false,false,false,false);
}

function getHostForWWN(wwn, vmax)
{
    var hostlist = System.getModule("com.dell.storage.vmax.host").getAllHosts(vmax);
    System.log("Found [" + hostlist.length + "] hosts in vmax [" + vmax.displayName + "]");

    for ( var i=0; i < hostlist.length; i++ )
    {
        var host = hostlist[i];
        System.log("Checking if wwn " + wwn + " is in initiator group " + host.hostName);
        var wwns = System.getModule("com.dell.storage.vmax.host").getInitiatorsInHost(host);
        System.log("There are [" + wwns.length + "] wwns in host [" + host.hostName + "]");
        for each ( var vwwn in wwns )
        {
            if ( vwwn.wwn.equals(wwn) )
            {
                System.log("Found " + wwn + " in initiator group " + host.hostName);
                return host;
            }
        }
    }
    System.error("Failed to find host for wwn [" + wwn + "]");
    return null;
}

function getWWNs(hostname)
{
    if ( !hostname )
        throw "host not found";

    var hostStorageSystem = VcPlugin.toManagedObject( hostname,
hostname.configManager.storageSystem );
    var wwnList = new Array();
    var i = 0;
    for each (var hba in hostStorageSystem.storageDeviceInfo.hostBusAdapter)
    {
        //System.log('Device Name: ' + hba.device);
        //System.log('Device Type: ' + hba.driver);
    }
}
```

```
    if (hba instanceof VcHostFibreChannelHba)
    {
        //System.log('FC HBA: ' + hba.portWorldWideNameHex);
        //wnList[i++] = hba.portWorldWideNameHex;
        var wwn = hba.portWorldWideNameHex;
        wwnList.push(wwn.replace(/:/g, ''));
    }
}

return wwnList;
}
```

## 1.5.5. Delete Snapshot Schedule for VMFS Datastore

Remove protection set for a VMFS datastore hosted on VMAX storage

### 1.5.5.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.5.5.2. Inputs

Name	Type	Description
schedule	VMAXPlugin:Snapshot Schedule	Select Snapshot Schedule

### 1.5.5.3. Parameter presentation

Parameter	Description	Parameter Info
schedule	Select Snapshot Schedule	

### 1.5.5.4. Workflow schema



### 1.5.5.5. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
Scriptable task	WorkflowItemTask		

### 1.5.5.6. Source code for the used actions

#### 1.5.5.6.1. Scriptable task

```
VMAXPluginSnapshotScheduleManager.remove(schedule.snapshotScheduleInfo);
```

### 1.5.6. Mount Snapshot of VMFS datastore on target ESX host

Mounts a VMAX Storage Snapshot on a target ESX host

#### 1.5.6.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.5.6.2. Inputs

Name	Type	Description
vcDatastore*	VC:Datastore	The VMware datastore



Name	Type	Description
snapshotToMount*	VMAXPlugin:Snapvxsnapshot	The Snapshot to mount
vmaxDatastore	VMAXPlugin:DataStore	
targetEsxHost*	Array/VC:HostSystem	The Target ESXi hosts on which the snapshot will be mounted

\*Parameter is mandatory

### 1.5.6.3. Attributes

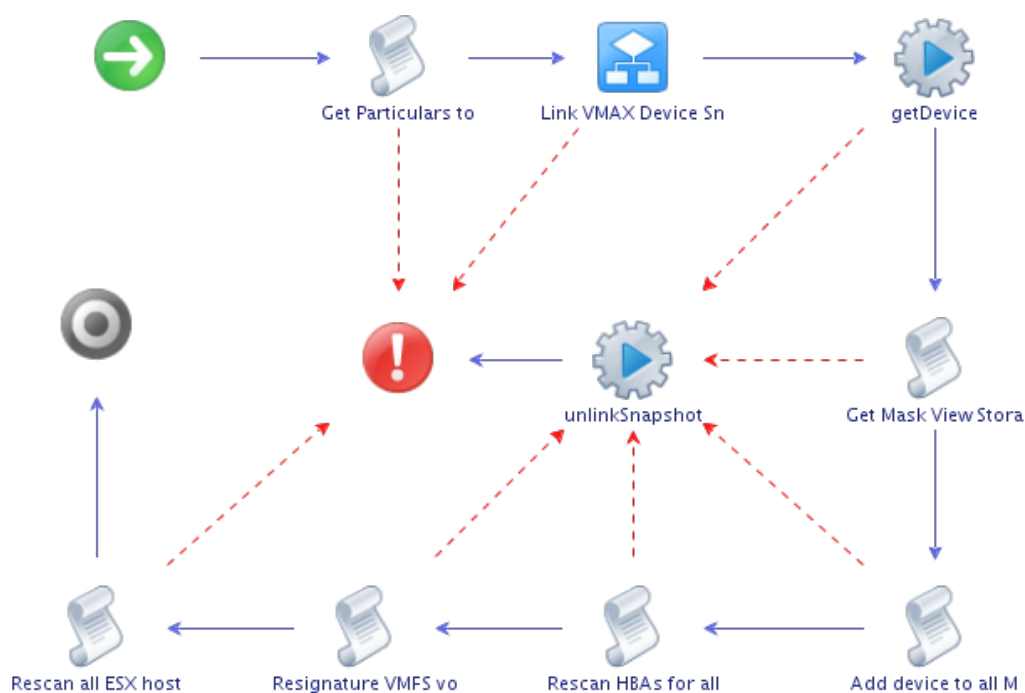
Name	Type	Description
device	VMAXPlugin:VmaxDevice	
snapshotName	string	
vmaxArray	VMAXPlugin:VMAXArray	
errorCode	string	
storageGroupList	Array/VMAXPlugin:StorageGroup	
linkedDeviceWwn	string	
linkedTargetDevice	VMAXPlugin:VmaxDevice	
deleteTarget	boolean	

### 1.5.6.4. Parameter presentation

Parameter	Description	Parameter Info
vcDatastore	The VMware datastore	Mandatory input: true
snapshotToMount	The Snapshot to mount	Mandatory input: true

Parameter	Description	Parameter Info
vmaxDatastore	vmaxDatastore	Data binding: GetAction("com.dell.storage.vmax.vmware","getVmaxDatastoreFromVcDatastore").call( #vcDatastore )
targetEsxHost	The Target ESXi host on which the snapshot will be mounted	Mandatory input: true

### 1.5.6.5. Workflow schema



### 1.5.6.6. Workflow items

Name	Type	Next item	Description
Get Particulars to link snapshot	WorkflowItemTask	Link VMAX Device Snapshot	

Name	Type	Next item	Description
Link VMAX Device Snapshot	WorkflowItemLink	getDevice	Link a VMAX Device Snapshot to a target
Get Mask View Storage Group for Host	WorkflowItemTask	Add device to all Mask View Storage Groups	
	WorkflowItemEnd		
getDevice	WorkflowItemTask	Get Mask View Storage Group for Host	
Resignature VMFS volume	WorkflowItemTask	Rescan all ESX hosts for new VMFS volumes	
	WorkflowItemEnd		
unlinkSnapshot	WorkflowItemTask		
Add device to all Mask View Storage Groups	WorkflowItemTask	Rescan HBAs for all hosts	
Rescan HBAs for all hosts	WorkflowItemTask	Resignature VMFS volume	
Rescan all ESX hosts for new VMFS volumes	WorkflowItemTask		

### 1.5.6.7. Source code for the used actions

#### 1.5.6.7.1. Get Particulars to link snapshot

```
device =
System.getModule("com.dell.storage.vmax.provisioning").getDevice(vmaxDatastore.uuid);
System.log("Found device " + device.displayName + " for datastore " +
vmaxDatastore.displayName);
snapshotName = snapshotToMount.displayName;

var vmaxArrays = System.getModule("com.dell.storage.vmax.array").getVmaxArrays();
for each ( var vmax in vmaxArrays )
{
    if ( vmaxDatastore.arrayId.equals(vmax.displayName) )
    {
        vmaxArray = vmax;
        break;
    }
    if ( vmaxArray )
        break;
}

if ( !vmaxArray )
    throw "Failed to find VMAX array for datastore " + vmaxDatastore.displayName;
```

### 1.5.6.7.2. Get Mask View Storage Group for Host

```
System.log("Finding masking views for " + esxHostList.length + " hosts");
storageGroupList = [];
for each ( var esxHost in esxHostList )
{
    var mvList =
System.getModule("com.dell.storage.vmax.vmware").getMaskViewsForEsx(vmaxArray, esxHost);
    if ( !mvList || mvList.length <= 0 )
    {
        System.warn("The host " + esxHost.name + " does not have any mask views for the
VMAX");
    }
    else
    {
        System.log("Found " + mvList.length + " mask views for target ESX " + esxHost.name +
", picking the first one");
        storageGroupList.push(mvList[0].storageGroup);
        System.log("Found storage group: " + mvList[0].storageGroup.displayName);
    }
}
```

### 1.5.6.7.3. getDevice

```
//Auto generated script, cannot be modified !
actionResult = System.getModule("com.dell.storage.vmax.provisioning").getDevice(deviceWwn)
;
```

### 1.5.6.7.4. Resignature VMFS volume

```
//var hostDatastoreSystem = null;
//var targetHostSystem = null;
var bResignatureDone = false;
for each ( var targetHostSystem in targetHostSystemList )
{
    var hostDatastoreSystem = targetHostSystem.configManager.datastoreSystem;

    var unresolvedVolumes = hostDatastoreSystem.queryUnresolvedVmfsVolumes();
    if ( !unresolvedVolumes || unresolvedVolumes.length <= 0 )
    {
        System.warn("No unresolved volumes, failed to see the snapshot on the target ESX host
" + targetHostSystem.name);
        continue;
    }

    System.log("Found " + unresolvedVolumes.length + " unresolved volumes, looking for
volume " + linkedDeviceWwn);

    var devicePath = null;
    for each ( var unresolvedVolume in unresolvedVolumes )
    {
        var extents = unresolvedVolume.extents;
        if ( ! extents || extents.length != 1 )
        {
            System.warn("Skipping volume " + unresolvedVolume + " since it has " +
extents.length + " extents");
            continue;
        }
        var diskName = extents[0].device.diskName;
        diskName = diskName.replace("naa.", "");
        if ( diskName.equals(linkedDeviceWwn) )
        {
            System.log("Found matching disk name!");
            devicePath = extents[0].devicePath;
            break;
        }
    }
}
```

```
    }
}

if ( !devicePath )
{
    System.warn("No suitable volume found to resolve for WWN " + linkedDeviceWwn + " on
host " + targetHostSystem.name);
    continue;
}

System.log("Found extent device path: " + devicePath);
bResignatureDone = true;

var extentDevicePaths = [];
extentDevicePaths.push(devicePath);
var resignatureSpec = new VcHostUnresolvedVmfsResignatureSpec() ;
resignatureSpec.extentDevicePath = extentDevicePaths;

var task = hostDatastoreSystem.resignatureUnresolvedVmfsVolume_Task(resignatureSpec);

System.log("Successfully launched task " + task.id + " for resignature of DS");
while ( task.info.state === VcTaskInfoState.queued ||
        task.info.state === VcTaskInfoState.running )
{
    System.log("Task " + task.info.name + " state: " + task.info.state + ", Progress: "
+ task.info.progress);
    System.sleep(5000);
}

System.log("Task " + task.info.name + " completed with status " + task.info.state);
if ( task.info.state !== VcTaskInfoState.success )
    throw "Task " + task.info.name + " failed: " + task.info.error.localizedMessage;
break;
}

if (!bResignatureDone)
    throw "The Volume " + linkedDeviceWwn + " was not found on any of the target ESX hosts";
```

### 1.5.6.7.5. unlinkSnapshot

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.snapshot").unlinkSnapshot(vmaxArray,snapshot,srcDe
vice,tgtDevice,deleteTarget) ;
```

#### 1.5.6.7.6. Add device to all Mask View Storage Groups

```
if ( !storageGroupList || storageGroupList.length <= 0 )
    throw "None of the ESX servers have a masking view for the selected VMAX";
for each ( var storageGroup in storageGroupList )
{
    try
    {
        System.getModule("com.dell.storage.vmax.provisioning").addDeviceToStorageGroup(linkedTargetDevice,storageGroup);
    }
    catch(e)
    {
        System.warn("Failed to add device " + linkedTargetDevice.displayName
            + " to storage group " + storageGroup.displayName + ": " + e.message);
    }
}
```

#### 1.5.6.7.7. Rescan HBAs for all hosts

```
for each ( var esxHost in targetEsxHost )
{
    System.getModule("com.vmware.library.vc.host").rescanHostHBAs(esxHost) ;
}
```

#### 1.5.6.7.8. Rescan all ESX hosts for new VMFS volumes

```
for each ( var host in targetEsxHost )
{
    System.getModule("com.vmware.library.vc.storage").rescanAll(host) ;
}

System.log("Refreshing Datastore List in inventory");
vmaxArray.getStorageVolumesFolder().getDataStoresFolder().datastoreAdded(linkedDeviceWwn);
```

### 1.5.7. Provision VMFS datastore to ESXi Server using new VMAX storage

Create necessary pre-requisites, provision a new VMAX device, present it to the ESX and create a VMFS datastore on this device

#### 1.5.7.1. Versions

Version	Date	User	Comment

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.5.7.2. Inputs

Name	Type	Description
hostname*	VC:HostSystem	Enter ESX server ip address or DNS name
vmaxArray*	VMAXPlugin:VMAXArray	Provide VMAX array
vmfsVersion	number	The VMFS major version
datastoreName*	string	The datastore name
capacity*	number	The capacity of the selected disk

\*Parameter is mandatory

### 1.5.7.3. Outputs

Name	Type	Description
datastore	VC:Datastore	The new datastore

### 1.5.7.4. Attributes

Name	Type	Description
wwnList	string	
storageGroup	string	
fcDisk	string	
errorCode	string	

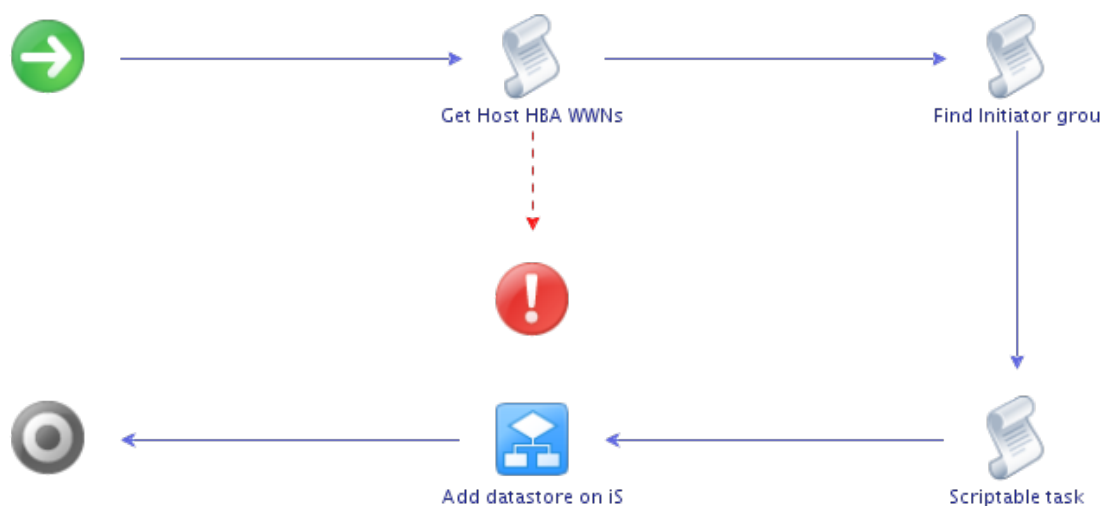
### 1.5.7.5. Parameter presentation

Parameter	Description	Parameter Info



Parameter	Description	Parameter Info
vmaxArray	Select VMAX array	Mandatory input: true
hostname	Select ESXi Server	Mandatory input: true
datastoreName	Enter Datastore name	Mandatory input: true
capacity	Enter Datastore size in GB	Mandatory input: true
vmfsVersion	Select VMFS major version	

### 1.5.7.6. Workflow schema



### 1.5.7.7. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
Get Host HBA WWNs	WorkflowItemTask	Find Initiator group	
Find Initiator group	WorkflowItemTask	Scriptable task	

Name	Type	Next item	Description
Scriptable task	WorkflowItemTask	Add datastore on iSCSI/FC/local SCSI	
Add datastore on iSCSI/FC/local SCSI	WorkflowItemLink		Creates a datastore on a Fibre Channel, iSCSI or local SCSI disk. Only disks that are not currently in use by an existing VMFS are applicable to new datastore creation. The new datastore allocates the maximum available space of the specified disk.
	WorkflowItemEnd		

## 1.5.7.8. Source code for the used actions

### 1.5.7.8.1. Get Host HBA WWNs

```
//-----

if ( !hostname )
    throw "host not found";

var hostStorageSystem = VcPlugin.toManagedObject( hostname,
hostname.configManager.storageSystem );
var wwnList = new Array();
var i = 0;
for each (var hba in hostStorageSystem.storageDeviceInfo.hostBusAdapter)
{
    System.log('Device Name: ' + hba.device);
    System.log('Device Type: ' + hba.driver);
    if (hba instanceof VcHostFibreChannelHba)
    {
        System.log('FC HBA: ' + hba.portWorldWideNameHex);
        wwnList[i++] = hba.portWorldWideNameHex;
    }
}
```

```
/*----- function to get All WWNs of each ESXi Host -----  
*/  
  
function getWWNs(hosts){  
    var hostWWNsProps = new Properties();  
  
    for (var hostCount = 0; hostCount < hosts.length; hostCount++) {  
        System.log("Finding FC WWNs for ESXi host named " + hosts[hostCount].name);  
        var hostStorageSystem = VcPlugin.toManagedObject( hosts[hostCount],  
hosts[hostCount].configManager.storageSystem );  
        var hostBusAdapters = hostStorageSystem.storageDeviceInfo.hostBusAdapter;  
        var wwns = [];  
        if ( hostBusAdapters != null && hostBusAdapters.length > 0 ) {  
            System.log( "Found " + hostBusAdapters.length + " storage adapters." );  
            for(var hbaCount=0; hbaCount < hostBusAdapters.length; hbaCount++ ) {  
                var wwn = hostBusAdapters[hbaCount].portWorldWideNameHex;  
                if(wwn){  
                    System.log( "Device: " + hostBusAdapters[hbaCount].device + " Model: " +  
hostBusAdapters[hbaCount].model);  
                    System.log( "WWN: " + wwn);  
                    wwns.push(wwn.replace(/:/g, ''));  
                }  
            }  
        }  
        else {  
            System.log( "No storage adapters found" );  
        }  
        if(wwns.length > 0) {  
            hostWWNsProps.put(hosts[hostCount].name, wwns);  
        }  
    }  
    return hostWWNsProps;  
}
```

### 1.5.7.8.2. Find Initiator group

```
var maskview = null;
```

```
//var igList = VMAXPluginVMAXManager.getInitiatorGroupList(vmaxArray);
//System.log("Found " + igList.length + " Initiator groups ");

for each ( var wwn in wwnList )
{
    wwn = wwn.replace(/:/g, "");
    System.log("Checking if wwn " + wwn + " belongs to any initiator group");
    var mvList = vmaxArray.maskViews;
    for each ( var mv in mvList )
    {
        var ig = mv.initiatorGroup;

        System.log("Getting initiators for Initiator group " + ig.displayName + " Masking
view " + ig.maskingView);
        var wwns = ig.initList;
        if ( typeof wwns === "undefined" )
        {
            System.log("No initiators found");
            continue;
        }
        System.log("Number of initiators = " + wwns.length);

        for each ( var vwwn in wwns )
        {
            System.log("WWN = " + vwwn.displayName);
            System.log ( "Comparing (" + wwn + "," + vwwn + ")");
            if ( wwn.equals(vwwn) )
            {
                System.log("Found match in initiator group = " + ig.displayName);
                System.log("Masking view is " + ig.maskingView);
                System.log("Getting storage group ... ");
                //var mv = VMAXPluginVMAXManager.getMaskingViewByName(vmaxArray,
ig.maskingView);
                var storageGroup = mv.storageGroup;
                System.log("Storage group name = " + storageGroup.displayName);
            }
        }
    }
}
```

### 1.5.7.8.3. Scriptable task

```
System.log("Creating device for " + vmaxArray.displayName + "," + storageGroup.displayName
+ ", with size = " + size );

var device = VMAXPluginVMAXManager.createDevice(vmaxArray , storageGroup, "GB", size *
1024 , "");
System.log("Newly created device name = " + device);

var deviceName = null;
if (hostname != null) {
    var hostDatastoreSystem = VcPlugin.toManagedObject( hostname,
hostname.configManager.datastoreSystem );
    var hostStorageSystem = VcPlugin.toManagedObject(hostname,
hostname.configManager.storageSystem);
    hostStorageSystem.rescanAllHba();
    var hostScsiDisks = hostDatastoreSystem.queryAvailableDisksForVmfs(null);

    for each (var disk in hostScsiDisks){
        System.log("Disk: " + disk.displayName);
        if ( disk.displayName.indexOf(device) !== -1 )
        {
            System.log("Found disk on ESX host");
            deviceName = disk.displayName;
        }
    }
}
```

### 1.5.8. Provision VMFS datastore to vSphere cluster using new VMAX storage

Create necessary pre-requisites, provision a new VMAX device, present it to the cluster and create a VMFS datastore on this device

#### 1.5.8.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

#### 1.5.8.2. Inputs

Name	Type	Description
vmaxArray*	VMAXPlugin:VMAXArray	Provide VMAX array

Name	Type	Description
vmfsVersion	number	The VMFS major version
datastoreName*	string	The datastore name
capacity*	number	The capacity of the selected disk
cluster*	VC:ClusterComputeResource	

\*Parameter is mandatory

### 1.5.8.3. Outputs

Name	Type	Description
datastore	VC:Datastore	The new datastore

### 1.5.8.4. Attributes

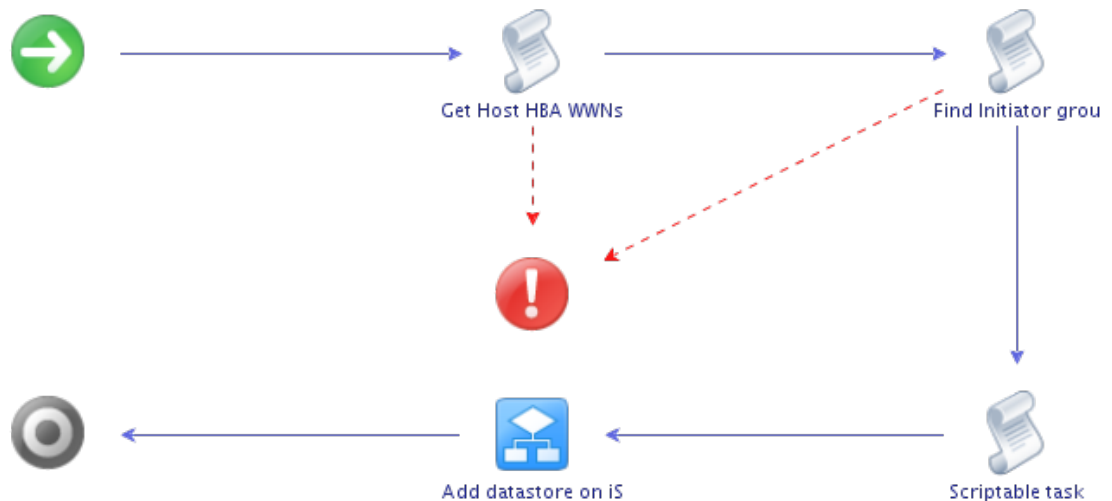
Name	Type	Description
storageGroup	string	
fcDisk	string	
errorCode	string	
host	VC:HostSystem	
hostWwnTable	Properties	

### 1.5.8.5. Parameter presentation

Parameter	Description	Parameter Info
vmaxArray	Provide VMAX array	Mandatory input: true
cluster	cluster	Mandatory input: true
datastoreName	Enter Datastore name	Mandatory input: true
capacity	Enter Datastore size in GB	Mandatory input: true

Parameter	Description	Parameter Info
vmfsVersion	Select VMFS major version	

### 1.5.8.6. Workflow schema



### 1.5.8.7. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		
Get Host HBA WWNs	WorkflowItemTask	Find Initiator group	
Find Initiator group	WorkflowItemTask	Scriptable task	
Scriptable task	WorkflowItemTask	Add datastore on iSCSI/FC/local SCSI	

Name	Type	Next item	Description
Add datastore on iSCSI/FC/local SCSI	WorkflowItemLink		Creates a datastore on a Fibre Channel, iSCSI or local SCSI disk. Only disks that are not currently in use by an existing VMFS are applicable to new datastore creation. The new datastore allocates the maximum available space of the specified disk.
	WorkflowItemEnd		

## 1.5.8.8. Source code for the used actions

### 1.5.8.8.1. Get Host HBA WWNs

```

if ( !cluster ) throw "Provider a valid cluster instance"

var hosts = cluster.host;
System.log("Found " + hosts.length + " hosts in the cluster");

// Table that holds host and their WWNs
var hostWwnTable = new Properties();

for each ( var host in hosts )
{
    var wwnList = new Array();
    var i = 0;
    System.log("Collecting WWNs for host " + host.name);
    var hostStorageSystem = VcPlugin.toManagedObject( host, host.configManager.storageSystem );
    for each (var hba in hostStorageSystem.storageDeviceInfo.hostBusAdapter)
    {
        System.log('Device Name: ' + hba.device);
        System.log('Device Type: ' + hba.driver);
        if (hba instanceof VcHostFibreChannelHba)
        {
            System.log('FC HBA: ' + hba.portWorldWideNameHex);
            wwnList[i++] = hba.portWorldWideNameHex;
        }
    }
    hostWwnTable.put(host.name, wwnList);
}

```



```
}
```

#### 1.5.8.8.2. Find Initiator group

```
var mvList = vmaxArray.maskViews;
for each ( var mv in mvList )
{
    var ig = mv.initiatorGroup;

    System.log("Getting initiators for Initiator group " + ig.displayName + " Masking view "
+ ig.maskingView);
    var wwns = ig.allWWNs;
    if ( typeof wwns === "undefined" )
    {
        System.log("No initiators found");
        continue;
    }
    System.log("Number of initiators = " + wwns.length);

    var hosts = hostWwnTable.keys;

    for each( host in hosts )
    {
        System.log("Checking wwns for host " + host + " In IG " + ig.displayName);
        var wwnList = hostWwnTable[host];
        System.log("There are " + wwnList.length + " initiators on " + host);
        var found = false;
        for each ( var wwn in wwnList )
        {
            wwn = wwn.replace(/:/g, "");
            for each ( var vwwn in wwns )
            //for ( var i=0; i < wwns.length; i++ )
            {
                //var vwwn = wwns[i];
                System.log ( "Comparing ( " + wwn + " ," + vwwn.wwn + " )");
                if ( wwn.equals(vwwn.wwn) )
                    found = true;
            }
        }
        if ( found === false )
            break;
    }
    if ( found === true )
```

```
{
    System.log("Found match in initiator group = " + ig.displayName);
    System.log("Masking view is " + ig.maskingView);
    System.log("Getting storage group ... ");
    var storageGroup = mv.storageGroup;
    System.log("Storage group name = " + storageGroup.displayName);
    break;
}
}
```

```
if ( !storageGroup )
    throw "Host group not found for the cluster";
```

### 1.5.8.8.3. Scriptable task

```
System.log("Creating device for " + vmaxArray.displayName + "," + storageGroup.displayName
+ ", with size = " + size );

var device = VMAXPluginVMAXManager.createDevice(vmaxArray , storageGroup, "GB", size, "");

System.log("Newly created device name = " + device);

var hosts = cluster.host;

if ( !hosts )
    throw "Hosts not found in the cluster";

for (var esxiCount = 0; esxiCount < hosts.length; esxiCount++) {
    System.log(hosts[esxiCount].name);
    var hostname = hosts[esxiCount];
    var deviceName = null;
    if (hostname != null) {
        var hostDatastoreSystem = VcPlugin.toManagedObject( hostname,
hostname.configManager.datastoreSystem );
        var hostStorageSystem = VcPlugin.toManagedObject(hostname,
hostname.configManager.storageSystem);
        hostStorageSystem.rescanAllHba();
        var hostScsiDisks = hostDatastoreSystem.queryAvailableDisksForVmfs(null);

        for each (var disk in hostScsiDisks){
            System.log("Disk: " + disk.displayName);
            if ( disk.displayName.indexOf(device) !== -1 )
            {
                System.log("Found disk on ESX host");
            }
        }
    }
}
```

```

        deviceName = disk.displayName;
    }
}
}
}
}

```

```
var host = hosts[0];
```

## 1.5.9. Unmount Snapshot of VMFS datastore on target ESX host/hosts

Unmounts the datastore that is provisioned from a datastore snapshot

### 1.5.9.1. Versions

Version	Date	User	Comment
1.0.0	Jan 23, 2018	admin	Initial Version

Current version:1.0.0

### 1.5.9.2. Inputs

Name	Type	Description
datastoreToUnmount*	VC:Datastore	vCenter DataStore that needs to be unmounted(provisioned from vmax)
esxHosts*	Array/VC:HostSystem	esx Hosts that Datastore needs to be unmounted from

\*Parameter is mandatory

### 1.5.9.3. Attributes

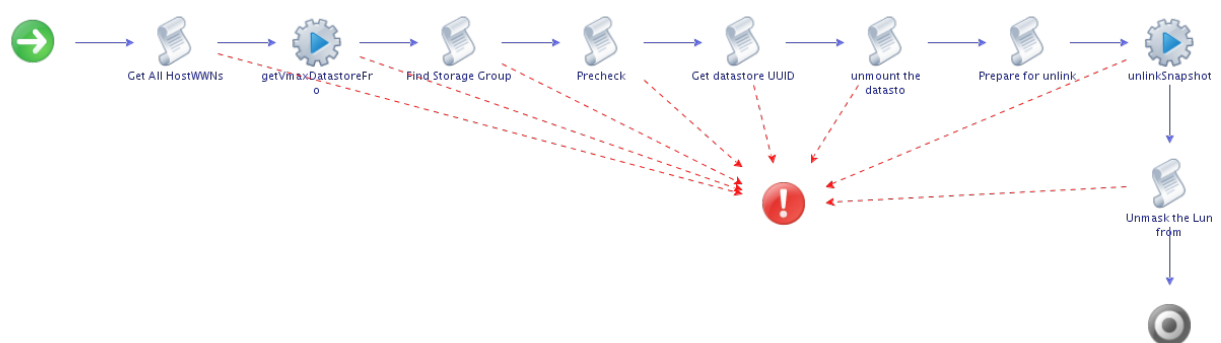
Name	Type	Description
errorCode	string	
dsUUID	string	
hostWwnTable	Properties	
vmaxDevice	VMAXVR0Plugin:VmaxDevice	
targetVmaxDevice	VMAXVR0Plugin:VmaxDevice	

Name	Type	Description
srcSnapshotName	string	
deleteTargetDevice	boolean	
snapStorageGroup	string	
hostToStorageGroupTable	Properties	
vmaxDatastore	VMAXVR0Plugin:DataStore	
vmaxArray	VMAXVR0Plugin:VMAXArray	
diskUuid	string	

#### 1.5.9.4. Parameter presentation

Parameter	Description	Parameter Info
datastoreToUnmount	datastoreToUnmount	Mandatory input: true
esxHosts	esxHost	Mandatory input: true

#### 1.5.9.5. Workflow schema



#### 1.5.9.6. Workflow items

Name	Type	Next item	Description
	WorkflowItemEnd		

Name	Type	Next item	Description
Get datastore UUID	WorkflowItemTask	unmount the datastore	
	WorkflowItemEnd		
unmount the datastore	WorkflowItemTask	Prepare for unlink target	
Precheck	WorkflowItemTask	Get datastore UUID	
Get All HostWWNs	WorkflowItemTask	getVmaxDatastoreFromVcDatastore	
Find Storage Group	WorkflowItemTask	Precheck	
unlinkSnapshot	WorkflowItemTask	Unmask the Lun from hosts	
Prepare for unlink target	WorkflowItemTask	unlinkSnapshot	
Unmask the Lun from hosts	WorkflowItemTask		
getVmaxDatastoreFromVcDatastore	WorkflowItemTask	Find Storage Group	

## 1.5.9.7. Source code for the used actions

### 1.5.9.7.1. Get datastore UUID

```

if ( !(vcDatastore.summary.type === ("VMFS")) )
{
    System.error("Only VMFS datastores are supported");
}

```

```
    throw "Datastore type [" + vcDatastore.summary.type + "] not supported";
}
```

```
var vmfsInfo = vcDatastore.info.vmfs;
var dsUUID = vmfsInfo.uuid;
```

```
System.log("UUID of the data " + dsUUID);
```

### 1.5.9.7.2. unmount the datastore

```
for each (var hostname in esxHost)
{
    System.log('HostName:: ' + hostname.name);
    if(hostname != null)
    {
        for each (var host in vcDatastore.host)
        {
            var checkHostName = host.key.name;
            System.log('DS is mounted on host:: '+checkHostName);
            if( checkHostName == hostname.name)
            {
                System.log('Input hostname is matching with Datastore that is mounted on:: ' +
hostname.name);
                var hostStorageSystem = VcPlugin.toManagedObject(hostname,
hostname.configManager.storageSystem);

                hostStorageSystem.unmountVmfsVolume(dsUUID);
                hostStorageSystem.rescanVmfs();
            }
            else
            {
                System.log('Input hostname is not matching with Datastore that is mounted on::
' + hostname.name);
            }
        }
    }
}

System.log("Refreshing Datastore List in inventory");
vmaxArray.getStorageVolumesFolder().getDataStoresFolder().datastoreDeleted(vcDatastore.info.name);
```

### 1.5.9.7.3. Precheck

```
//Check 1# VMFS DS check
if ( !(vcDatastore.summary.type === ("VMFS")) )
{
    System.error("Only VMFS datastores are supported");
    throw "Datastore type [" + vcDatastore.summary.type + "] not supported";
}

var vmfsInfo = vcDatastore.info.vmfs;
if ( vmfsInfo.extent === undefined)
{
    throw "Failed to get extents for VMFS " + vmfsInfo.name;
}
var extents = vmfsInfo.extent;
if ( extents.length != 1 )
{
    System.error("Datastore " + vcDatastore.name + " is created from [" + extents.length +
"] extents");
    throw "Resizing VMFS datastores created from more than one extent is not supported";
}

// check #2 Backing LUN for DS is from VMAX
diskUuid = new String(extents[0].diskName);
System.log("Found VMFS backing volume exact WWN: " + diskUuid);
diskUuid = diskUuid.replace("naa.", "");
System.log("Found VMFS backing volume WWN: " + diskUuid);

var snapStorageGroup = "vroSG_";
snapStorageGroup = snapStorageGroup.concat(diskUuid);
System.log("storage group of lun:." + diskUuid + " is --> " + snapStorageGroup);

var flag = false;
var hosts = hostToStorageGroupTable.keys;

for each (var host in hosts)
{
    var vmaxDevice =
vmaxArray.getVmaxDeviceByWWN(hostToStorageGroupTable[host].name,diskUuid) ;
```

```
if( vmaxDevice == null )
{
    System.log("DataStore not backed by VMAX array "+vmaxArray.displayName);

}
else
{
    flag = true;
    System.log("VMAX deviced fetched from VMAX Array " + vmaxDevice.displayName);
}
}

if(flag == false)
    throw "DataStore not provisioned with VMAX Array";

System.log("Checking if Datastore is provisioned by snapvx target device...");
var isLinkedTarget =    vmaxArray.isVmaxDeviceLinkedTarget(vmaxDevice);

if(isLinkedTarget == false)
{
    System.log("Target Device is not linked from snapVX snapshot");
    throw "Target Device is not linked from snapVX snapshot";
}
```

### 1.5.9.7.4. Get All HostWWNs

```
var hostWwnTable = new Properties();

for each (var hostname in esxHosts)
{
    if ( !hostname )
        throw "host not found";

    var hostStorageSystem = VcPlugin.toManagedObject( hostname,
hostname.configManager.storageSystem );
    var wwnList = new Array();
    var i = 0;
    for each (var hba in hostStorageSystem.storageDeviceInfo.hostBusAdapter)
    {
        System.log('Device Name: ' + hba.device);
        System.log('Device Type: ' + hba.driver);
        if (hba instanceof VcHostFibreChannelHba)
```



```
        {
            System.log('FC HBA: ' + hba.portWorldWideNameHex);
            wwnList[i++] = hba.portWorldWideNameHex;
        }
    }
    hostWwnTable.put(hostname.name, wwnList);
}
```

### 1.5.9.7.5. Find Storage Group

```
if(vmaxDatastore == null)
    throw "vmax Datastore is empty... please check";

// #1. Fetch Vmax array that Datastore is mounted on
var vmaxArray = null;
var vmaxArrayID = vmaxDatastore.getArrayId();
var arrays = System.getModule("com.dell.storage.vmax.array").getVmaxArrays();

System.log("Datastore belongs to vmax::"+ vmaxArrayID);

for each (var array in arrays)
{
    if(array.getArrayId() === vmaxArrayID)
    {
        System.log("Datastore:: " + vmaxDatastore.displayName + " belongs to vmaxArray:: " +
array.getArrayId());
        vmaxArray = array;
        break;
    }
}

if(vmaxArray == null)
{
    System.log("Datastore provided in input::" + vmaxDatastore.displayName + " is not
Provisioned from Vmax Array");
    throw " Datastore provided in input is not Provisioned from Vmax Array";
}
//=====
// #2. Fetch Storage group to which device belongs
```

```
var hostToStorageGroupTable = new Properties();

var mvList = vmaxArray.maskViews;

for each ( var mv in mvList )
{
    var ig = mv.initiatorGroup;

    System.log("Getting initiators for Initiator group " + ig.displayName + " Masking view "
+ ig.maskingView);
    var wwns = ig.allWWNs;
    if ( typeof wwns === "undefined" )
    {
        System.log("No initiators found");
        continue;
    }
    System.log("Number of initiators = " + wwns.length);

    var hosts = hostWwnTable.keys;

    System.log("Checking against num of hosts:: " + hosts.length);

    for each( host in hosts )
    {
        System.log("Checking wwns for host " + host + " In IG " + ig.displayName);
        var wwnList = hostWwnTable[host];
        System.log("There are " + wwnList.length + " initiators on " + host);
        var found = false;
        for each ( var wwn in wwnList )
        {
            wwn = wwn.replace(/:/g, "");
            for each ( var vwwn in wwns )
            //for ( var i=0; i < wwns.length; i++ )
            {
                //var vwwn = wwns[i];
                System.log ("Comparing (" + wwn + "," + vwwn.wwn + ")");
                if ( wwn.equals(vwwn.wwn) )
                {
                    System.log("Setting flag to TRUE");
                    found = true;
                }
            }
        }
        if ( found === true )
```

```
{
    System.log("Found match in initiator group = " + ig.displayName);
    System.log("Masking view is " + ig.maskingView);
    System.log("Getting storage group ... ");
    var storageGroup = mv.storageGroup;
    System.log("Storage group name = " + storageGroup.displayName)
    hostToStorageGroupTable.put(host,storageGroup);
    //break;
}
}
```

```
if ( hostToStorageGroupTable.size == 0 )
    throw "Masking view not found for provided ESX hosts";
```

### 1.5.9.7.6. unlinkSnapshot

```
//Auto generated script, cannot be modified !
System.getModule("com.dell.storage.vmax.snapshot").unlinkSnapshot(vmaxArray,snapshot,srcDevice,tgtDevice,deleteTarget) ;
```

### 1.5.9.7.7. Prepare for unlink target

```
var snapName = vmaxArray.getSourceSnapshotDetailsForLinkedDevice(tgtVmaxDevice);
```

```
if(snapName == null)
    throw "snapshot associated with target device linking is empty";
```

```
System.log('snapshot name of linked target is::'+snapName);
```

```
var srcVmaxDevice = vmaxArray.getVmaxDeviceFromSnapshot(snapName);
```

```
if ( srcVmaxDevice == null )
{
    throw "source device not found";
}
```

```
System.log('source device is:: ' + srcVmaxDevice.displayName);
```

```
System.log('Setting delete target device as false');
var deleteTargetDevice = false;
```

### 1.5.9.7.8. Unmask the Lun from hosts

```
System.log("Unpresenting Lun from hosts...");

var hosts = hostToStorageGroupTable.keys;
System.log("Unmasking needs to be done to #" + hosts.length + " hosts");

for each (var host in hosts)
{
    var sg = hostToStorageGroupTable[host];
    System.log("Removing " + targetVmaxDevice.displayName + " fom SG:: "+ sg.name);
    var delDeviceNum = vmaxArray.removeDeviceFromSG(sg.name,targetVmaxDevice);
    System.log("Device::"+delDeviceNum +" sucessfully unmasked from esxHost::"+host);
}
```

### 1.5.9.7.9. getVmaxDatastoreFromVcDatastore

```
//Auto generated script, cannot be modified !
actionResult =
System.getModule("com.dell.storage.vmax.vmware").getVmaxDatastoreFromVcDatastore(vcDatastore) ;
```