

Foglight™ for Microsoft Hyper-V  
ActionPack 7.3.0

## **User and Reference Guide**



© 2024 Quest Software Inc.

**ALL RIGHTS RESERVED.**

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software Inc.

The information in this document is provided in connection with Quest Software products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest Software does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.  
Attn: LEGAL Dept.  
4 Polaris Way  
Aliso Viejo, CA 92656

Refer to our website (<https://www.quest.com>) for regional and international office information.

**Patents**

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at <https://www.quest.com/legal>.

**Trademarks**

Quest, the Quest logo, and Join the Innovation are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit <https://www.quest.com/legal/trademark-information.aspx>. "Apache HTTP Server", Apache, "Apache Tomcat" and "Tomcat" are trademarks of the Apache Software Foundation. Google is a registered trademark of Google Inc. Android, Chrome, Google Play, and Nexus are trademarks of Google Inc. Red Hat, JBoss, the JBoss logo, and Red Hat Enterprise Linux are registered trademarks of Red Hat, Inc. in the U.S. and other countries. CentOS is a trademark of Red Hat, Inc. in the U.S. and other countries. Fedora and the Infinity design logo are trademarks of Red Hat, Inc. Microsoft, .NET, Active Directory, Internet Explorer, Hyper-V, Office 365, SharePoint, Silverlight, SQL Server, Visual Basic, Windows, Windows Vista and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. AIX, IBM, PowerPC, PowerVM, and WebSphere are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Java, Oracle, Oracle Solaris, PeopleSoft, Siebel, Sun, WebLogic, and ZFS are trademarks or registered trademarks of Oracle and/or its affiliates in the United States and other countries. SPARC is a registered trademark of SPARC International, Inc. in the United States and other countries. Products bearing the SPARC trademarks are based on an architecture developed by Oracle Corporation. OpenLDAP is a registered trademark of the OpenLDAP Foundation. HP is a registered trademark that belongs to Hewlett-Packard Development Company, L.P. Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. MySQL is a registered trademark of MySQL AB in the United States, the European Union and other countries. Novell and eDirectory are registered trademarks of Novell, Inc., in the United States and other countries. VMware, ESX, ESXi, vSphere, vCenter, vMotion, and vCloud Director are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions. Sybase is a registered trademark of Sybase, Inc. The X Window System and UNIX are registered trademarks of The Open Group. Mozilla and Firefox are registered trademarks of the Mozilla Foundation. "Eclipse", "Eclipse Foundation Member", "EclipseCon", "Eclipse Summit", "Built on Eclipse", "Eclipse Ready" "Eclipse Incubation", and "Eclipse Proposals" are trademarks of Eclipse Foundation, Inc. IOS is a registered trademark or trademark of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries. Apple, iPad, iPhone, Mac OS, Safari, Swift, and Xcode are trademarks of Apple Inc., registered in the U.S. and other countries. Ubuntu is a registered trademark of Canonical Ltd. Symantec and Veritas are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries. OpenSUSE, SUSE, and YAST are registered trademarks of SUSE LCC in the United States and other countries. Citrix, AppFlow, NetScaler, XenApp, and XenDesktop are trademarks of Citrix Systems, Inc. and/or one or more of its subsidiaries, and may be registered in the United States Patent and Trademark Office and in other countries. AlertSite and DéjàClick are either trademarks or registered trademarks of Boca Internet Technologies, Inc. Samsung, Galaxy S, and Galaxy Note are registered trademarks of Samsung Electronics America, Inc. and/or its related entities. MOTOROLA is a registered trademark of Motorola Trademark Holdings, LLC. The Trademark BlackBerry Bold is owned by Research In Motion Limited and is registered in the United States and may be pending or registered in other countries. Quest is not endorsed, sponsored, affiliated with or otherwise authorized by Research In Motion Limited. Ixia and the Ixia four-petal logo are registered trademarks or trademarks of Ixia. Opera, Opera Mini, and the O logo are trademarks of Opera Software ASA. Tevron, the Tevron logo, and CitraTest are registered trademarks of Tevron, LLC. PostgreSQL is a registered trademark of the PostgreSQL Global Development Group. MariaDB is a trademark or registered trademark of MariaDB Corporation Ab in the European Union and United States of America and/or other countries. Vormetric is a registered trademark of Vormetric, Inc. Intel, Itanium, Pentium, and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries. Debian is a registered trademark of Software in the Public Interest, Inc. OpenStack is a trademark of the OpenStack Foundation. Amazon Web Services, the "Powered by Amazon Web Services" logo, and "Amazon RDS" are trademarks of Amazon.com, Inc. or its affiliates in the United States and/or other countries. Infobright, Infobright Community Edition and Infobright Enterprise Edition are trademarks of Infobright Inc. POLYCOM®, RealPresence® Collaboration Server, and RMX® are registered trademarks of Polycom, Inc. All other trademarks and registered trademarks are property of their respective

owners.

## Legend

- **WARNING:** A WARNING icon indicates a potential for property damage, personal injury, or death.
  
- ! **CAUTION:** A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.
  
- ! **IMPORTANT NOTE, NOTE, TIP, MOBILE, or VIDEO:** An information icon indicates supporting information.

Foglight for Microsoft Hyper-V ActionPack User and Reference Guide  
Foglight Version - 7.3.0  
Cartridge Version - 7.3.0

# Contents

<b>Using the ActionPack for Microsoft Hyper-V</b> .....	<b>6</b>
ActionPack for Microsoft Hyper-V Requirements .....	6
Hyper-V ActionPack Support Matrix .....	6
Hyper-V Agent .....	6
PowerShell Requirements .....	6
Services Requirements .....	7
Local Security Settings .....	7
Firewall Settings .....	8
COM and Automation Objects .....	9
Workflows .....	11
Actions .....	11
Add Network Adapter .....	11
Apply Snapshot .....	12
Attach Disk .....	12
Clone VM .....	13
Configure Network Adapter .....	13
Create Differencing VHD .....	14
Create Dynamic VHD .....	14
Create Fixed Size VHD .....	14
Create Snapshot .....	15
Create VM .....	15
Delete VM .....	16
Edit Drive .....	16
Edit VM .....	16
Move VM .....	17
Pause VM .....	18
Reboot Hyper-V Server .....	18
Remove All Network Adapters .....	18
Remove All Snapshots .....	18
Remove Drive .....	19
Remove Network Adapter .....	19
Remove Snapshot .....	19
Rename Snapshot .....	19
Resume VM .....	19
Revert to Snapshot .....	20
Save VM .....	20
Set MAC Address .....	20
Set Network .....	20
Shutdown Hyper-V Server .....	21
Shutdown VM .....	21
Start VM .....	21
Stop VM .....	21
Wait VM State .....	22

<b>About Us</b> .....	<b>23</b>
We are more than just a name .....	23
Our brand, our vision. Together. ....	23
Contacting Quest .....	23
Technical support resources .....	23

# Using the ActionPack for Microsoft Hyper-V

This *Foglight for Microsoft Hyper-V™ ActionPack User and Reference Guide* provides requirements, configuration instructions, conceptual information, and instructions on how to use the Foglight for Microsoft Hyper-V™ ActionPack to connect Foglight to Hyper-V hosts and manage virtual machines on those hosts.

This guide is intended for any user who wants to manage Hyper-V virtual machines using Foglight.

This chapter provides information about requirements that need to be met for the ActionPack for Microsoft Hyper-V to work properly, and about the actions included in the ActionPack.

## ActionPack for Microsoft Hyper-V Requirements

### Hyper-V ActionPack Support Matrix

Table 1. Support Matrix

ActionPack	ActionPack Version	Requires vFoglight Version	Supported Target Systems
Microsoft Hyper-V	5.6.2	6.6	Microsoft Windows Server 2008 R2

You must install and enable the vFoglight Cartridge for Hyper-V 5.6.2 before you can enable the ActionPack for Microsoft Hyper-V 5.6.2.

### Hyper-V Agent

You must install the Hyper-V agent to use this ActionPack. For more information, see the *Installing the Hyper-V Management Capabilities* guide.

### PowerShell Requirements

- 1 Install Microsoft .NET 2.0 (if necessary).
- 2 Install PowerShell v2 (if necessary).
- 3 Make sure *powershell.exe* is on the user's PATH.
- 4 Configure PowerShell to enable script files execution. For example:

```
powershell -command "& {Set-ExecutionPolicy remotesigned}"
```

- 5 Reboot the OS.

## Services Requirements

Ensure that the **Server** service and **Remote Registry** service are running on the workstation where the COM server resides.

## Local Security Settings

All settings in this section are configured using the **Local Security Policy** console.

### **To launch the console:**

- 1 Open the Windows **Control Panel**.
- 2 Go to **Administrative Tools**.
- 3 Start the **Local Security Policy**. The **Local Security Settings** window opens.

## Sharing and security model for local accounts

Navigate to **Security Settings > Local Policies > Security Options > Network access: Sharing and security model for local accounts**. Change the setting to **Classic**.

This only applies to Windows computers that are not a part of a domain.

## DCOM Restrictions Policy

Make sure that the user account used has permissions to access, launch, and activate COM/DCOM/Automation objects.

### **To grant these permissions:**

- Add the user to the predefined local group: *Administrators* for Windows XP; or *Distributed COM Users* for Windows Vista, Windows 2003, Windows 2008, and Windows 7.

If you cannot grant the group permission to the user, do the following:

- 1 Create a local user in the **Users** group.
- 2 Navigate to **Control Panel > Administrative Tools > Local Security Policy > Security Settings > Local Policies > Security Options**.
- 3 Double-click **DCOM: Machine Access Restrictions policy**. Click **Edit Security**. Add the user created above. Enable the **Remote Access** option.
- 4 Double-click **DCOM: Machine Launch Restrictions policy**. Click **Edit Security**. Add the user created above. Enable **Local Launch**, **Remote Launch**, **Local Activation**, and **Remote Activation** options.
- 5 Navigate to **Control Panel > Administrative Tools > Component Services > Computers**. Right-click **My Computer**, click **Properties**, and open the **COM Security** tab.
- 6 In the **Access Permissions** section, click **Edit Default**. Add the user created above. Enable the **Remote Access** option.
- 7 In the **Launch and Activation Permissions** section, click **Edit Default**. Add the user created above. Enable the **Local Launch**, **Remote Launch**, **Local Activation**, and **Remote Activation** options.

**i | NOTE:** In the Component Services section you can navigate to a specific component and grant permission from there, instead of doing so from the My Computer menu.

## User Account Control

**For Windows machines that are not part of a domain:**

- 1 Open **Security Settings > Local Policies > Security Options**.
- 2 Disable the **User Account Control: Run all administrators in Admin Approval Mode** option.

## Firewall Settings

**To configure the firewall:**

- 1 Enable all incoming traffic to the default DLL surrogate (*dllhost.exe*).
  - Create a rule that allows all incoming traffic for %systemroot%\system32\dllhost.exe.
  - For 64-bit systems only: create a rule that allows all incoming traffic for %systemroot%\SysWOW64\dllhost.exe.
- 2 Enable COM network access.
  - For Windows XP only: create a rule that allows all incoming traffic for TCP Port 135 (DCE/RPC Locator service).
  - For Windows Vista, 2003, and 2008: enable **COM+ Network Access (DCOM-In)** rule for active profile.
- 3 Enable **File and Printer sharing access**.
  - For Windows XP: enable **File and Printer sharing exception** rule.
  - For Windows Vista, 2003, and 2008: enable all rules in the **File and Printer sharing** group for active profile.

**i | NOTE:** Make sure that the scope defined for rules includes the host, which runs vFoglight.

## Configuration Script

Use the script below to configure the firewall.

- 1 On the target machine create a file named *firewall-config.ps1* with the script listed below.
- 2 Run the script with Administrator's privileges using the following command:  
`powershell -File firewall-config.ps1`

```
# Copyright 2010 Quest.  
# ALL RIGHTS RESERVED.  
#  
# PROPRIETARY/CONFIDENTIAL.  
#  
# This software is the confidential and proprietary information  
# of Quest ("Confidential Information").  
#  
# You shall not disclose such Confidential Information and shall  
# use it only in accordance with the terms of the license agreement  
# you entered into with Quest.  
#
```



```

$OS = Get-WmiObject Win32_OperatingSystem
$OSBuildNumber = $OS.BuildNumber

$OSCaption = $OS.Caption

$useAdvancedFirewall = $true
$COMNetworkAccessGroup = "COM+ Network Access (DCOM-In)"
if (($OSBuildNumber -eq 2600) -or ($OSBuildNumber -eq 3790)) {
    $useAdvancedFirewall = $false
}

if ($OSBuildNumber -eq 7600) {
    # Windows 7
    $COMNetworkAccessGroup = "Windows Management Instrumentation (WMI)"
}

if ($useAdvancedFirewall) {
    Echo "Configuring firewall for Windows Vista/2008/7"
    netsh advfirewall firewall add rule name="DLL Host (32-Bits)" dir=in
action=allow program="%systemroot%\system32\dllhost.exe"
    netsh advfirewall firewall add rule name="DLL Host (64-Bits)" dir=in
action=allow program="%systemroot%\SysWOW64\dllhost.exe"
    netsh advfirewall firewall set rule group=$COMNetworkAccessGroup new enable=yes
    netsh advfirewall firewall set rule group="File and Printer Sharing" new
enable=yes
}
else {
    Echo "Configuring firewall for Windows XP/2003"
    netsh firewall add allowedprogram "%systemroot%\system32\dllhost.exe" "DLL Host
(32-Bits)" ENABLE
    netsh firewall add allowedprogram "%systemroot%\SysWOW64\dllhost.exe" "DLL Host
(64-Bits)" ENABLE
    netsh firewall add portopening TCP 135 "DCE/RPC Locator service" ENABLE
    netsh firewall set service FileAndPrint ENABLE}

```

## COM and Automation Objects

The **COM and Automation** objects are required to perform remote tasks on Windows machines that are not configured for remote activation. Therefore additional configuring of the DLL surrogate is required.

## Registry Permissions

Make sure that the Administrator user has Full Control access to the following registry keys:

- HKEY\_LOCAL\_MACHINE/Software/Classes/AppID
- HKEY\_LOCAL\_MACHINE/Software/Classes/CLSID/{0D43FE01-F093-11CF-8940-00A0C9054228}
- HKEY\_LOCAL\_MACHINE/Software/Classes/CLSID/{13709620-C279-11CE-A49E-444553540000}
- HKEY\_LOCAL\_MACHINE/Software/Classes/CLSID/{72C24DD5-D70A-438B-8A42-98424B88AFB8}
- HKEY\_LOCAL\_MACHINE/Software/Classes/CLSID/{76A64158-CB41-11D1-8B02-00600806D9B6}

## Configuration Script

Use the script below to configure DCOM.

- 1 Create a file named *dcom-config.ps1* that contains the script below on the target machine.

## 2 Run the script on behalf of the **Administrator** user using the following command:

```
runas /user:Administrator powershell -File dcom-config.ps1
```

```
# Copyright 2010 Quest
# ALL RIGHTS RESERVED.
#
# PROPRIETARY/CONFIDENTIAL.
#
# This software is the confidential and proprietary information
# of Quest ("Confidential Information").
#
# You shall not disclose such Confidential Information and shall
# use it only in accordance with the terms of the license agreement
# you entered into with Quest
#

Function ConfigureAppID($appId,$appDescription,$systemGlobal) {
    Echo "Processing $appId ($appDescription)"

    if ($systemGlobal) {
        $classesKey = "HKLM:\Software\Classes"
    }
    else {
        $classesKey = "HKCU:\Software\Classes"
    }

    $AppIDKey = $classesKey + "\AppID"

    if (!(Test-Path $AppIDKey)) {
        $item=New-Item -Name "AppID" -Path $classesKey -Type directory
    }

    $CLASS_AppIDKey = $AppIDKey + "\{$appId}"
    if (!(Test-Path $CLASS_AppIDKey)) {
        $item=New-Item -Name "{$appId}" -Path $AppIDKey -Type directory
    }
    Set-ItemProperty -Path $CLASS_AppIDKey -Name "(default)" -Value
"$appDescription"
    Set-ItemProperty -Path $CLASS_AppIDKey -Name "DllSurrogate" -Value ""
}

Function ConfigureCLSID($clsId,$appId,$systemGlobal) {
    Echo "Processing $clsId"

    $name = (Get-ItemProperty -Path "HKLM:\Software\Classes\CLSID\{$clsId}\ProgID" -
Name "(default)")."(default)"
    Echo $name

    if ($systemGlobal) {
        $classesKey = "HKLM:\Software\Classes"
    }
    else {
        $classesKey = "HKCU:\Software\Classes"
    }
    $CLSIDKey = $classesKey + "\CLSID"
    $AppIDKey = $classesKey + "\AppID"

    if (!(Test-Path $CLSIDKey)) {
        $item=New-Item -Name "CLSID" -Path $classesKey -Type directory
    }
}
```

```

$CLASS_CLSIDKey = $CLSIDKey + "\{$clsId}"
if (!(Test-Path $CLASS_CLSIDKey)) {
    $item=New-Item -Name "{$clsId}" -Path $CLSIDKey -Type directory
}
Set-ItemProperty -Path $CLASS_CLSIDKey -Name "AppID" -Value "{$appId}"
}

Function ConfigureDCOM($appId,$systemGlobal) {
    if (!$appId) {
        $appId = [System.Guid]::NewGuid()
    }
    $appDescription = "Default DLL Surrogate"

    ConfigureAppID -appId $appId -appDescription $appDescription -systemGlobal
    $systemGlobal

    $CLSIDS = @(
        # Scripting.FileSystemObject
        "0D43FE01-F093-11CF-8940-00A0C9054228",
        # Shell.Application
        "13709620-c279-11ce-a49e-444553540000"
        # WScript.Shell
        "72C24DD5-D70A-438B-8A42-98424B88AFB8"
        # WbemScripting.SWbemLocator
        "76A64158-CB41-11D1-8B02-00600806D9B6"
    )

    foreach ($clsId in $CLSIDS) {
        ConfigureCLSID -clsId $clsId -systemGlobal $systemGlobal -appId $appId
    }
}
ConfigureDCOM -systemGlobal $true

```

## Workflows

The Foglight for Microsoft Hyper-V ActionPack includes a number of pre-defined workflows, which are accessible from the Workflow Library in the Automation Workflow Studio.

If none of these workflows suit your needs, you can create your own workflows using the actions described in the next section.

## Actions

This section contains information about the actions included in the Cartridge for Microsoft Hyper-V ActionPack.

### Add Network Adapter

Adds a network adapter to a virtual machine. This action cannot be stopped.

## Pre-conditions:

- VM must be Stopped.
- MAC address must have a valid format. For example, “00155D010101” for the address “00:15:5D:01:01:01”.

Table 2. Add Network Adapter Input Parameters

Name/Scripting Name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine to which a new network adapter will be added.
Network/network	TopologyObject	[HPVVirtualNetwork]. Network that will be assigned to the created network adapter.
Legacy/legacy	Boolean	If true, the legacy network adapter will be created.
MAC address/macAddress	String	MAC address to set onto the created network adapter.

Table 3. Add Network Adapter Output Parameters

Name/Scripting Name	Type	Description
Created NIC/createdNIC	TopologyObject	[HPVNIC]. Created network adapter.

## Apply Snapshot

Restores the saved snapshot of a Hyper-V virtual machine.

Table 4. Apply Snapshot Input Parameters

Name/Scripting name	Type	Description
snapshot/Snapshot	TopologyObject	[HPVSnapshot]. The snapshot to restore.

## Attach Disk

Attaches a virtual hard disk to Hyper-V virtual machine. This action cannot be stopped.

Pre-conditions:

- VM must be Stopped.
- The specified VM location in the specified controller is free for target VM.

Table 5. Attach Disk Input Parameters

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine to which a virtual hard disk will be attached
VHD Path/vhdPath	String	The path to virtual hard disk.
is SCSI controller/isSCSI	Boolean	If true, the virtual hard disk will be attached to a SCSI controller; otherwise it will be attached to an IDE controller

# Clone VM

Clones an existing Hyper-V virtual machine. This action cannot be stopped.

The target server, which will host the cloned VM, should run a network interface that the original VM uses. For example:

- The source server runs 2 networks: *Network1* and *Network2*.
- The target server runs 2 networks: *Network2* and *Network3*.
- VM1 and VM2 are running on the source server.
- VM1 is connected to *Network1*.
- VM2 is connected to *Network2*.

In such configuration you cannot create a clone of VM1 on the target server because it does not run *Network1*. However, you can create a clone of VM2 on the target server, because it has the *Network2* interface.

Table 6. Clone VM Input Parameters

Name/Scripting name	Type	Description
Virtual machine/ <i>VirtualMachine</i>	TopologyObject	[ <i>HPVVirtualMachine</i> ]. Virtual machine to be cloned.
Target Hyper-V server/ <i>targetServer</i>	TopologyObject	[ <i>HPVServer</i> ]. Hyper-V server where the clone will be created.
Target folder/ <i>targetFolder</i>	String	The folder where the clone will be stored.
Virtual Machine Name/ <i>vmName</i>	String	The cloned virtual machine name.

Table 7. Clone VM Output Parameters

Name/Scripting name	Type	Description
Virtual machine/ <i>targetVirtualMachine</i>	TopologyObject	[ <i>HPVVirtualMachine</i> ]. The cloned virtual machine.

# Configure Network Adapter

Configures a network adapter of a Hyper-V virtual machine. This action cannot be stopped.

## Pre-conditions

- VM must be Stopped.
- MAC address must have a valid format, for example "00155D010101" for the address "00:15:5D:01:01:01".

Table 8. Configure Network Adapter Input Parameters

Name/Scripting name	Type	Description
Network adapter/ <i>networkAdapter</i>	TopologyObject	[ <i>HPVNIC</i> ]. Network adapter to configure.
Network/ <i>network</i>	TopologyObject	[ <i>HPVVirtualNetwork</i> ]. Network that will be assigned to the network adapter.
MAC address/ <i>macAddress</i>	String	MAC Address of the network adapter.

# Create Differencing VHD

Creates a differencing virtual hard disk (.VHD) file. The VHD name must not contain characters that are forbidden in Windows file names.

Table 9. Create Differencing VHD Input Parameters

Name/Scripting name	Type	Description
Hyper-V server/ <i>server</i>	TopologyObject	[ <i>HPVServer</i> ]. Hyper-V server, which will be used for creating the VHD.
Storage Folder/ <i>storageFolder</i>	String	The path to the folder where the VHD will be created.
VHD name/ <i>vhdName</i>	String	Name of the new virtual hard disk.
Parent VHD/ <i>parentVhd</i>	String	The parent VHD path.

Table 10. Create Differencing VHD Output Parameters

Name/Scripting name	Type	Description
VHD Path/ <i>vhdPath</i>	String	The created VHD path.

# Create Dynamic VHD

Creates a dynamically-expanding virtual hard disk (.VHD) file. The VHD name must not contain characters that are forbidden in Windows file names.

Table 11. Create Dynamic VHD Input Parameters

Name/Scripting name	Type	Description
Hyper-V server/ <i>server</i>	TopologyObject	[ <i>HPVServer</i> ]. Hyper-V server, which will be used for creating the VHD.
Storage Folder/ <i>storageFolder</i>	String	[ <i>HPVStorageFolder</i> ]. The folder where the VHD will be created.
VHD name/ <i>vhdName</i>	String	Name of the new virtual hard disk.
VHD size (GB)/ <i>vhdSizeGB</i>	Integer	Size of the virtual hard disk in GB.

Table 12. Create Dynamic VHD Output Parameters

Name/Scripting name	Type	Description
VHD/ <i>createdVHD</i>	TopologyObject	[ <i>HPVVirtualDisk</i> ]. Created virtual hard disk.

# Create Fixed Size VHD

Creates a fixed-sized virtual hard disk (.VHD) file. The VHD name must not contain characters that are forbidden in Windows file names.

**Table 13. Create Fixed Size VHD Input Parameters**

Name/Scripting name	Type	Description
Hyper-V server/server	TopologyObject	[HPVServer]. Hyper-V server, used for creating the VHD.
Storage Folder/storageFolder	String	The folder where the VHD will be created.
VHD name/vhdName	String	Name of the new virtual hard disk.
VHDsize (GB)/SizeGB	Integer	Size of the virtual hard disk in GB.

**Table 14. Create Fixed Size VHD OutputParameters**

Name/Scripting name	Type	Description
VHD	TopologyObject	[HPVVirtualDisk]. Created virtual hard disk.

## Create Snapshot

Creates a snapshot of a virtual machine.

**Table 15. Create Snapshot Input Parameters**

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Target virtual machine.
Snapshot name/snapshotName	String	Name of a snapshot that will be created.

**Table 16. Create Snapshot Output Parameters**

Name/Scripting name	Type	Description
Snapshot/createdSnapshot	TopologyObject	[HPVVirtualMachineSnapshot]. Created snapshot.

## Create VM

Creates a virtual machine.

**Table 17. Create VM Input Parameters**

Name/Scripting name	Type	Description
Hyper-V server/targetServer	TopologyObject	[HPVServer]. Microsoft Hyper-V Server where the VM will be located.
VM location/vmLocation	String	The virtual machine location. This is an optional parameter. If absent the server's default virtual machine location is used.
VM name/vmName	String	Name of the virtual machine.
Memory size (MB)/memorySizeMB	String	Amount of memory to allocate to the VM.
CPU count/cpuCount	Integer	Number of virtual CPUs for the VM.

**Table 17. Create VM Input Parameters**

Name/Scripting name	Type	Description
Virtual machine reserve/cpuReservation	Integer	Virtual machine reserve (percentage).
Virtual machine limit/cpuLimit	Integer	Virtual machine limit (percentage).
Relative weight/cpuWeight	Integer	Relative weight.
Notes/notes	String	Notes about virtual machine.

**Table 18. Create VM Output Parameters**

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. The created virtual machine.

## Delete VM

Deletes a Hyper-V virtual machine.

**Table 19. Delete VM Input Parameters**

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine to delete.
Delete content/deleteContent	Boolean	If true, the virtual machine content (for example VHD images) will be deleted (default value is false).

## Edit Drive

Connects a new virtual disk image to a virtual hard drive.

## Pre-conditions

- VM must be Turned Off. Any other state will result in an `Illegal` state for this operation exception.

**Table 20. Edit Drive Input Parameters**

Name/Scripting name	Type	Description
Hard Drive/hardDrive	TopologyObject	[HPVVirtualMachinePhysicalDisk]. The hard drive.
VHD path/vhdPath	String	The path to the virtual hard disk

## Edit VM

Changes the Hyper-V virtual machine basic settings. You should specify all parameters. If you are not going to modify a parameter, specify its current value.

Changing the CPU count and memory size requires the virtual machine to be in the Turned Off state.



**Table 21. Edit VM Input Parameters**

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine to edit.
VM name/vmName	String	The virtual machine name. Cannot be null or empty.
VM notes/notes	String	The virtual machine notes. Null and empty values are accepted.
Virtual memory size/memorySizeMb	Integer	The virtual memory size in MB. Valid range is between 8 and 4094 MB.
Virtual processor count/cpuCount	Integer	The virtual processor count. Valid range is between 1 and the actual number of processors on the physical machine.
Virtual machine reserve/cpuReservation	Integer	Percent of the logical CPU that is set aside for the running virtual machine. As each VM is started, the available capacity on the Hyper-V server is reduced.
Virtual machine limit/cpuLimit	Integer	Percentage of logical CPU that a running virtual machine is not allowed to exceed.
Relative weight/cpuWeight	Integer	How CPU is distributed when there is contention among all running virtual machines. The higher the number, the more processing power is allocated to the VM.

## Move VM

Moves a powered off virtual machine.

### Pre-conditions

- Virtual machine must be located on a shared storage.

The target server, which will host the VM, should run a network interface that the VM uses. For example:

- The source server runs 2 networks: *Network1* and *Network2*.
- The target server runs 2 networks: *Network2* and *Network3*.
- VM1 and VM2 are running on the source server.
- VM1 is connected to *Network1*.
- VM2 is connected to *Network2*.

In such configuration you cannot move the VM1 to the target server, because it does not have the *Network1*. However, you can move VM2 to the target server, because it has the *Network2* interface.

**Table 22. Move VM Input Parameters**

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine to move.
MS Hyper-V server/targetServer	TopologyObject	[HPVServer]. Server to which the virtual machine will be moved.

Table 23. Move VM Output Parameters

Name/Scripting name	Type	Description
VM/resultVM	TopologyObject	[HPVVirtualMachine]. Virtual machine.

## Pause VM

Pauses a virtual machine. It must be Running to be paused.

Table 24. Pause VM Input Parameters

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine to pause.

## Reboot Hyper-V Server

Reboots a Hyper-V server. The Hyper-V server must be Powered On.

Table 25. Reboot Hyper-V Server Input Parameters

Name/Scripting name	Type	Description
MS Hyper-V server/server	TopologyObject	[HPVServer]. The Hyper-V server to reboot.
Reason/reason	String	The reason for the shutdown operation.
Force/force	Boolean	If true, forces applications to close despite having unsaved data.

## Remove All Network Adapters

Removes all network adapters for the selected virtual machine. The virtual machine must be Stopped.

Table 26. Remove All Network Adapters Input Parameters

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine on which network adapters will be removed.

## Remove All Snapshots

Removes all snapshots of the specified Hyper-V virtual machine.

Table 27. Remove All Snapshots Input Parameters

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine from which snapshots are to be removed.

# Remove Drive

Disconnects a virtual hard disk from a virtual machine, but does not delete the .VHD file.

Table 28. Remove Drive Input Parameters

Name/Scripting name	Type	Description
Hard drive/drive	TopologyObject	[HPVHardDrive]. Hard drive that will be disconnected.

# Remove Network Adapter

Removes a network adaptor of the selected virtual machine. The virtual machine must be Stopped.

Table 29. Remove Network Adapter Input Parameters

Name/Scripting name	Type	Description
Network adaptor/nic	TopologyObject	[HPVNIC]. Network adapter to remove.

# Remove Snapshot

Deletes a virtual machine snapshot.

Table 30. Remove Snapshot Input Parameters

Name/Scripting name	Type	Description
Snapshot/snapshot	TopologyObject	[HPVVirtualMachineSnapshot]. Snapshot to delete.
Delete children/removeChildren	Boolean	If true, the entire snapshot sub-tree is deleted.

# Rename Snapshot

Renames a virtual machine snapshot.

Table 31. Rename Snapshot Input Parameters

Name/Scripting name	Type	Description
Snapshot/snapshot	TopologyObject	[HPVVirtualMachineSnapshot]. Snapshot to rename.
Snapshot name/snapshotName	String	A user-defined string that specifies the new name.

# Resume VM

Resumes a paused virtual machine.

Table 32. Resume VM Input Parameters

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine to resume.

# Revert to Snapshot

Restores virtual machine from a snapshot.

Table 33. Revert to Snapshot Input Parameters

Name/Scripting name	Type	Description
Snapshot/snapshot	TopologyObject	[HPVVirtualMachineSnapshot]. Snapshot to revert.

# Save VM

Saves the guest session image into a file. The virtual machine must be either Running or Paused.

Table 34. Save VM Input Parameters

Name/Scripting name	Type	Description
Virtual machine/ virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine on which to save the guest session image.

# Set MAC Address

Sets the MAC address of a virtual machine.

## Pre-conditions:

- VM must be Stopped.
- MAC address must have a valid format. For example, "00155D010101" for the address "00:15:5D:01:01:01".

Table 35. Set MAC Address Input Parameters

Name/Scripting name	Type	Description
Network Adapter/Network Adapter	TopologyObject	[HPVVirtualMachineNetworkInterface] Network adapter for the virtual machine.
MAC Address/ MAC Address	String	MAC address to assign to the virtual machine.

# Set Network

Sets the network for a virtual machine.

Table 36. Set Network Input Parameters

Name/Scripting name	Type	Description
Network Adapter/Network Adapter	TopologyObject	[HPVVirtualMachineNetworkInterface] Network adapter for the virtual machine.
Network/network	TopologyObject	[HPVVirtualSwitch]. Network that will be assigned to the network adapter.

# Shutdown Hyper-V Server

Shuts down a Hyper-V server. The Hyper-V server must be Running.

## Input Parameters

Table 37. Shutdown Hyper-V Server Input Parameters

Name/Scripting name	Type	Description
MS Hyper-V server/server	TopologyObject	[HPVServer]. Microsoft Hyper-V server to shut down.
Reason/reason	String	The reason for the shutdown operation. This is a user-defined string.
Force/force	Boolean	If true, forces applications to close despite having unsaved data.

## Shutdown VM

Shuts down a virtual machine. The virtual machine must be Running.

Table 38. Shutdown VM Input Parameters

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine that will be shut down.
Reason/reason	String	The reason for the shutdown operation. This is a user-defined string.
Force/force	Boolean	If true, forces applications to be closed despite having unsaved data.

## Start VM

Powers on a virtual machine. The virtual machine must be Stopped, Saved, or Paused.

Table 39. Start VM Input Parameters

Name/Scripting name	Type	Description
Virtual machine/virtualMachine	TopologyObject	[HPVVirtualMachine]. Virtual machine to start.
Wait OS start/waitOSStart	Boolean	If true, the action will wait while the OS of the VM starts (default value is false).

## Stop VM

Powers off a virtual machine. The virtual machine must be Running.

**Table 40. Stop VM Input Parameters**

<b>Name/Scripting name</b>	<b>Type</b>	<b>Description</b>
Virtual machine/virtualMachine	TopologyObject	[ <i>HPVVirtualMachine</i> ]. Virtual machine to power off.

## Wait VM State

Wait for the virtual machine to enter a specified state.

**Table 41. Wait VM State Input Parameters**

<b>Name/Scripting name</b>	<b>Type</b>	<b>Description</b>
Timeout	Integer	The amount of time to wait for the action to complete.
Expected State	String	A string describing the state of the virtual machine.
Virtual machine/virtualMachine	TopologyObject	[ <i>HPVVirtualMachine</i> ]. Virtual machine name.

## We are more than just a name

We are on a quest to make your information technology work harder for you. That is why we build community-driven software solutions that help you spend less time on IT administration and more time on business innovation. We help you modernize your data center, get you to the cloud quicker and provide the expertise, security and accessibility you need to grow your data-driven business. Combined with Quest's invitation to the global community to be a part of its innovation, and our firm commitment to ensuring customer satisfaction, we continue to deliver solutions that have a real impact on our customers today and leave a legacy we are proud of. We are challenging the status quo by transforming into a new software company. And as your partner, we work tirelessly to make sure your information technology is designed for you and by you. This is our mission, and we are in this together. Welcome to a new Quest. You are invited to Join the Innovation™.

## Our brand, our vision. Together.

Our logo reflects our story: innovation, community and support. An important part of this story begins with the letter Q. It is a perfect circle, representing our commitment to technological precision and strength. The space in the Q itself symbolizes our need to add the missing piece—you—to the community, to the new Quest.

## Contacting Quest

For sales or other inquiries, visit <https://www.quest.com/company/contact-us.aspx> or call +1-949-754-8000.

## Technical support resources

Technical support is available to Quest customers with a valid maintenance contract and customers who have trial versions. You can access the Quest Support Portal at <https://support.quest.com>.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request.
- View Knowledge Base articles.
- Sign up for product notifications.
- Download software and technical documentation.
- View how-to-videos.
- Engage in community discussions.
- Chat with support engineers online.
- View services to assist you with your product.

