

Virtual Machine Management Guide

Novell® PlateSpin Orchestrate

2.5

March 11, 2011

www.novell.com



Legal Notices

Novell, Inc., makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc., makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the [Novell International Trade Services Web page \(http://www.novell.com/info/exports/\)](http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 2008-2011 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see the [Novell Documentation Web page \(http://www.novell.com/documentation\)](http://www.novell.com/documentation).

Novell Trademarks

For Novell trademarks, see the [Novell Trademark and Service Mark list \(http://www.novell.com/company/legal/trademarks/tmlist.html\)](http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Guide	5
1 Managing Virtual Machine Hosts	7
1.1 Discovering VM Hosts and Repositories	7
1.2 Discovering VM Images	8
1.3 Resynchronizing the VM Host's State	9
1.4 Shutting Down VM Hosts	9
1.5 Restarting VM Hosts	10
1.6 Understanding VM Host Failover	11
2 Managing Virtual Machines	13
2.1 Provisioning a Virtual Machine	13
2.2 Provisioning Actions and History	14
2.2.1 What are Provisioning Actions?	14
2.2.2 How Actions Are Displayed in the Development Client	14
2.3 Managing a Running Virtual Machine	17
2.3.1 Using the Right-Click Menu for Provisioning Actions	17
2.3.2 Prerequisites for Creating or Deleting vNIC and vDisk Objects on Hyper-V Managed Linux VMs	21
2.3.3 Releasing a Virtual Machine from Usage	21
2.3.4 Managing Virtual Machine Templates	21
2.3.5 Managed Virtual Machine Actions	23
2.3.6 Virtual Machine Technology-Specific Actions	24
2.4 Resynchronizing the State of All VMs	24
2.5 Resynchronizing the State of All VMs of a Specific VM Host	24
2.6 Shutting Down Multiple VMs	25
2.7 Destroying and Deleting a Virtual Machine	25
3 Managing VM Repositories	27
3.1 Provisioning a VM from a Local Repository	27
3.2 The Datagrid Repository	28
3.3 Provisioning a VM from a NAS Repository	28
3.4 The SAN Repository	28
4 Configuring VM Provisioning Adapters and Discovered VMs	29
4.1 Configuring the VMware vSphere Provisioning Adapter	29
4.1.1 Initial Configuration	29
4.1.2 Configuring Other Policies for vSphere	32
4.1.3 Assigning a vSphere VM to a Resource Pool	33
4.1.4 Setting Up VNC for a VM Managed by vSphere	34
4.1.5 Setting Up Orchestrate to Accommodate VMware DRS Clustering and Updates	36
4.1.6 Discovering Enterprise Resources in Multiple vSphere Environments	40
4.1.7 Known Configuration Limitations for Virtual Center VMs	43
4.2 Configuring the Xen Provisioning Adapter and Xen VMs	44
4.2.1 Configuring Policies for Xen	44
4.2.2 Known Configuration Limitations for Xen VMs	44

4.3	Configuring the Amazon EC2 Provisioning Adapter	45
4.3.1	Components of the EC2 Provisioning Adapter	45
4.3.2	EC2 Objects Mapped to PlateSpin Orchestrate Objects	45
4.3.3	Managing VMs in the EC2 Cloud	45
4.3.4	Specifying Properties of an EC2 Template	46
4.4	Configuring the hyperv Provisioning Adapter and Hyper-V VMs	48
4.4.1	Configuring Policies for the hyperv Provisioning Adapter	48
4.4.2	Known Configuration Limitations for Hyper-V VMs	48
5	Troubleshooting Provisioning Actions	51
6	Autoprep and Sysprep	55
6.1	Understanding Autoprep	55
6.1.1	How Autoprep Works	55
6.1.2	Setting Autoprep Facts in the Development Client	56
6.1.3	Applying Autoprep Facts	59
6.1.4	Example Autoprep Scenarios	60
6.1.5	Known Autoprep Limitations	60
6.2	Understanding Sysprep	61
6.2.1	How Sysprep Works	62
6.2.2	Setting Sysprep Facts in the Development Client	63
6.2.3	Using the Sysprep deploy.cab Files	72
6.2.4	Applying Sysprep Facts	74
6.2.5	Example Sysprep Scenarios	75
6.2.6	Known Sysprep Limitations	75
A	Virtual Machine Technologies and Actions	79
A.1	Virtual Machine Technologies	79
A.2	Xen Hypervisor on SLES 10 SP3 or SLES 11 Host	81
A.2.1	Additional Xen Provisioning Adapter Information	82
A.3	VMware vSphere	83
A.3.1	Additional VMware vSphere Provisioning Adapter Information	85
A.4	Microsoft Hyper-V Hypervisor	85
A.5	Amazon EC2	87
B	Tuning the vsphere Provisioning Adapter for Large Scale VM Provisioning	89
C	Discovering or Dynamically Creating vSphere Port Groups	91
C.1	How Orchestrate Models a vSphere Port Group and Exposes vSwitch Settings	91
C.2	Dyanamically Creating Port Groups	91
C.2.1	Network Configuration Facts	92
C.2.2	Enabling Dynamic Creation of Port Groups	92
D	Documentation Updates	93
D.1	March 11, 2011	93
D.2	November 12, 2010	93
D.3	October 15, 2010	93

About This Guide

In addition to managing virtual machines (VMs) and host servers by using the PlateSpin Orchestrate VM Client, you can do other management work with the PlateSpin Orchestrate Development Client. This *Virtual Machine Management Guide* provides instructions on the management tasks that you can do in the Development Client.

The guide is organized as follows:

- ♦ [Chapter 1, “Managing Virtual Machine Hosts,” on page 7](#)
- ♦ [Chapter 2, “Managing Virtual Machines,” on page 13](#)
- ♦ [Chapter 3, “Managing VM Repositories,” on page 27](#)
- ♦ [Chapter 4, “Configuring VM Provisioning Adapters and Discovered VMs,” on page 29](#)
- ♦ [Chapter 5, “Troubleshooting Provisioning Actions,” on page 51](#)
- ♦ [Chapter 6, “Autoprep and Sysprep,” on page 55](#)
- ♦ [Appendix A, “Virtual Machine Technologies and Actions,” on page 79](#)
- ♦ [Appendix B, “Tuning the vsphere Provisioning Adapter for Large Scale VM Provisioning,” on page 89](#)
- ♦ [Appendix C, “Discovering or Dynamically Creating vSphere Port Groups,” on page 91](#)
- ♦ [Appendix D, “Documentation Updates,” on page 93](#)

For documentation on using Orchestrate jobs to further manage VMs, host machines, and physical machines, see “[Virtual Machine Job Development](#)” in the *PlateSpin Orchestrate 2.5 Developer Guide and Reference*.

Audience

This book is for data center administrators. It assumes that users of the product have the following background:

- ♦ General understanding of network operating environments and systems architecture
- ♦ Knowledge of basic Linux shell commands, the Windows command prompt, and text editors

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html (<http://www.novell.com/documentation/feedback.html>) and enter your comments there.

Additional Documentation

In addition to this *Virtual Machine Management Guide*, PlateSpin Orchestrate 2.5 includes the following additional guides that contain valuable information about the product:

- ♦ [PlateSpin Orchestrate 2.5 Getting Started Reference](#)
- ♦ [PlateSpin Orchestrate 2.5 Installation and Configuration Guide](#)

- ◆ *PlateSpin Orchestrate 2.5 Upgrade Guide*
- ◆ *PlateSpin Orchestrate 2.5 VM Client Guide and Reference*
- ◆ *PlateSpin Orchestrate 2.5 Development Client Reference*
- ◆ *PlateSpin Orchestrate 2.5 Administrator Reference*
- ◆ *PlateSpin Orchestrate 2.5 High Availability Configuration Guide*
- ◆ *PlateSpin Orchestrate 2.5 Command Line Reference*
- ◆ *PlateSpin Orchestrate 2.5 Server Portal Reference*
- ◆ *PlateSpin Orchestrate 2.5 Developer Guide and Reference*

Managing Virtual Machine Hosts

1

After you install the PlateSpin Orchestrate Agent on a physical resource, the Xen and Hyper-V hypervisor technologies running on that resource are determined by the Discover VM hosts job. You can then discover and manage VMs residing on the VM hosts.

For the VMware vSphere hypervisor, however, you need to associate the `vsphere_client` policy to a VSphere resource before the discovery works.

- ♦ [Section 1.1, “Discovering VM Hosts and Repositories,” on page 7](#)
- ♦ [Section 1.2, “Discovering VM Images,” on page 8](#)
- ♦ [Section 1.3, “Resynchronizing the VM Host’s State,” on page 9](#)
- ♦ [Section 1.4, “Shutting Down VM Hosts,” on page 9](#)
- ♦ [Section 1.5, “Restarting VM Hosts,” on page 10](#)
- ♦ [Section 1.6, “Understanding VM Host Failover,” on page 11](#)

1.1 Discovering VM Hosts and Repositories

- 1 Ensure that the policies appropriate to the VM technology are configured. For more information, see [Chapter 4, “Configuring VM Provisioning Adapters and Discovered VMs,” on page 29](#).

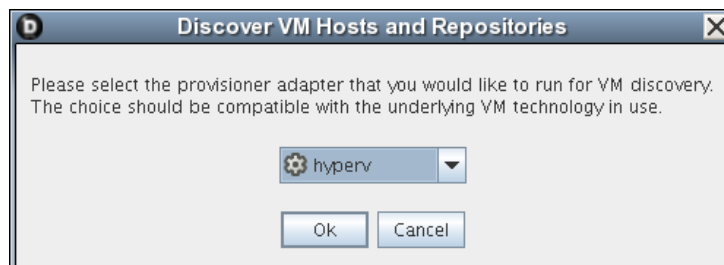
For vSphere, the default number of slots is 4. We recommend that you increase this number to 10, depending on the hardware and available computing resources (RAM/CPU) of the server where the agent is running (this is also the server associated with the `vsphere_client` policy). Each joblet slot causes a separate Java instance on this resource. Each Java instance uses 50-60 MB of RAM and is quite CPU-intensive.

For Xen, we recommend that you accept the default slot number of 1. No more than one provision operation should happen concurrently on the Xen host, particularly any operation that is disk-related.

For more information on the policies, see [Chapter 4, “Configuring VM Provisioning Adapters and Discovered VMs,” on page 29](#).

- 2 Ensure that you have set the correct number of joblet slots for the VM hosts in the policies appropriate to the VM technology. For more information, see [Joblet Slots](#) in the [PlateSpin Orchestrate 2.5 Development Client Reference](#).
- 3 In the Development Client, click *Provision > Discover VM Hosts and Repositories*.

The Discover VM Hosts and Repositories dialog box is displayed.



- 4 Select your provisioning adapter from the drop-down list.
- 5 Click *OK*.
- 6 Click *Jobs* to view the *Jobs* section in the Development Client and verify that the job has started.

After your VM host machines are discovered, you can refresh your tree view or wait for the automatic tree refresh to see the VM host machine listed under the provisioning adapter, although no VMs are listed.

This also discovers:

- ♦ Local repositories for all types of hypervisors.
- ♦ SAN and NAS repositories for Xen and vSphere.

To view the discovered repositories, click *Repositories*, then click *xen30* or *esx*.

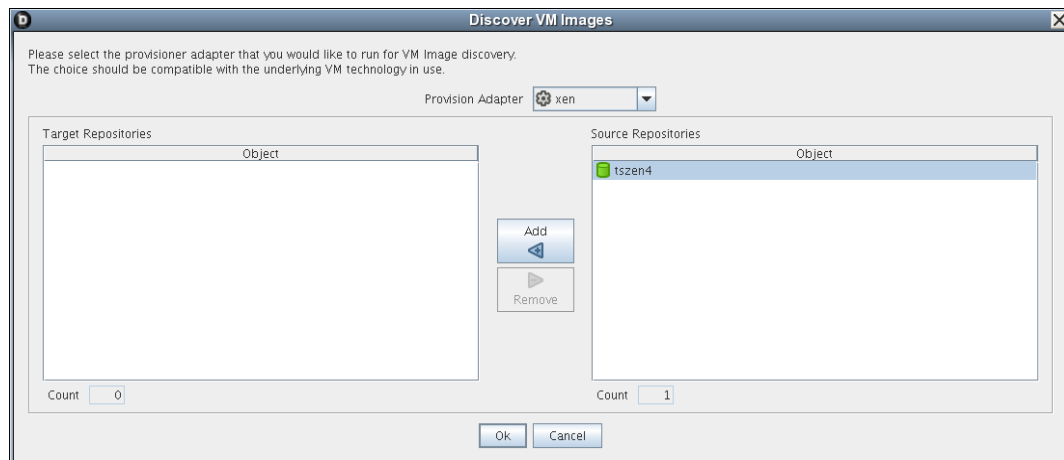
For a list of the VM technologies and supported host and guest operating systems, see [Section A.1, “Virtual Machine Technologies,” on page 79](#).

1.2 Discovering VM Images

To discover the VM images on a specific repository:

- 1 In the Development Client, click *Provision > Discover VM Images*.

The Discover VM Images dialog box is displayed.



- 2 In the *Provisioning Adapter* drop-down list, select the provisioning adapter for which you want to discover the VM images.

The source repositories for the selected provisioning adapter are displayed.

For information on provisioning adapters, see [Section 2.1, “Provisioning a Virtual Machine,” on page 13](#).

- 3 Select the source repositories, then click *Add*.

The selected repositories are added to the *Target Repositories* pane.

4 Click *OK*.

The VM images are discovered: a separate job is launched for each repository that the user selected. After all jobs are complete, you can refresh the Explorer tree in the Orchestrate Development Client to see the discovered VMs.

1.3 Resynchronizing the VM Host's State

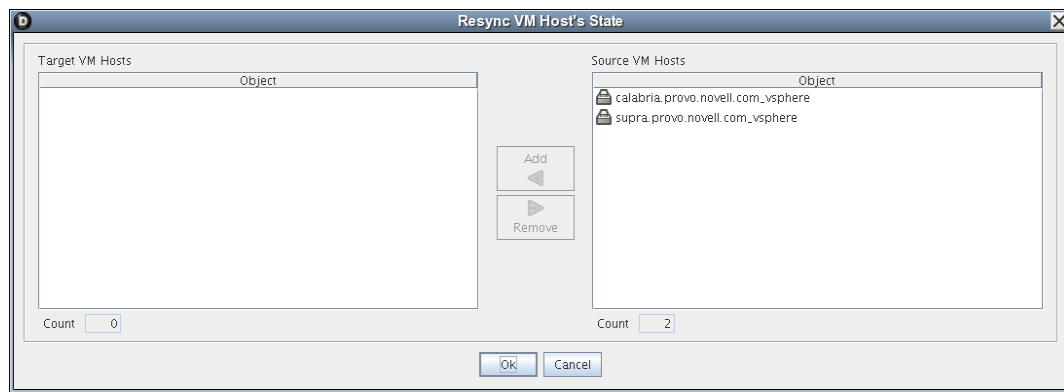
To manually verify and ensure that the state of a VM host displayed in the Development Client is accurate:

1 In the Development Client, right-click the VM host, then click *Discover*.

To manually verify and ensure that the state of multiple VM hosts displayed in the Development Client is accurate:

1 In the Development Client, click *Provision > Resync VM Host's State*.

The Resync VM Host's State dialog box is displayed.



2 In the *Source VM Hosts* pane, select the VM hosts to be resynchronized, then click *Add*.

The selected VM hosts are added to the *Target VM Hosts* pane.

3 Click *OK*.

1.4 Shutting Down VM Hosts

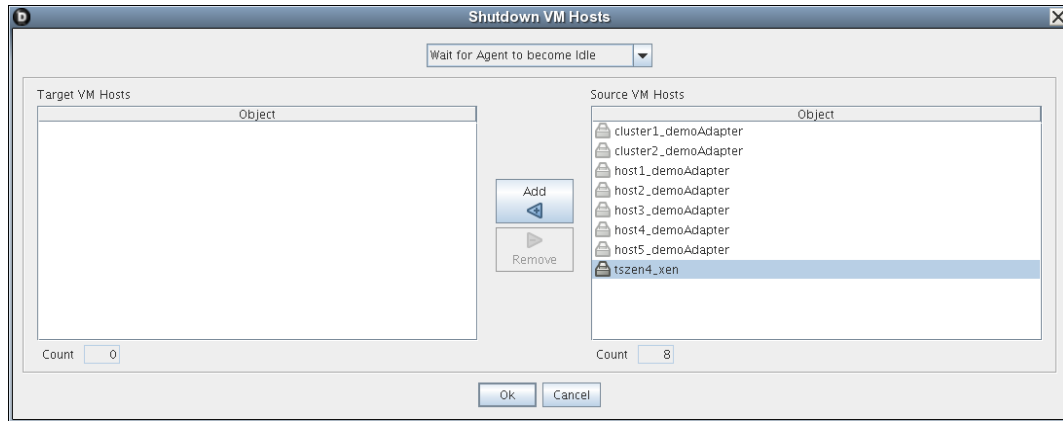
To shut down a single VM host:

1 In the Development Client, right-click the VM host you want to shut down, then click *Shutdown*.

To shut down multiple VM hosts:

1 In the Development Client, click *Provision > Shutdown Hosts*.

The Shut Down VM Hosts dialog box is displayed.



2 Choose when to shut down the VM hosts.

You can choose to shut down the VM hosts after the Orchestrate Agent becomes idle or to immediately shut down the VM hosts. By default, the *Wait for Agent to become Idle* option is selected.

3 In the *Source VM Hosts* pane, select the VM hosts you want to shut down, then click *Add*.

The selected VM hosts are added to the *Target VM Hosts* pane.

4 Click *OK*.

The VMs running on the host are automatically shut down and the VM host is moved to the Shutting Down state, where it will not accept any Provisioning actions.

1.5 Restarting VM Hosts

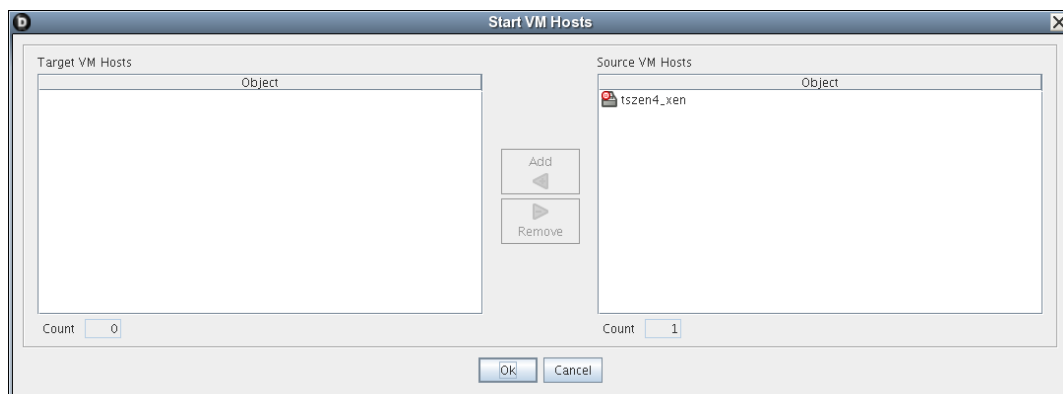
To restart a single VM host:

1 In the Development Client, right-click the VM host you want to start, then click *Start*.

To restart multiple VM hosts:

1 In the Development Client, click *Provision > Start VM Hosts*.

The Start VM Hosts dialog box is displayed.



2 In the *Source VM Hosts* pane, select the VM hosts you want to restart, then click *Add*.

The selected VM hosts are added to the *Target VM Hosts* pane.

3 Click *OK*.

1.6 Understanding VM Host Failover

When the PlateSpin Orchestrate Server comes back online after being offline, it rediscovers the state of all resources, including VM hosts and the VMs running on those hosts. This section provides more information about how the Orchestrate Server behaves when the VM Host loses its agent connection.

There are two possible scenarios that can occur when a VM Host fails while running VMs. The failover behavior depends on where the VM image is stored and whether the VM has the agent installed.

The following table shows possible failover scenarios with the VM Host and the expected server behavior when it occurs.

Table 1-1 Orchestrate Server Behavior when the VM Host Loses Its Agent Connection

Scenario	Failover Behavior
<p>Scenario 1: The VM image is:</p> <ul style="list-style-type: none">◆ Stored on a non-local repository (for example, the zos repository)◆ Accessible by other VM hosts◆ Successfully provisioned <p>Situation: The VM host fails.</p>	<p>The VMs that had been running on the failed VM host are reprovisioned to other available VM hosts.</p> <ul style="list-style-type: none">◆ If the VM was provisioned from a template, there is now another instance of the VM. For example, if the template name is <code>sles10template</code>, the original VM provisioned from the template is then named <code>sles10template-1</code>. <p>If the host running <code>sles10template-1</code> goes down, or if it loses its agent connection, a new instance of the template named <code>sles10template-2</code> is reprovisioned to an available host.</p> <ul style="list-style-type: none">◆ If the original VM was a standalone VM, it is reprovisioned to an available host.
<p>Scenario 2: The VM image is stored on a local repository.</p> <p>Situation: The VM host loses its agent connection.</p>	<ul style="list-style-type: none">◆ Because the VM image is stored locally, it cannot be reprovisioned to another VM host.◆ When the VM host comes back online, it is reprovisioned to the host where it is stored.

In either of these scenarios, if the Orchestrate Agent is installed on the VM and if the VM host loses its agent connection but the VMs retain their agent connection (for example, if someone kills the agent process on the VM host), no reprovisioning occurs.

If the VM host loses its agent connection, and if the Orchestrate Agent is not installed on the running VMs, the VMs can continue running indefinitely. However, if the location of the VM image warrants it, the VMs are reprovisioned to other available hosts. When there are two (or more) of the

same VM instance running on different VM hosts, the Orchestrate Server is aware only of the VMs running on a VM host with an active agent connection, so the administrator must stop the VMs on the host that has lost its agent connection.

NOTE: If you are interested in failover in a high availability environment, see the [PlateSpin Orchestrate 2.5 High Availability Configuration Guide](#).

Managing Virtual Machines

2

Review the following sections for information about the ongoing tasks in PlateSpin Orchestrate VM Management:

- ◆ [Section 2.1, “Provisioning a Virtual Machine,” on page 13](#)
- ◆ [Section 2.2, “Provisioning Actions and History,” on page 14](#)
- ◆ [Section 2.3, “Managing a Running Virtual Machine,” on page 17](#)
- ◆ [Section 2.4, “Resynchronizing the State of All VMs,” on page 24](#)
- ◆ [Section 2.5, “Resynchronizing the State of All VMs of a Specific VM Host,” on page 24](#)
- ◆ [Section 2.6, “Shutting Down Multiple VMs,” on page 25](#)
- ◆ [Section 2.7, “Destroying and Deleting a Virtual Machine,” on page 25](#)

2.1 Provisioning a Virtual Machine

Provisioning is the first step in a VM’s life cycle. The Orchestrate Server determines the best VM host machine for running the VM, unless you select a specific host server, datastore, or network to run the VM.

By default, you can run eight VMs at one time on a VM host. If you want to provision additional VMs, you must proportionately increase the `vmhost.maxvmslots` fact value for a particular VM host in the Orchestrate Development Client.

Provisioning VMs that have only an NPIV disk is not supported. You can provision a VM that has a hard disk and an NPIV disk (SAN repository). The OS image of the VM is stored on the local hard disk and the data resides in the SAN repository.

The Orchestrate Server uses provisioning adapters to perform life cycle functions. Provisioning adapters are programs that control (start, stop, snapshot, migrate, or pause) a VM. They run as regular jobs on the Orchestrate Server.

The system can discover SAN repositories for Xen and vSphere hosts.

The constraints used to determine a suitable VM host evaluate the following criteria to provide heterogeneous VM management:

- ◆ Machine architectures
- ◆ CPU
- ◆ Bit width
- ◆ Available virtual memory
- ◆ Pther administrator-configured constraints, such as the number of virtual machine slots

For procedures and more information on provisioning VMs, see [Section 2.3, “Managing a Running Virtual Machine,” on page 17](#).

2.2 Provisioning Actions and History

The following information is included in this section:

- ◆ [Section 2.2.1, “What are Provisioning Actions?” on page 14](#)
- ◆ [Section 2.2.2, “How Actions Are Displayed in the Development Client,” on page 14](#)

2.2.1 What are Provisioning Actions?

The provisioning operations you perform in PlateSpin Orchestrate are recorded as “actions.” For example, in the PlateSpin Orchestrate Development Client main menu:

- ◆ A VM Host Discovery action is initiated if you select *Provision > Discover VM Hosts and Repositories* and then you select a provisioning adapter in the Discover VM Hosts and Repositories dialog box.
- ◆ A VM Discovery action is initiated if you select *Provision > Discover VM Images* and then you select a provisioning adapter in the Discover VM Images dialog box. An action is specified for each Repository you specify.
- ◆ A Migrate action is initiated when you perform the migration of a Virtual Machine (VM).

2.2.2 How Actions Are Displayed in the Development Client

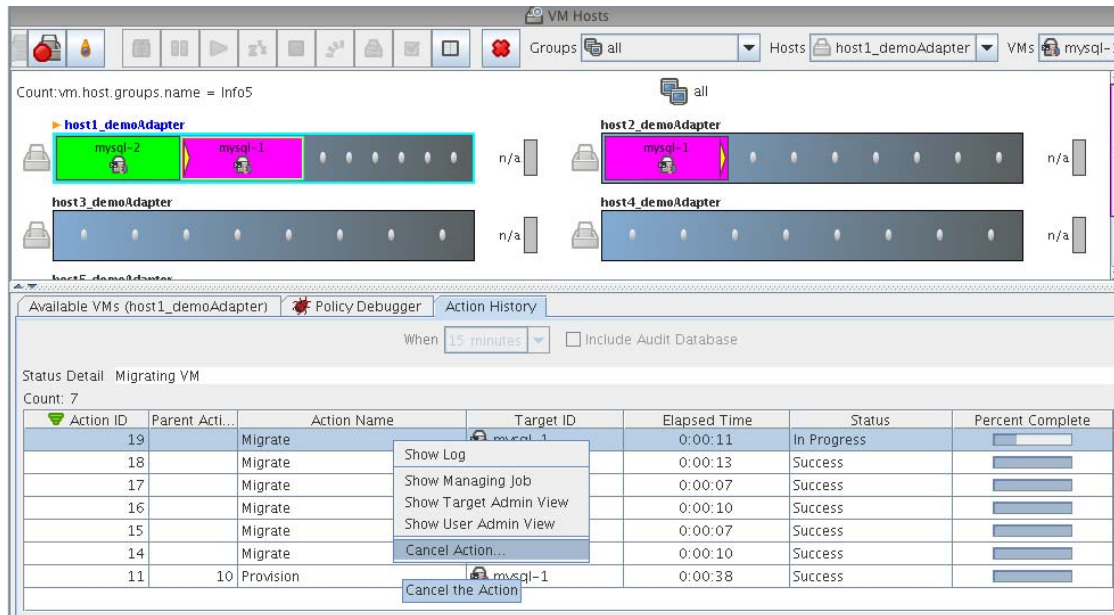
Depending on the Grid object you select, an Action History tab is displayed in several views of the Orchestrate Development Client.

- ◆ [“Action History in Monitor Views of the Development Client” on page 14](#)
- ◆ [“Action History in Admin Views of the Development Client” on page 15](#)

Action History in Monitor Views of the Development Client

You can see the Action History tab in the VM Hosts monitor view if you select a migrating VM:

Figure 2-1 VM Hosts Actions



Two action-specific menu selections are available if you right-click an action in the action history table:

- ◆ *Show Log* opens the provision log for the VM
- ◆ *Cancel Action* cancels the selected active action

The action history table is updated at the same time the polling view is updated.

NOTE: The format of action history table is similar in the Provisioner monitor view and in the Users monitor view.

If the *Include Audit Database* check box is selected in this view, the action status is not polled. Click the refresh icon to fetch and display fresh data.

NOTE: The Orchestrator Server must be connected to an audit database for the *Include Audit Database* check box to be available. This behavior is the same in the Job monitor view.

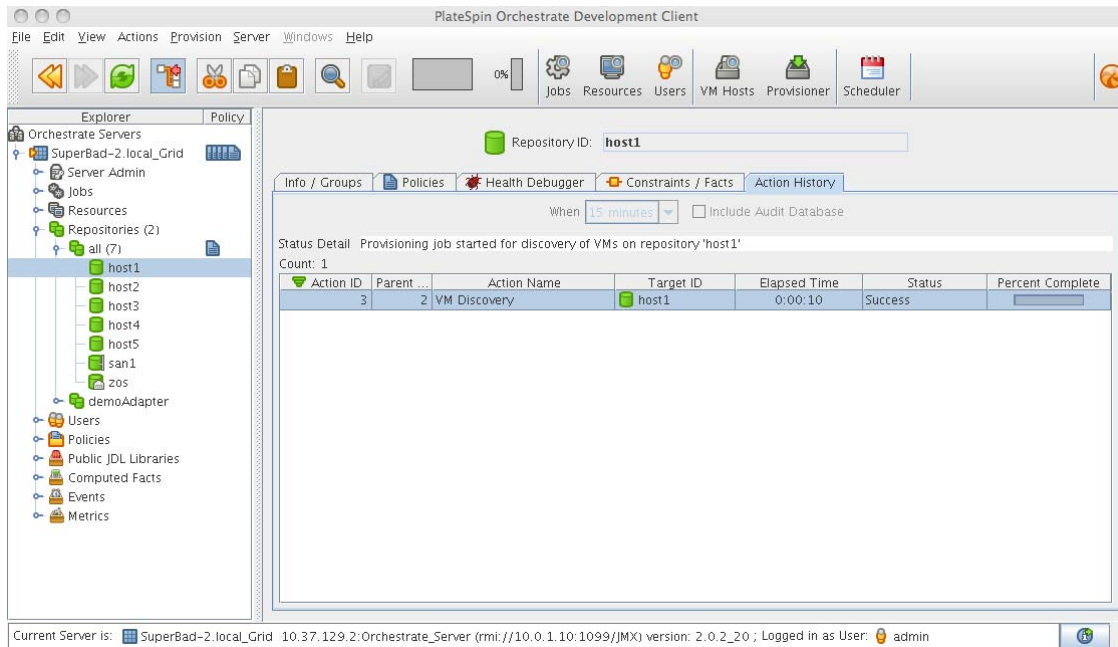
Action History in Admin Views of the Development Client

Action history is displayed in other Grid object admin views of the Development Client:

- ◆ User object
- ◆ Resource object
- ◆ Repository object

The following illustration shows an example of action history in the Repository admin view:

Figure 2-2 Repository Action History



The table below defines some of the column names and the values that can populate those columns in the action history table:

Table 2-1 Action History Table Columns

Column Name	Purpose
Action ID	A unique integer value used to identify the action.
Parent Action ID	If a value is displayed, the action is a child of this parent, identified with a unique integer value.
Action Name	Specific to the action being invoked.
Target ID	The identifying string of the object where the action is to occur.
Status	Displays the status of this action (specific to the action being invoked). Possible values in this column include: <ul style="list-style-type: none"> ◆ Started ◆ In Progress ◆ Success ◆ Failed ◆ Canceled

2.3 Managing a Running Virtual Machine

There are many ways you can control a VM after it has been deployed. All actions from provisioning to shutting down the VM can be managed directly from the Orchestrate Development Client, through provisioning adapter jobs, and through custom jobs written by the user.

Review the following sections for ways to manage running VMs:

- ♦ [Section 2.3.1, “Using the Right-Click Menu for Provisioning Actions,” on page 17](#)
- ♦ [Section 2.3.2, “Prerequisites for Creating or Deleting vNIC and vDisk Objects on Hyper-V Managed Linux VMs,” on page 21](#)
- ♦ [Section 2.3.3, “Releasing a Virtual Machine from Usage,” on page 21](#)
- ♦ [Section 2.3.4, “Managing Virtual Machine Templates,” on page 21](#)
- ♦ [Section 2.3.5, “Managed Virtual Machine Actions,” on page 23](#)
- ♦ [Section 2.3.6, “Virtual Machine Technology-Specific Actions,” on page 24](#)

2.3.1 Using the Right-Click Menu for Provisioning Actions

You can perform provisioning actions by right-clicking a VM in the Explorer tree of the Orchestrate Development Client.

For information on provisioning adapters, see [Section 2.1, “Provisioning a Virtual Machine,” on page 13](#).

The provisioning actions available from the right-click menu are as follows:

Table 2-2 *Right-Click VM Commands*

Action	Description
Provision	<p>Starts a VM to a running state. The Orchestrate Server automatically looks for the best VM host machine to run the VM on, unless you have specifically designated another server to run the VM.</p> <p>If a VM has snapshots, you cannot start the VM on a different host. If a VM that has snapshots is on a shared repository, you can register the VM to a different host and start the VM if the host is also connected to the shared repository.</p>
Pause	Prevents the VM from gaining access to the processor of the host machine, although it is still resident in the memory of the host machine.
Resume	Allows a paused VM to access the processor of the host machine again.
Suspend	<p>Pauses the VM and takes a snapshot of its disk and memory status. In the suspended state, a VM can be moved or migrated to another host machine.</p> <p>NOTE: A suspended VM must be provisioned to make it active again. The <i>Resume</i> action does not perform this function.</p>

Action	Description
Shutdown	Stops a VM from running, just like shutting down a physical machine. The operating system stops and acts as if it is shut down.
Restart	Shuts down and restarts a running VM.
Migrate	<p>vSphere: Migrates the VM from one host machine to another only if both the source and destination host machines have VMotion enabled. VM migrations can be of the following types:</p> <ul style="list-style-type: none"> ◆ A “warm migrate” is the migration of a suspended VM to another host. From a user’s perspective, if a VM is suspended, it is effectively “down.” This function requires shared storage. ◆ A “hot migrate” (also called a “live migrate”) is the migration of a running VM to another host and starting it there with minimal resulting downtime (measured in milliseconds). This function requires shared storage. <p>For more information, see the VMware VMotion documentation (http://www.vmware.com/products/vmotion/).</p> <p>Hyper-V: VM migration is not supported by PlateSpin Orchestrate.</p> <p>Xen: VM migrations can be of the following types:</p> <ul style="list-style-type: none"> ◆ A “hot migrate” (also called a “live migrate”) is the migration of a running VM to another host and starting it there with minimal resulting downtime (measured in milliseconds). This function requires shared storage. <p>NOTE: Migration of a Xen VM on Fibre Channel SAN disks is not supported.</p>
Resync State	Ensures that the state of the VM displayed in the Orchestrate Development Client is accurate.
Save Config	Requests that the runtime VM configuration be persisted on the VM image for future use.
	NOTE: If you add a VNIC or a vDisk to a VM, you must initiate the <i>Save Config</i> action. Clicking the <i>Save</i> icon on the Development Client toolbar does not save the change.
Apply Config	Updates the VM transient configuration. The VM must be running.
Create Template (VM only)	Creates a new template object based on the VM. Other versions of the VM can be cloned from this template. This menu item is replaced by the <i>Clone</i> menu item when you right-click a template VM.
Clone (VM template only)	Creates a standalone VM instance from the template but does not power on that instance. This action is unlike the <i>Provision</i> action from a template, which performs a <i>Clone</i> action, powers on the instance, then destroys the VM instance when it is powered off.
Delete/Destroy Resource	Removes a VM from the <i>Resources</i> list in the Orchestrate Development Client. If you want to delete the VM from the host machine, select the <i>Destroy VM Instance</i> option.

Action	Description
Move Disk Images	<p>A “move” is the relocation of VM disk images between two storage devices when the VM is in a not running state (including VMs that are suspended with a checkpoint file). This function does not require shared storage; the move is between separate repositories. Select the storage location from the drop-down menu.</p> <p>You can also move a VM from one VM host machine to another. This is a “cold” migration. VMware Server VMs must be migrated in this manner.</p> <p>If you want to move a VM of considerable size, appropriately increase the timeout fact value in the VM policy. The default value is 2400. For more information on editing the policy, see Chapter 4, “Configuring VM Provisioning Adapters and Discovered VMs,” on page 29.</p> <p>If a VM has snapshots, you cannot move the VM but you can register it to a different host if the VM and the host are connected to a shared repository.</p>
Checkpoint	<p>Creates a named snapshot of a VM image. This image is stored on the disk of the repository machine. Xen VMs cannot have a checkpoint applied to them.</p> <p>All the snapshots of a VM are chronologically listed in the <code>resource.vm.snapshots</code> fact, and the latest snapshot is listed in the <code>resource.vm.current_snapshot</code> fact.</p> <p>If the vSphere VM or the Hyper-V VM already has snapshots taken through other management consoles, the snapshots are synchronized with the latest snapshot taken through the Orchestrate Development Client, and are listed in the <code>resource.vm.snapshots</code> fact.</p>
Restore	<p>Starts a Checkpoint VM (that is, resumes the operations of a VM made into a stored checkpoint from the moment of storage).</p> <p>If the vSphere VM already has snapshots taken through other management consoles, the snapshots are synchronized with the latest snapshot taken through the Orchestrate Development Client, and are listed in the <code>resource.vm.snapshots</code> fact.</p>
Remove Template Dependency	Changes a cloned instance of a VM into a VM instance.
Install Agent	<p>Launches a job that automatically installs the Orchestrate Agent on a Xen-managed VM the next time you provision the VM.</p> <hr/> <p>IMPORTANT: If you stop or cancel a running Install Agent job, the VM is locked and you cannot provision the VM. The VM is automatically released after a period of time.</p>

Action	Description
Personalize	<p>Allows you to customize the VM. This includes changing elements like the DNS server. The changes are made to a VM that is shut down.</p> <p>This action relates to all guest OS-level changes, not hardware changes. With vSphere, guest customization occurs the next time the VM is powered on. The vSphere administrator must have properly configured the vCenter server to handle sysprep and the VMware tools must be installed to the VM.</p> <hr/> <p>IMPORTANT: If you stop or cancel a running Personalize job, the VM is locked and you cannot provision the VM. The VM is automatically released after a period of time.</p>
Shutdown Agent	<p>Shuts down the Orchestrate Agent and makes the VM unavailable as a resource.</p>
Create Virtual NIC	<p>Manually creates a vNIC on the VM to configure its network interface configuration by way of a vBridge connection.</p> <p>For more information, see “Creating Or Deleting a vNIC in the Development Client” in the <i>PlateSpin Orchestrate 2.5 Development Client Reference</i>.</p> <hr/> <p>NOTE: You can manually delete the new vNIC after it is created by right-clicking the vNIC object in the Explorer tree, then selecting <i>Delete</i>.</p>
Create Virtual Disk	<p>Manually creates a vDisk image of a specified size (measured in Mb) to associate to the VM.</p> <p>For more information, see “Creating Or Deleting a vDisk in the Development Client” in the <i>PlateSpin Orchestrate 2.5 Development Client Reference</i>.</p> <hr/> <p>NOTE: You can manually delete the new vDisk after it is created by right-clicking the vDisk object in the Explorer tree, then selecting <i>Delete</i>.</p>
Cancel Action	<p>Stops an action that has been requested.</p>
Check Host Assignment	<p>Opens a window so you can compare the VM hosts capable of hosting the VM.</p>
Launch Remote Desktop	<p>Launches a VNC terminal in which you can view and control the VM. Specify the credentials configured for the Web service in the appropriate VM policy.</p> <p>For more information, see Section 4.1.4, “Setting Up VNC for a VM Managed by vSphere,” on page 34.</p>
<p>TIP: For information about using the PlateSpin Orchestrate VM Client to perform many of these actions, see “Managing Virtual Machines” in the <i>PlateSpin Orchestrate 2.5 VM Client Guide and Reference</i>.</p>	

2.3.2 Prerequisites for Creating or Deleting vNIC and vDisk Objects on Hyper-V Managed Linux VMs

Although you can create vNICs and vDisks on Windows VMs managed by the Microsoft Hyper-V hypervisor, there is a prerequisite for creating or deleting these objects on a Linux VM managed by Hyper-V.

To make these actions work, it is necessary to install Linux Integration Components from Microsoft. For download and installation information, see the [Linux Integration Components page \(http://www.microsoft.com/downloads/details.aspx?FamilyID=c299d675-bb9f-41cf-b5eb-74d0595ccc5c&displaylang=en\)](http://www.microsoft.com/downloads/details.aspx?FamilyID=c299d675-bb9f-41cf-b5eb-74d0595ccc5c&displaylang=en) at the Microsoft downloads Web site.

2.3.3 Releasing a Virtual Machine from Usage

When the demand and load on your data center decreases, the Orchestrate Server analyzes the remaining resources and releases the most appropriate resource. If a VM meets the requirements of the remaining job demands better than a physical machine, the physical machine is released before the VM is released. This dynamic analysis allows you to make sure that the needs of your data center are met.

2.3.4 Managing Virtual Machine Templates

A VM template is a special kind of VM that is not deployed separately.

Review the following tasks to manage VM templates:

- ♦ “[Making a Virtual Machine Instance into a Template](#)” on page 21
- ♦ “[Changing a Virtual Machine Template Clone to an Instance](#)” on page 22
- ♦ “[Hyper-V VM Template Actions](#)” on page 22

This section also includes some information about support for the provisioning actions of a VM template created from a Hyper-V VM. See “[Hyper-V VM Template Actions](#)” on page 22.

Making a Virtual Machine Instance into a Template

- 1 In the Orchestrate Development Client, right-click the VM.
- 2 Select *Create Template*.
- 3 Name the template.
- 4 Specify a repository.
- 5 Specify a visible VM host.
- 6 Select a recommended host for the VMs to be launched on, if any are present.
- 7 Click *OK*.

When the instance of the template VM is provisioned, it appears as a sub-branch of the template’s location in the resources tree.

This clone functions as an instance of a VM and runs as though it were its own version with its own MAC address and other unique identifiers. The UUID is the only new information that is automatically generated for the clone. All the rest of the new information comes from *autoprep*, including the MAC address if an asterisk (*) is placed in the *Mac Address* field in the *Autoprep*

Network Adapter section of the *Info/Groups* tab for the template (the default is a blank field, meaning no MAC address is created), and if the *Use Autoprep* check box is enabled in the Create VM from *Template* dialog box.

Changing a Virtual Machine Template Clone to an Instance

- 1 If you decide to keep a clone VM, go to the PlateSpin Orchestrate Development Client, right-click it, and select *Remove Template Dependency*.

The Remove Template Dependency dialog box is displayed.

- 2 Click *OK*.

NOTE: This procedure works only when a VM instance has been cloned from a template. It does not work when a VM instance is provisioned from a template.

Hyper-V VM Template Actions

VM templates created in PlateSpin Orchestrate from discovered Hyper-V VMs have limited functionality. The following actions are supported in a Hyper-V VM template:

- ◆ Provision
- ◆ Move Disk Images
- ◆ Clone
- ◆ Delete/Destroy Resource
- ◆ Resync State

IMPORTANT: Even though the *Create Virtual NIC* and *Create Virtual Disk* actions appear to be legitimate actions on the right-click menu of a Hyper-V VM template (that is, they are not dimmed), these are not supported actions.

The following actions are not supported in a Hyper-V VM template:

- ◆ Save Config
- ◆ Checkpoint
- ◆ Restore
- ◆ Install Agent
- ◆ Personalize
- ◆ Create Virtual NIC
- ◆ Create Virtual Disk

2.3.5 Managed Virtual Machine Actions

You can perform many actions on the VM through the Orchestrate Development Client and the Orchestrate VM Client or you can write jobs to have actions performed on the VMs in your data center. The following table lists the managed VM actions that you can perform or use in a written job.

Table 2-3 *Managed VM Actions*

Action	Description
Provision	Starts a VM. This action clones and start a cloned VM template.
Clone	Creates a new, unique instance of a VM template.
Cold Migrate	Moves the storage location of the configuration and first disk files to another physical storage host. This might allow the VM to start faster.
Shutdown	Stops an active VM instance (including a started template VM). NOTE: If you manually shut down a running VM (either by logging in to the VM and shutting it down manually or by using any hypervisor-specific tool to shut down the VM), that VM is still displayed in the PlateSpin Orchestrate Client as running. If you then choose the <i>Shutdown</i> action rather than the <i>Resync State</i> action for that VM, the Orchestrate job log shows that the shutdown job fails. This condition always exists unless a Resync operation occurs on the VM or when an Orchestrate Agent on the VM relays its current state to the Orchestrate Server.
Delete/Destroy	Removes a VM from the <i>Resources</i> list in the Orchestrate Development Client. If you want to delete the VM from the host machine, select the <i>Destroy VM Instance</i> option.
Suspend	Takes a snapshot of an active VM and pauses it in order to move it to another VM host.
Pause	Prevents the VM from obtaining CPU cycles, but it stays resident.
Resume	Allows a paused VM to access the CPU again.
Create Template	Creates a VM template from a VM instance.
Hot Migrate	Changes the association of the VM, which is residing in a shared storage location, from one host machine to another.
Checkpoint	Create a named snapshot of a moment that can later be accessed to restart from the same point
Restore	Resumes a VM at a previously stored checkpoint.
Install Orchestrator Agent	Opens a VM image and installs the Orchestrate Agent.
Make Standalone	Removes the association of a template and makes the active VM into its own instance.
Check Status	Checks the current state of the VM to verify if the VM is provisioned or shut down.
Personalize	Modifies the Orchestrate Agent properties and disk image that are currently part of a clone.

Action	Description
Save Config	Transfers changes made to a VM to its permanent image storage.

2.3.6 Virtual Machine Technology-Specific Actions

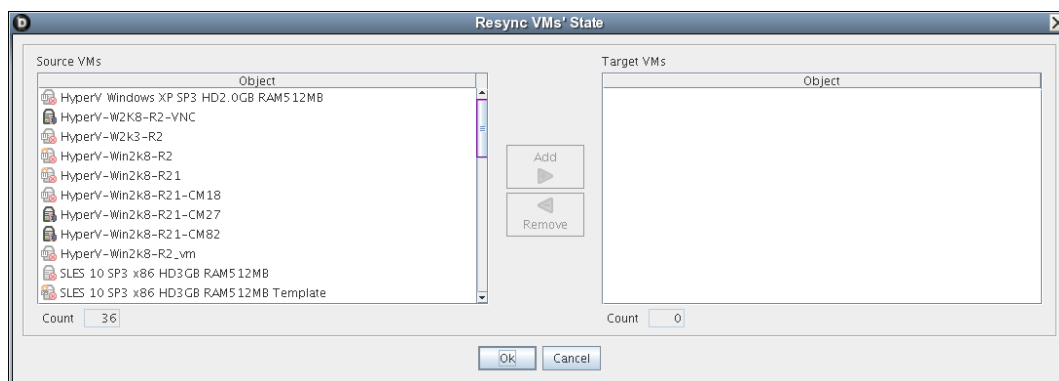
For a detailed breakdown of the actions you can perform on and with a VM, see the appropriate VM technology and configuration section in [Appendix A, “Virtual Machine Technologies and Actions,”](#) on page 79.

2.4 Resynchronizing the State of All VMs

To manually verify and ensure that the state of the VMs of all VM hosts displayed in the Orchestrate Development Client is accurate:

- 1 In the Orchestrate Development Client, click *Provision > Resync VM’s State*.

The Resync VMs’ State dialog box is displayed.



- 2 In the *Source VMs* pane, select the VMs to be resynchronized, then click *Add*.

The selected VMs are added to the *Target VMs* pane.

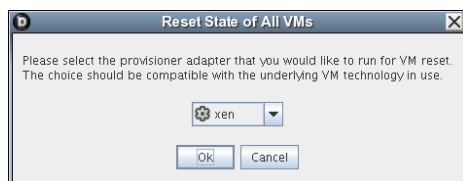
- 3 Click *OK*.

2.5 Resynchronizing the State of All VMs of a Specific VM Host

To manually verify and ensure that the state of the VMs of a specific VM host displayed in the Orchestrate Development Client is accurate:

- 1 In the Orchestrate Development Client, click *Provision > Reset State of All VMs*.

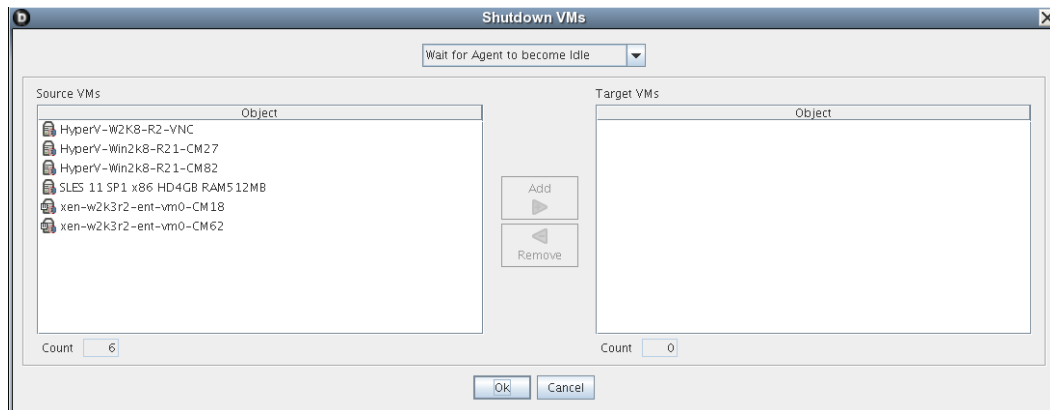
The Reset State of All VMs dialog box is displayed.



- 2 Select the VM host whose VMs you want to resynchronize.
- 3 Click *OK*.

2.6 Shutting Down Multiple VMs

- 1 In the Development Client, click *Provision > Shutdown VMs*.
The Shutdown VMs dialog box is displayed.
- 2 You can choose to shut down the VMs after the Orchestrate Agent becomes idle or to immediately shut down the VMs. By default, the *Wait for Agent to become Idle* option is selected.
- 3 In the *Source VMs* pane, select the VMs you want to shut down, then click *Add*.



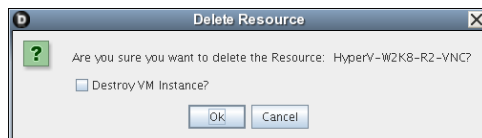
The selected VMs are added to the *Target VMs* pane.

- 4 Click *OK*.

2.7 Destroying and Deleting a Virtual Machine

- 1 In the PlateSpin Orchestrate Development Client, right-click the VM in the tree and select *Delete/Destroy Resource*.

The Delete Resource dialog box is displayed.



- 2 (Optional) To delete a VM from the VM host, select the *Destroy VM Instance* option.
This completely deletes the VM and all its versions from your data center. You cannot restore any version of the VM after you delete it.
If you do not choose this option, the VM is removed from the resource list. However, the actual image of the VM is still stored in its directory.
- 3 Click *OK*.

If you choose only to delete a VM from your resource tree, you can rediscover the VM by running a discovery job (click *Provision > Discover VM Images*).

Managing VM Repositories

PlateSpin Orchestrate uses the Repository object to represent where VMs are stored. VMs can be stored on local disks, the Orchestrate datagrid, a network attached storage (NAS), or a storage area network (SAN, Fibre Channel, or iSCSI).

Before VMs can be used by PlateSpin Orchestrate, you must create or discover Repository objects and then discover the VM images within the Repository:

- ♦ [Section 3.1, “Provisioning a VM from a Local Repository,” on page 27](#)
- ♦ [Section 3.2, “The Datagrid Repository,” on page 28](#)
- ♦ [Section 3.3, “Provisioning a VM from a NAS Repository,” on page 28](#)
- ♦ [Section 3.4, “The SAN Repository,” on page 28](#)

3.1 Provisioning a VM from a Local Repository

By default, the Xen and vSphere provisioning adapters create a local Repository object for local VM images (or datastores) on a VM host when PlateSpin Orchestrate performs the Discover VM Hosts action.

NOTE: For vSphere, PlateSpin Orchestrate models all repositories the Virtual Center server is aware of. There is not necessarily a local repository for each ESX host, although that is the default.

A local repository represents VMs residing in a VM Host’s local storage where the VMs are only visible to the VM Host. VMs are searched for in the default paths for each adapter.

IMPORTANT: Do not use local repositories for shared directories visible to more than one VM Host. Instead, create a new NAS or SAN repository.

For information on NAS storage, see [“Provisioning a VM from a NAS Repository” on page 28](#). For information on SAN storage, see [“The SAN Repository” on page 28](#).

When discovering VM Images, the adapters use the `location`, `searchpath` and `preferredpath` facts for searching. The `repository.location` is usually the root path, such as `/`. For Xen, the adapter creates a local repository with search paths of `/etc/xen/vm` and a preferred search path of `/var/lib/xen/images`.

When the Discover VM Images action is run, the [provisioning adapter](#) follows these steps:

- ♦ Concatenates the `repository.location` and every element of `repository.searchpath` and searches for VMs in those directories.
- ♦ Concatenates the `repository.location` and `repository.preferredpath` and searches for VMs in that directory.

These steps are also followed when searching in NAS and SAN repositories when representing auto-mounted file systems, and when the location, search path, and preferred path are set.

For more information on facts, see [“Understanding Grid Object Facts and Computed Facts”](#) in the *PlateSpin Orchestrate 2.5 Developer Guide and Reference*.

3.2 The Datagrid Repository

By default, a datagrid repository named `zos` is automatically created when the PlateSpin Orchestrate Server is initialized. The datagrid repository represents VMs residing in the PlateSpin Orchestrate datagrid, which is a storage area on the Orchestrate Server.

The `zos` datagrid repository has a location of `grid:///vms`, which points to an area in the datagrid reserved for VM archival storage.

The datagrid repository storage is archival because VMs cannot be run from the datagrid repository. You must move VMs out of the datagrid to a VM Host in order to run them.

3.3 Provisioning a VM from a NAS Repository

The Network Attached Storage (NAS) repository represents VMs stored on a NAS. This is a storage area where VMs are visible to multiple VM hosts, so they can be run by any one of the connected hosts.

The following procedure shows an example of setting up a NAS repository. For the example, assume you have a Xen setup where the `/vms` directory is auto-mounted on multiple VM hosts as the shared storage location for your VMs.

- 1 To create a new Repository object, go to the [PlateSpin Orchestrate Development Client](#), then click *Actions > Create Repository*.
- 2 Specify a new name and choose which adapter group this repository is used for.
This example is for Xen VMs, so choose the `xen30` adapter.
- 3 Close the dialog box to display the *Info/Groups* tab for the new repository.
- 4 Set the location path.
This is the root path for the repository. It is usually `/`.
- 5 Set the search path and preferred path.
In this example, the VMs are all in `/vms`, so leave `searchpath` empty and set the `preferredpath == "vms"`.
- 6 Select the VM Host objects that have visibility to the shared directory and add the new repository to the VM hosts list of available repositories.
To find a VM host, either select *VM Hosts* to open the VM Hosts admin view or open the *Physical* tree under *Resources* and open the physical host representing the VM host.
- 7 Run *Provision > Discover VM Images* on the new repository.

3.4 The SAN Repository

The Storage Area Network (SAN) repository is a single storage server that can be accessed by multiple machines. PlateSpin Orchestrate 2.x does not support booting a VM from a SAN repository. SAN repositories can only be used as data disks for VMs.

Configuring VM Provisioning Adapters and Discovered VMs

4

This section includes the following information:

- ♦ [Section 4.1, “Configuring the VMware vSphere Provisioning Adapter,” on page 29](#)
- ♦ [Section 4.2, “Configuring the Xen Provisioning Adapter and Xen VMs,” on page 44](#)
- ♦ [Section 4.3, “Configuring the Amazon EC2 Provisioning Adapter,” on page 45](#)
- ♦ [Section 4.4, “Configuring the hyperv Provisioning Adapter and Hyper-V VMs,” on page 48](#)

4.1 Configuring the VMware vSphere Provisioning Adapter

This section includes the following information:

- ♦ [Section 4.1.1, “Initial Configuration,” on page 29](#)
- ♦ [Section 4.1.2, “Configuring Other Policies for vSphere,” on page 32](#)
- ♦ [Section 4.1.3, “Assigning a vSphere VM to a Resource Pool,” on page 33](#)
- ♦ [Section 4.1.4, “Setting Up VNC for a VM Managed by vSphere,” on page 34](#)
- ♦ [Section 4.1.5, “Setting Up Orchestrate to Accommodate VMware DRS Clustering and Updates,” on page 36](#)
- ♦ [Section 4.1.6, “Discovering Enterprise Resources in Multiple vSphere Environments,” on page 40](#)
- ♦ [Section 4.1.7, “Known Configuration Limitations for Virtual Center VMs,” on page 43](#)

4.1.1 Initial Configuration

Before you can provision and manage VMs with the vsphere provisioning adapter, you must perform some initial steps to configure it in order to get it running.

- 1** Make sure that the Orchestrate Agent is installed and started on a supported host (that is, a Windows resource or a Linux resource).

For more information, see [“Installing the Orchestrate Agent Only”](#) in the *PlateSpin Orchestrate 2.5 Installation and Configuration Guide*.

We recommend that you do not install the Orchestrate Agent on a machine running the vCenter Server. This will allow the server to function without additional Orchestrate load.

- 2** In the PlateSpin Orchestrate Development Client, log in to the grid that manages vSphere VMs, then in the Explorer tree, select *vsphere*.
- 3** In the Policy Editor admin view, modify `vsphere.policy`.
 - 3a** Search for the `vcenters` fact name.

```

    <fact name="vcenters">
      <dictionary>
        <dictelement key="DEFAULT">
          <dictionary>
            <dictelement key="webservice_url" value="https://
/sampl-vserver.zenworks.lab.novell.com/sdk"/>
            <!--This fact is used to determine which stored
credential will be used for access-->
            <!--to the vsphere webservice. By default, this
value is 'vsphere_<name of the vcenter key>'-->
            <!--A credential with this name must be added
to the credential manager before any provisioning-->
            <!--actions will be able to complete
successfully.-->
            <!--<dictelement
key="webservice_credential_name" value="vsphere_DEFAULT"/>
            <dictelement key="auto_portgroup_creation"
type="Boolean" value="False"/>
            <dictelement key="auto_portgroup_disconnection"
type="Boolean" value="False"/>
            <dictelement key="auto_portgroup_deletion"
type="Boolean" value="False"/>
          </dictionary>
        </dictelement>
      </dictionary>
    </fact>

```

The `job.vcenters` fact defines the configuration for connecting multiple vCenter Servers to PlateSpin Orchestrate. If you have only one vCenter Server, the `DEFAULT` dictionary element is all that is necessary to connect to the server.

- 3b** Set the `webservice_url` fact to either the IP address or DNS name of the VCenter Server.
- 3c** (Conditional) If you plan to manage the VMs of more than one vCenter Server, copy the `DEFAULT` section of the policy to represent each additional server, then rename the keys. For example:

```

<fact name="vcenters">
  <dictionary>
    <dictelement key="SAMPL-VSERVER">
      <dictionary>
        <dictelement key="webservice_url" value="https://
/sampl-vserver.zenworks.lab.novell.com/sdk"/>
        <!--This fact is used to determine which stored
credential will be used for access-->
        <!--to the vsphere webservice. By default, this
value is 'vsphere_<name of the vcenter key>'-->
        <!--A credential with this name must be added
to the credential manager before any provisioning-->
        <!--actions will be able to complete
successfully.-->
        <dictelement key="webservice_credential_name"
value="vsphere_SAMPL-VSERVER"/>
        <dictelement key="auto_portgroup_creation"
type="Boolean" value="False"/>
        <dictelement key="auto_portgroup_disconnection"
type="Boolean" value="False"/>
        <dictelement key="auto_portgroup_deletion"
type="Boolean" value="False"/>
      </dictionary>
    </dictelement>
  </dictionary>

```

```

        </dictelement>
        <dictelement key="SAMP2-VSERVER">
            <dictionary>
                <dictelement key="webservice_url" value="https://
/some-other-dns/sdk"/>
                <!--This fact is used to determine which stored
credential will be used for access-->
                <!--to the vsphere webservice. By default, this
value is 'vsphere_<name of the vcenter key>'-->
                <!--A credential with this name must be added
to the credential manager before any provisioning-->
                <!--actions will be able to complete
successfully.-->
                <dictelement key="webservice_credential_name"
value="vsphere_SAMP2-VSERVER"/>
                <dictelement key="auto_portgroup_creation"
type="Boolean" value="False"/>
                <dictelement key="auto_portgroup_disconnection"
type="Boolean" value="False"/>
                <dictelement key="auto_portgroup_deletion"
type="Boolean" value="False"/>
            </dictionary>
        </dictelement>
    </dictionary>
</fact>

```

IMPORTANT: The key value must be the actual name of the VCenter server. The value is case-sensitive.

- 4 In the Explorer tree, select the Orchestrate Server or “grid” object, then select the *Authentication* tab in the admin view to open the Authentication page.
- 5 On the Authentication page, scroll to the Credential Manager (consisting of the *Stored Credentials* panel and the *Stored Certificates* panel), then click *Add Credential* to display the Edit Credential dialog box.
 - 5a Fill in the fields of the dialog box:
 - ♦ **Name:** Enter a value in format `vsphere_<vCenterId>`. If you used the `DEFAULT` vCenter in the `vsphere.policy` (see [Step 3](#)), you can set this value to `vsphere_DEFAULT`.
 - ♦ **User:** Enter the username used to connect to the vCenter Web service.
 - ♦ **Secret:** Enter the the password of the vCenter Web service user.
 - ♦ **Type:** (Optional) Enter any string that lets you categorize similar credentials into a category or group. For example, for the vsphere provisioning adapter you might enter a “type” called `vsphere`.

For more information, see “[The Credential Manager](#)” in the *PlateSpin Orchestrate 2.5 Development Client Reference*.
- 6 In the Explorer tree, select the PlateSpin Orchestrate resource (or Resource group) that can see the vCenter Server, then select the *Policies* tab to open the Policies page in the admin view.
- 7 On the Policies page, click *Choose* to open the Policy Selection dialog box, then in the *Source Policies* list, select the `vsphere_client` policy, click *Add*, then click *OK* to associate this policy with this resource (or Resource group). For more information, see “[The Resource Policies Page](#)” in the *PlateSpin Orchestrate 2.5 Development Client Reference*.

When the vsphere provisioning adapter job starts, this policy constrains one of these resources for the job to run the Web service commands.

NOTE: If you want to connect multiple vCenter Servers, make sure you modify the `vsphere.vcenters` fact of the `vsphere_client.policy` as described in the policy comments.

4.1.2 Configuring Other Policies for vSphere

The following table provides detailed information about other policies associated with the vSphere provisioning adapter that are used to manage the vSphere hosts and the VMs in the grid. The policy settings are applied to all the VMware VMs in the grid.

Table 4-1 Virtual Machine Management Policies for vSphere

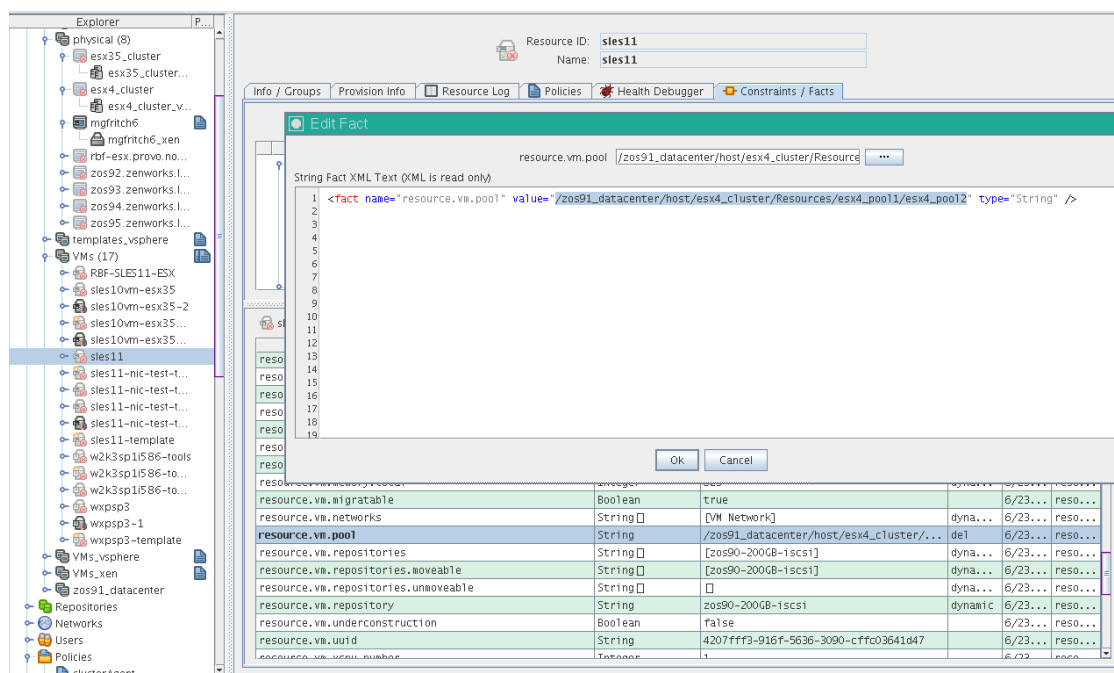
Policy Name	Description	Additional Details
vsphere	Contains the constraints used to select the vCenter Server resources.	Modify this policy with authentication information for the vCenter server and follow instructions in the policy for creating the credential store.
vsphere_assignPool	If you need to assign the VMs to a certain cluster (for example, a cluster root pool), or if you want to assign VMs to pools “owned” by your customers, use this policy.	When applied this policy allows the VM to reside only on VM hosts that have access to the assigned resource pool (<code>resource.vm.pool</code>).
vsphere_client	Contains the settings used to run the vsphere job on the associated vSphere resource.	You need to associate the vsphere_client policy to a vSphere resource before the discovery works. For more information, see Step 7 on page 31 .
vsphere_ignoreNetwork	Includes special facts that allow VMs to consider a VM host despite a missing required Network.	If ignoreNetworkCheck is set, a vbridge (portgroup) can be dynamically created on a VM power-on event. This works in conjunction with the auto_portgroups_creation fact found in the vsphere.policy. Make sure that you set the auto_portgroups_creation fact to true or else the portgroup will not be created during the VM power-on event.
vspheredaemon	Includes settings applicable to daemon job. For tuning the daemon, idle timeout period, or debug on or off.	We recommend that the values for the facts within the policy stay unchanged. For advanced users tuning a server, you can change the maxConcurrentOperations fact default value higher if you consistently run more than 30 actions concurrently.
vsphereJobDefault	Negates the default constraint on provisioning adapters to run only on physical resources.	This policy is used for PlateSpin Orchestrate 2.0.2 compatibility.

Policy Name	Description	Additional Details
vspherePA	Includes the basic constraints for the vsphere provisioning adapter.	Do not modify this policy.
vSphereUpdate	Includes settings for the vsphereUpdateDaemon job. The policy can be modified directly or the user can edit job args in the schedule that is created by default installation of Orchestrate.	For more information, see “Configuring the vSphere Update Client” on page 37.
vsphereVnc	Includes a setting to allow remote desktop connections to vSphere VMs.	For more information, see Section 4.1.4, “Setting Up VNC for a VM Managed by vSphere,” on page 34.

4.1.3 Assigning a vSphere VM to a Resource Pool

All VMs managed by vSphere are assigned to a resource pool. When vSphere VM images are discovered by PlateSpin Orchestrate, the `resource.vm.pool` fact for each VM is set with what is known by vSphere as a “pool assignment.”

Figure 4-1 A vSphere Resource Pool Discovered in the PlateSpin Orchestrate Development Client



In this example, the VM `sles11` is contained within the `esx4_pool2` pool, which is a sub-pool of the `esx4_pool1` pool, which is also a sub-pool of the `esx4_cluster` cluster root resource pool. The resource pool hierarchy in this example is roughly:

```
esx4_cluster > esx4_pool1 > esx4_pool2
```

By default, every VMware cluster contains a resource pool known as the “root” resource pool. The root resource pool is usually in a vCenter inventory path similar to this:

/zos91_datacenter/host/esx4_cluster/Resources

If you do not need to restrict VMs based on resource pool assignment, then no policy configuration is necessary and you can provision the VMs as usual, but if you need to assign the VMs to a certain cluster (for example, the cluster root pool), or if you want to assign VMs to a pools “owned” by your customers, you can configure the `vsphere_assignPool` policy to accomplish this.

Use the following steps to assign a VM to a resource pool, for example to disallow PlateSpin Orchestrate from provisioning the VM to another resource pool:

- 1 Assign the `vsphere_assignPool` policy to the VM or a group of VMs. No changes to the actual policy file are necessary.

During provisioning of the VM, Orchestrate verifies and relocates the VM (as necessary) to maintain the validity of the pool assignment.

- 2 (Conditional) If the VM does not reside in the correct resource pool, modify the `resource.vm.pool` fact to reflect the correct pool assignment. Orchestrate relocates the VM to the specified resource pool at the next provision.

4.1.4 Setting Up VNC for a VM Managed by vSphere

When you [right-click a VM resource](#), you have the option of launching a remote virtual network computing (VNC) session console of that VM’s desktop. This section provides information about setting up PlateSpin Orchestrate to accommodate a VNC session for a VM.

ESX servers managed by vSphere might have a firewall in place to protect some ports from being open or closed. The `vsphere` provisioning adapter opens the appropriate ports to accommodate VNC connections from a remote console. These ports are opened when PlateSpin Orchestrate discovers the servers.

Use these steps to set up VNC session connectivity for the VM managed by the `vsphere` provisioning adapter.

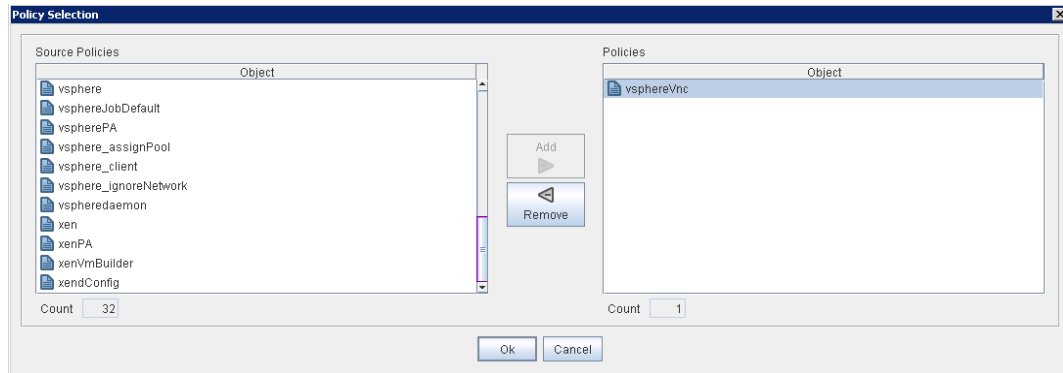
NOTE: Although you can change these settings at any time, they take effect in vSphere only when the VM is not running, such as at provision time.

- 1 In the Explorer tree of the PlateSpin Orchestrate Development Client, select the Grid Server object where you are logged in, then select the *Authentication* tab to open the server’s authentication page.
- 2 In the *Stored Credentials* panel (also known as the “Credential Manager”) of the Authentication page, click *Add Credential* to open the Add Credential dialog box.
- 3 In the Add Credential dialog box, specify a credential that includes the VNC password you want to use, then click *Add*.

In the next steps, the VNC consoles for one or more of the `vsphere` VMs in this grid are configured to use the password from this credential. For more information about adding a credential, see “[The Users Panel](#)” in “[The Orchestrate Server and the Server Admin Objects](#)” in the *PlateSpin Orchestrate 2.5 Development Client Reference*.

- 4 In the Explorer tree, expand the Policies folder to display the list of policies, then select *vsphereVNC* policy to open the Policy Editor.
- 5 In the Policy Editor, modify the `vnc.credential` fact value to be the name of the credential you created in [Step 3](#), then click the *Save* icon.

- 6 In the Explorer tree, select the VM Resource Group (or an individual VM) managed by the vsphere provisioning adapter, then in the admin view, select the *Policies* tab to open the Policies page for this group.
- 7 On the Policies page, select *Choose* to open the Policy Selection dialog box.



- 8 In the *Source Policies* list of the Policy Selection dialog box, select the *vsphereVnc* policy, click *Add* to move it to the associated *Policies* list, then Click *OK*.
- 9 On the Development Client Menu Bar, click *Provision > Discover VM Hosts and Repositories*. This action opens or closes the firewall on the VM hosts to allow VNC access. This access is based on the *vsphere.openVncFirewallPort* fact in the *vsphere* policy.
- 10 (Conditional: VM must be shut down) From the Explorer tree, right-click a vSphere-managed VM, then select *Provision*.
If the VM for which you want to open a VNC session is running, you need to shut it down, then reprovision it.
- 11 (Conditional) If the vSphereUpdate Client is running for your vCenter server, refresh the Orchestrate Client.
or
(Conditional) If the vSphereUpdate Client is not running for your vCenter server, right-click the VM object and select *Resync State*.
If you don't want to resync before using the VNC console, make sure you configure the vSphere Update Client beforehand. For more information, see [“Configuring the vSphere Update Client” on page 37](#).
- 12 Right-click the VM object and select *Launch Remote Desktop* to open the login dialog box for the VNC session.
- 13 In the login dialog box, enter the VNC password that you created in the Credential Manager in [Step 3](#).

NOTE: If you migrate a VM you previously configured and provisioned for VNC availability from one ESX host to another ESX host, it is possible that the port you assigned to the VM for firewall access might be in use by the new ESX host.

If the `vnc.mode` fact in the `vsphereVnc` policy is set to `automatic` and the VM is restarted (or if the process currently occupying the port is restarted), PlateSpin Orchestrate chooses another port for the VM. Because the number of ports available for VNC console use is limited if you are migrating multiple vSphere VMs, you might have to reprovision the VMs with new port numbers if you discover a conflict.

vSphere VNC Facts

The following table lists the VNC-related facts in the `vsphere` provisioning adapter and provides a description of each of those facts.

Table 4-2 *vSphere VNC Facts*

Fact Name	Description
<code>resource.vnc.ip</code>	The IP address of the VM host where the VM is running
<code>resource.vnc.port</code>	The port currently assigned to the VM. The value is -1 if VNC is disabled for the VM.
<code>resource.vnc.credential</code>	The credential containing the VNC password. This is the name of the credential itself, not the username or the password contained in the credential.
<code>resource.vnc.mode</code>	Determines how VNC port assignments are handled. This value must be <code>automatic</code> , <code>manual</code> , or <code>off</code> . <ul style="list-style-type: none"> ◆ If <code>mode = automatic</code>: Orchestrate attempts to select the next available VNC port. ◆ If <code>mode = manual</code>: The port value specified in the VM's <code>resource.vnc.port</code> fact is used. ◆ If <code>mode = off</code>: The VNC console is disabled.
<code>resource.remotedesktop</code>	Controls enabling or disabling the Launch Remote Desktop action in the Development Client.

4.1.5 Setting Up Orchestrate to Accommodate VMware DRS Clustering and Updates

PlateSpin Orchestrate supports the discovery of VMware vSphere clusters used for high availability in a VMware environment or managed by the VMware Distributed Resource Scheduler (DRS) after an Orchestrate Agent has been deployed into such an environment. In this scenario, Orchestrate also allows you to verify when actions have taken place outside of Orchestrate, such as when DRS moves a VM to an alternate host in the cluster or when an administrator moves a VM into a different resource pool.

Any vSphere clusters discovered by PlateSpin Orchestrate and managed by DRS are listed in the Development Client as members of a convenience group (for example, a group named `clusters_vsphere`).

You can learn about the read-only cluster-related facts for these discovered clusters in the following PlateSpin Orchestrator documentation references:

- ◆ “Orchestrator Facts in the VM Host Cluster Object” in the *PlateSpin Orchestrator 2.5 Development Client Reference*
- ◆ “Orchestrator Facts in a VM Host Residing in a Cluster” in the *PlateSpin Orchestrator 2.5 Development Client Reference*
- ◆ “Orchestrator Facts in VMs Hosted in Clusters” in the *PlateSpin Orchestrator 2.5 Development Client Reference*

The Orchestrator update infrastructure consists of two main components:

- ◆ A **vSphere Update Client** component, which is contained in the PlateSpin Orchestrator Agent
- ◆ The **vSphereUpdate monitor job**, which starts the client component and ensures that it runs when necessary

Other information in this section that you can use to update the Orchestrator vSphere provisioning adapter for clustering includes the following:

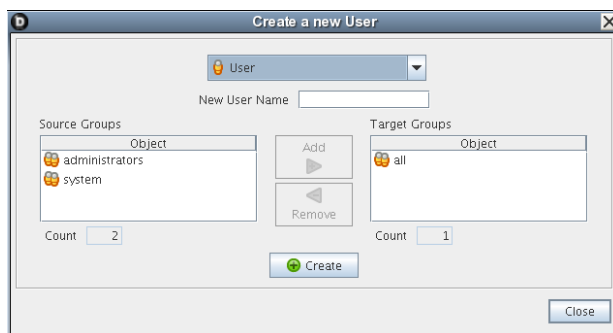
- ◆ “Configuring Orchestrator to Limit Datastore Visibility in vSphere Clusters” on page 39

Configuring the vSphere Update Client

To configure the vSphere Update Client:

1 Create a proxy user:

- 1a** In the Orchestrator Development Client, click *Actions > Create User* to open the Create a New User dialog box.



- 1b** In the *Source Groups* list, select administrators, then click *Add* to move the administrators user group to the *Target Group* list.

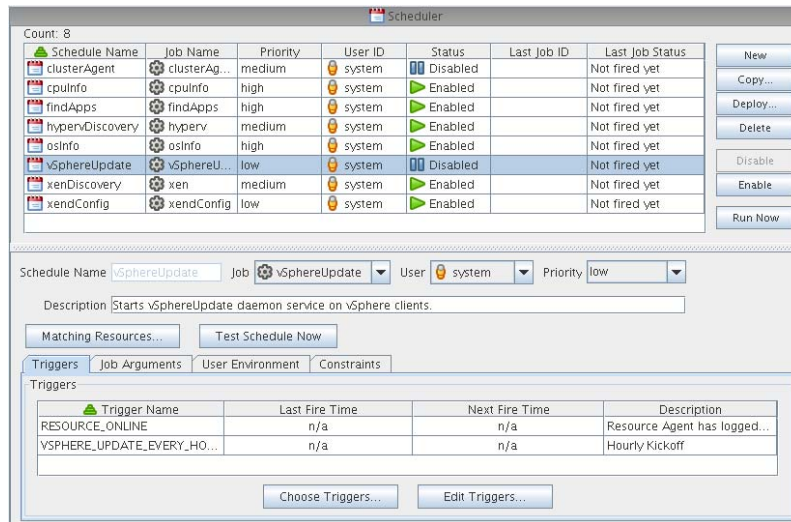
- 1c** In the *New User Name* field, specify a user name, click *Create*, then click *Close*.

This is the proxy user. The username must contain the word “proxy,” for example, *my_proxy*, or *proxy1*.

- 2** Modify the vSphereUpdate.policy (or modify the jobargs in the scheduler) so that `zos.proxy.user` contains the name of the user created in [Step 1](#):

- 2a** In the Explorer Tree, select the *Policies* group to expand the list of policies included on this grid.

- 2b** Select the *vSphereUpdate* policy to open the Policy Editor view.
- 2c** Find the `zos.proxy.user` fact in the policy, then specify the name of the proxy user you created in [Step 1c on page 37](#) as the value for this fact.
- 3** Run the *vSphereUpdate* schedule and job:
 - 3a** In the toolbar of the Development Client, select *Scheduler* to open the Orchestrate Job Scheduler.



- 3b** Select the *vSphereUpdate* schedule, click *Enable*, then click *Run Now*.

The vSphereUpdate Monitor Job

The *vSphereUpdate* monitor job is located in the “all” jobs group. It is associated with both the *vsphere* policy (for VCenter configuration information) and the *vSphereUpdate* policy. The *vSphereUpdate* policy specifies the following cluster-related facts. You can modify these facts to accommodate your environment.

Table 4-3 Cluster-Related Facts in the *vSphereUpdate* Policy

Fact Name	Type	Description
<code>jobargs.zos.proxy.user</code>	String	An administrative user used by the Orchestrate Development Client to log in to the Orchestrate Server in order to perform update operations there. You must create an administrative user for this purpose, if you have not already done so. The name of this user must contain the word “proxy,” for example, <i>my_proxy</i> , or <i>proxy1</i> . When you change the value of this fact, you must restart the Orchestrate Server. For information about configuring the vSphere Update Client, see “Configuring the vSphere Update Client” on page 37 .

Fact Name	Type	Description
jobargs.zos.proxy.passwd_validity	Integer	<p>The amount of time (measured in seconds) that the <code>zos.proxy.user</code> password is valid.</p> <p>Example: 86400 (1 day)</p> <p>Although the default value (-1) implies that the password is valid forever, the actual validity time is limited to the uptime of the Orchestrate Server.</p> <p>When the password expires, the Development Client is automatically restarted with a new password the next time that the monitor job runs.</p>
jobargs.debug	Boolean	<p>Specifies whether you want extra verbose debug logging sent to a job log.</p> <hr/> <p>NOTE: The client logs its output to the <code>log.txt</code> and <code>err.txt</code> files located in <code><agent_install_dir>/node.default/.vSphereUpdate/<hostname>/<vcenterId></code>.</p>
jobargs.verbose	Boolean	<p>Specifies whether you want verbose logging sent to a job log.</p> <p>This fact is implicitly set when <code>jobargs.debug</code> is set.</p>
jobargs.mode	String	<p>The value for this fact can be optionally set to "clear." This resets the <code>passwd_validity</code> and forces a restart on the next invocation where the mode is not set.</p> <p>The value can also be set to "stop" to stop all running update clients.</p>

Configuring Orchestrate to Limit Datastore Visibility in vSphere Clusters

If you want to limit the number of datastores (that is, Orchestrate Repositories) that are available to a vSphere cluster, you can assign a policy similar the policy below to the undesired repository or repositories:

```
<policy>
  <repository>
    <fact name="enabled" type="Boolean" value="False" />
    <fact name="provisioner.jobs">
      <array type="String">
      </array>
    </fact>
  </repository>
</policy>
```

This disables the repository for use with the cluster.

4.1.6 Discovering Enterprise Resources in Multiple vSphere Environments

A data center administrator running VMware products might organize the virtual resources in his or her enterprise into several different vSphere environments. PlateSpin Orchestrate lets you discover and manage all of these enterprise VMs, discovering each relevant VM host, network, repository, and VM within the several vSphere environments and modeling them as objects in the Orchestrate Development Client.

The discovery is facilitated through a PlateSpin Orchestrate Agent that acts as a proxy for the VMware enterprise to provide information about the resources of each vSphere environment. Policies defined and configured on the proxy agent enable communication between each vSphere environment and the established PlateSpin Orchestrate environment.

After the resources of all the vSphere environments are discovered, the provisioning actions available for the discovered objects behave correctly without the need to create new vSphere jobs, policies, or constraints.

This section includes the following information:

- ♦ [“Setting Up an Orchestrate Proxy Agent in the vSphere Environment” on page 40](#)
- ♦ [“Running Discovery” on page 42](#)

Setting Up an Orchestrate Proxy Agent in the vSphere Environment

For PlateSpin Orchestrate to support virtual resource management in multiple vSphere environments, you need to deploy a PlateSpin Orchestrate Agent to a server in one of those environments. The server is then referred to as the “agent host.” For more information, see [“Installing the Orchestrate Agent Only”](#) in the *PlateSpin Orchestrate 2.5 Installation and Configuration Guide*.

This section includes the following information:

- ♦ [“Installing the Proxy Agent on the vCenter Server” on page 40](#)
- ♦ [“Configuring a Proxy Agent outside of the vSphere Environment” on page 41](#)

Installing the Proxy Agent on the vCenter Server

It is not necessary to install the agent on the machine where the vCenter Server is running. In fact, we recommend that you do not install the Orchestrate Agent on this server because the agent’s potential processing needs might put an unsatisfactory load on the vCenter Server. If your circumstances require you to install the agent on the vCenter Server, however, there are some things you need to consider:

- ♦ If you want the agent on the server to forego processing of vSphere provisioning adapter (PA) joblets, do not associate the `vsphere_client` policy from the object in the Development Client. Verify that the value of the `resource.vsphere.client` fact is false.
- ♦ If your vCenter Server is running behind a Windows firewall, you need to configure the firewall to allow connections from remote machines to port 443 on the server.
- ♦ If you are using Windows Server 2008 as your vCenter Server platform, remember that VMware adds a firewall rule to allow connections, but only on “public” network types. If you are connecting from a “private” network type, you need to change the firewall rule.

Configuring a Proxy Agent outside of the vSphere Environment

Use the following steps to configure a PlateSpin Orchestrate resource to act as a proxy for the Orchestrate Agent running in the VMware enterprise.

- 1 (In the vSphere environment) Find the vSphere public certificate on the machine running the vCenter Server. The location of the certificate depends on the server platform:
 - ♦ **Windows Server 2008:** The path to the public key is `C:\ProgramData\VMware\VMware VirtualCenter\SSL\rui.crt`.
 - ♦ **Windows Server 2003:** The path to the public key is `C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\SSL\rui.crt`.
- 2 (In the vSphere environment) Copy the public certificate to an accessible directory on the agent host (see [“Setting Up an Orchestrate Proxy Agent in the vSphere Environment” on page 40](#)). You can use FTP, copy and paste, or access from a shared directory.
- 3 (In the vSphere environment) Install the public certificate on the agent host, outside of the agent JVM.

The agent JVM has a built-in keystore that includes a default setup to accept certain SSL certificates. Because the vSphere SSL certificate is usually self-signed, it is not accepted by the JVM by default.

- 3a On the agent host, change to the C: drive.

```
cd c:\
```

- 3b Create the `\sslCert` directory.

```
mkdir c:\sslCert
```

- 3c Import the certificate to the new directory.

```
c:\Program Files\Novell\ZOS\Agent\jre\bin\keytool.exe -importcert -file  
\<path_to_rui.crt> -keystore C:\sslCert\client.keystore
```

- 3d When prompted, enter a password for the new keystore.
- 3e When prompted, enter `yes` to indicate that you trust the certificate.

TIP: When the `client.keystore` file has been created on the agent host of one vSphere environment, you can copy it to the agent host in every other vSphere environment of the enterprise.

- 4 (In the Orchestrate environment) In the Orchestrate Development Client, select a resource in the Explorer tree that you want to act as the proxy agent for the VMware enterprise.
- 5 (In the Orchestrate environment) Modify the `vsphere` policy associated to the proxy agent:
 - 5a In the Explorer view, expand the *Policies* group, then select the `vsphere` policy to open the *Policy Editor* page.
 - 5b In the *Policy Editor* page, find the `vcenters` fact name and the `<dictionary></dictelement>` section of that fact, then modify it to identify the Web service credential store location of a discrete vSphere environment:
 - 5b1 Change the first dictelement key from `DEFAULT` to the exact name of the vCenter Server.
 - 5b2 Change the value of the `webservice_url` dictelement key from `http` to `https` and `localhost` to the appropriate DNS name or IP address of the vCenter server.

For example, change `http://localhost/sdk`
to
`https://10.10.10.10/sdk`

- 5c** (Optional) Copy the text of the entire `<dictionary>` component of the `vcenters` fact, then paste it directly following the text of the prior `<dictionary>` element. Repeat this for each vSphere environment you want to connect to and whose resources you want to discover, then execute [Step 5b](#) on each new `<dictionary>` component you created for the fact.
- 6** (In the Orchestrate environment) Add a new fact to the `vsphere_client` policy to facilitate SSL communication between PlateSpin Orchestrate and the vSphere environment:
 - 6a** In the Explorer tree, expand the *Policies* group, then select the `vsphere_client` policy to open the Policy Editor page.
 - 6b** On the Policy Editor page, locate and then uncomment the following section of code:

```
<!--  
<fact name="vsphere.trustStore"  
      type="String"  
      value="c:\sslCert\client.keystore"  
      description="Location on this resource of the trust store  
(required for SSL)" />  
-->
```

Make sure that the value string in the code matches the path to the keystore on your selected proxy agent (see [Step 3c](#)).
 - 6c** Save your changes in the `vsphere_client` policy.
- 7** (In the Orchestrate environment) Associate the `vsphere_client` policy to the proxy agent:
 - 7a** In the Explorer tree, select the Resource object for the machine that has the agent you are working with, select the *Policies* tab for this object in the admin view, then click *Choose* to open the Policy Selection dialog box.
 - 7b** In the Policy Selection dialog box, select the `vsphere_client` policy, click *Add*, then click *OK*.
- 8** (In the Orchestrate environment) Add vSphere credentials to the proxy agent:
 - 8a** On the agent host, change to the `C:\Program Files\Novell\ZOS\Agent` directory, then create a `\credentials` subdirectory under it.
 - 8b** Open a text editor, paste or type in the password of the vCenter Server, then save the file as `vspherecredentials.txt` to the new subdirectory you created in [Step 8a](#).

After a discovery is executed, the credentials are securely stored for future use on the agent, but the plain text `vspherecredentials.txt` file is removed.

You can also use the `webservice_password` value found in the `vsphere` policy for the contents of this text file, but we do not recommend doing this for production use.

Running Discovery

When a proxy agent has been deployed and properly configured in the vSphere environment and the policies of the proxy agent have been properly edited, you can go to the Orchestrate Development Client menu toolbar and select *Provision > Discover VM Hosts and Repositories*.

When you perform this discovery action, PlateSpin Orchestrate runs jobs that discover the VM hosts, repositories, and networks in each of the vSphere environments. On each discovered object, Orchestrate also generates a `*.vsphere.vcenter` fact that contains a vCenter ID from the hosting vSphere environment.

NOTE: When a VM has the same name in two different vSphere environments, the second VM discovered is given a name in the following form:

`VMNAME_VCENTERID`

You can rename these objects if you want to. For more information about renaming Grid objects, see “Resource Object Naming and Renaming,” “VM Host Object Naming and Renaming,” “VM Host Object Naming and Renaming,” “Virtual Disk Object Naming and Renaming,” “Virtual NIC Object Naming and Renaming,” “Virtual Bridge Object Naming and Renaming,” or “Repository Object Naming and Renaming” in the *PlateSpin Orchestrate 2.5 Development Client Reference*.

After the objects are discovered in the vSphere environments, you can use PlateSpin Orchestrate to discover existing VMs in those environments. From the Orchestrate Development Client menu toolbar, select *Provision > Discover VM Images*. PlateSpin Orchestrate discovers all of the VMs managed in the vSphere environments and places them in the Orchestrate model for you to manage.

When a VM with a given name is discovered in two different vSphere environments, the second VM discovered is named in the form of `VMNAME_VCENTERID`, rather than named by appending an incrementing number, as explained above. As with other such object names that are automatically generated, these VM names can be [changed](#).

If the discovery fails, use the following steps to restart the `vspheredaemon` job:

- 1 In the Jobs view of the Orchestrate Client, locate the running `vspheredaemon` job, then click the stop button to cancel the job.
- 2 Re-run the *Discover VM Images* provisioning action you attempted before. This restarts the `vSphereDaemon`.

4.1.7 Known Configuration Limitations for Virtual Center VMs

When a vSphere VM is deleted by the vSphere client, it can leave behind a VM template in a “defined” state that can be subsequently discovered and registered as a resource in PlateSpin Orchestrate.

If you attempt to remove this VM template in the Orchestrate Client, PlateSpin Orchestrate attempts to remove the VM’s files from the file system, but because the VM has no files to delete, the provisioning job fails with the following error:

```
Error: Delete : Failed delete of Managed Entity :
```

For other identified Virtual Center VM issues, see the *vSphere VM Issues in the Development Client* (http://www.novell.com/documentation/ps0_orchestrate25/readme/readme_pso25.html#bff169t) section of the *PlateSpin Orchestrate 2.5 Readme* (http://www.novell.com/documentation/ps0_orchestrate25/readme/readme_pso25.html).

4.2 Configuring the Xen Provisioning Adapter and Xen VMs

This section includes the following information:

- ♦ [Section 4.2.1, “Configuring Policies for Xen,” on page 44](#)
- ♦ [Section 4.2.2, “Known Configuration Limitations for Xen VMs,” on page 44](#)

4.2.1 Configuring Policies for Xen

The following table provides detailed information about the policies associated with the xen provisioning adapter that are used to manage the Xen hosts and VMs in the grid. The policy settings are applied to all the VMs in the grid.

Table 4-4 *Virtual Machine Management Policies for Xen 3.0 Server*

Policy Name	Explanation	Additional Details
xen	Contains the policy settings for the xen provisioning adapter.	By default, the optimal values are configured for the job and joblets in the policy.
xenDiscovery	Contains the settings required to discover the Xen host machines. It also contains the default installation path of the Xen server.	If the Xen server is not installed in the default path, edit this policy to provide the correct information.
xenPA	Contains the constraints used to check whether the Xen host is registered to the Orchestrate Server, and the host is up and running.	Do not edit the policy.

4.2.2 Known Configuration Limitations for Xen VMs

The following list describes the known limitations you can encounter when configuring Xen VMs in PlateSpin Orchestrate:

- ♦ PlateSpin Orchestrate modifies a paravirtualized or hardware virtualized Xen VM with the appropriate driver to match its network adapters. The “ioemu” driver is meant only for a fully virtualized guest, while the “netfront” driver is a paravirtualized driver that can be used either with a paravirtualized guest or with a hardware virtualized guest that has the proper paravirtualized drivers installed.

If you find that the VM is not obtaining proper connectivity or if you encounter some other networking error, check the `vnic.type` custom fact for the vNIC that is associated with the VM. Make sure the value for this fact matches the type of network adapter required by the VM.

- ♦ If you use `virt-manager` to create a VM in the default repository and then copy its config file from `/etc/xen/vm/<VM_name>` to a new location on a different repository, both of these config files point to the same physical disk image. If you do not remember to use the `xm delete` command on the original VM, the Discovery action on both the original repository and the new repository results in the same VM being discovered in both repositories. This can cause the provisioning adapter to become confused as to which repository holds the VM image

you want to use. To avoid this issue, we recommend that you use the Orchestrate Development Client to move any VMs you happen to create manually. The client takes care of properly deleting and re-creating the configuration for the VM.

4.3 Configuring the Amazon EC2 Provisioning Adapter

This section includes the following information:

- ♦ [Section 4.3.1, “Components of the EC2 Provisioning Adapter,” on page 45](#)
- ♦ [Section 4.3.2, “EC2 Objects Mapped to PlateSpin Orchestrate Objects,” on page 45](#)
- ♦ [Section 4.3.3, “Managing VMs in the EC2 Cloud,” on page 45](#)
- ♦ [Section 4.3.4, “Specifying Properties of an EC2 Template,” on page 46](#)

4.3.1 Components of the EC2 Provisioning Adapter

When you install PlateSpin Orchestrate, a new `ec2` provisioning job is deployed. This provisioning adapter uses the following components, which are also deployed during Orchestrate startup:

- ♦ **ec2.policy:** This file contains information that is used by the `ec2` provisioning job. In it, you define properties for your `ec2` accounts.
- ♦ **ec2_client.policy:** This policy must be associated with any resource that you want to use to manage EC2.
- ♦ **ec2PA.policy:** This policy is automatically associated with VMs discovered by the `ec2` provisioning adapter.
- ♦ **Boto Library:** A Python library that the provisioning adapter uses to communicate with EC2.

4.3.2 EC2 Objects Mapped to PlateSpin Orchestrate Objects

After discovery, EC2 objects are represented in the Orchestrate Development Client according to the following mappings:

- ♦ **EC2 Regions:** Each region creates a new virtual repository.
- ♦ **EC2 Availability Zones:** Each zone creates a new VM host object.
- ♦ **AMIs:** Each Amazon instance creates a new VM object.

4.3.3 Managing VMs in the EC2 Cloud

To begin managing VMs in the EC2 cloud:

- 1 Define the account information in `ec2.policy`:
 - 1a (Line 27) Name the account with a unique name. The account name serves as the key for account related information.

NOTE: Each token that needs to be replaced in the file is defined by using a `%TOKEN_NAME%` syntax. In other words, the text to be replaced is surrounded by “%” characters.

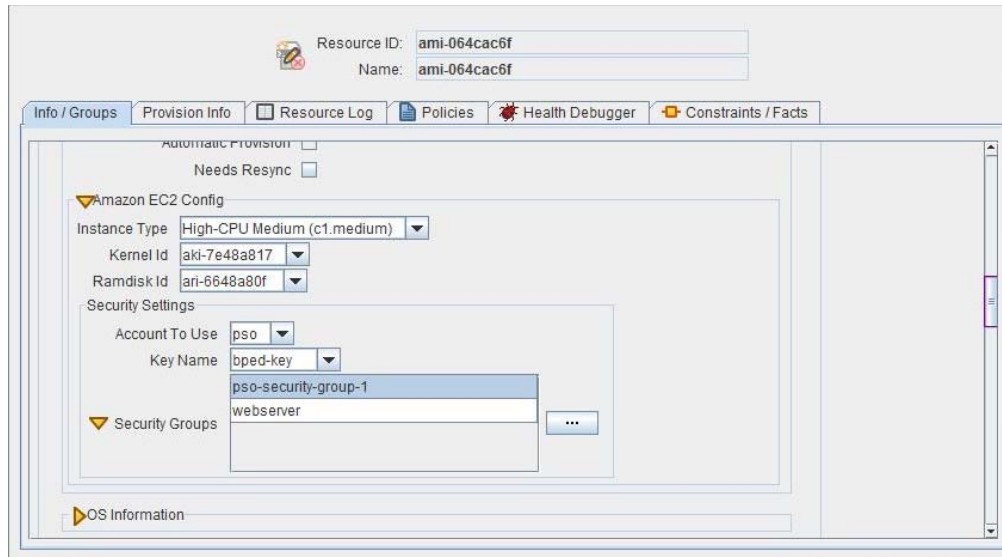
- 1b** (Line 31) Specify your AWS account ID. You can obtain this data from the EC2 Web interface.
- 1c** (Optional) Specify default values that will be applied to provisioning actions. Options that can be assigned across the zone are:
 - ◆ Security Groups (Line 64)
 - ◆ Keypair Names (Line 54)
 - ◆ Instance Types (Line 45 for 32-bit instance types, Line 48 for 64-bit instance types)
- 2** In the *Stored Credentials* panel of the *Orchestrate Server Authentication* tab, click *Add Credential* to add your EC2 account credentials to the PlateSpin Orchestrate credential manager.

The credential name must be in the form of `ec2_${account_id}`, where `${account_id}` is the name you used as the dictionary key in `ec2.policy`. (See [Step 1](#), above).
- 3** Choose a node that communicates with EC2. Associate the `ec2_client.policy` file with that node. You can do this by:
 - 3a** Dragging the `ec2_client.policy` file onto the target resource in the Explorer tree in the Development Client.
 - 3b** Performing the following operation:
 - 3b1** In the Development Client, select the target resource.
 - 3b2** Select the *Policies* tab, then click *Choose* to display the Policy Selection dialog box.
 - 3b3** In the Policy Selection dialog box, select the `ec2_policy` from the *Source Policies* list, then click *Add* to move it to the associated *Policies* list.
- 4** From the Orchestrate Development Client main menu, select *Provision > Discover VM Hosts & Repositories* to discover available Amazon regions.
- 5** When the Discover VM Hosts and Repositories dialog box is displayed, make sure you select the `ec2` provisioning adapter.
- 6** From the Development Client main menu, select *Provision > Discover VM Images* to discover available templates and images in the Amazon cloud.
- 7** From the Explorer tree, right-click a discovered Amazon VM and select an available action to begin managing templates and VMs.

4.3.4 Specifying Properties of an EC2 Template

When you want to provision a discovered Amazon EC2 template, you can specify properties for that template.

- 1** In the PlateSpin Development Client select an EC2 template object to open its Info/Groups page in the admin view.
- 2** Scroll to the *Amazon EC2 Config* subpanel of the Provisioning Information panel in the Info/Groups page.



3 Click *Define* on each field to configure the settings for the EC2 provisioning adapter:

- ◆ **Instance Type:** Specifies the size (measured in CPU, memory, and local storage) of the provisioned instance. For a breakdown of the available options, see the [Amazon Elastic Compute Cloud Developer Guide \(http://docs.amazonwebservices.com/AWSEC2/latest/DeveloperGuide/\)](http://docs.amazonwebservices.com/AWSEC2/latest/DeveloperGuide/).
- ◆ **Kernel ID:** Provides the ability to use a different Linux kernel with the selected image when provisioning. For example, you might choose a Xen kernel for your instance. Another use would be to ensure that your instances are kept up-to-date with the latest security fixes and updates.

NOTE: The architecture of the AMI, the RAM disk, and the kernel must match.

See the [Amazon EC2 User Selectable Kernels \(http://developer.amazonwebservices.com/connect/entry.jspa?externalID=1345&categoryID=112\)](http://developer.amazonwebservices.com/connect/entry.jspa?externalID=1345&categoryID=112) *Feature Guide* for more information.

- ◆ **RamdiskID:** Used in conjunction with *Kernel ID*. Some kernels require additional drivers when they launch. The RAM disk stores the required drivers.
- ◆ **Account to Use:** Select an account name from this list to limit the information in the *Key Name* and *Security Group* fields to data from the selected account.
If no value is selected for this field, the *Key Name* and *Security Group* fields are disabled. In addition, the first account configured in the `ec2.policy` file is used during Resync State operations for public templates.
- ◆ **Key Name:** Overrides the key name specified in the `ec2PA.policy` file. The key name identifies the public/private key pair that is used to ensure that only you have access to the instances you launch. The public key is stored in Amazon EC2. The private key, which you store, allows secure access to the instance. See the [EC2 user guide \(http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/\)](http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/) for more information.

- ♦ **Security Group:** A security group defines firewall rules for your instance. These rules specify which incoming network traffic should be delivered to your instance. All other incoming traffic will be discarded. This field lets you select multiple values.

See the [EC2 user guide \(http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/\)](http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/) for more information.

4.4 Configuring the hyperv Provisioning Adapter and Hyper-V VMs

This section includes the following information:

- ♦ [Section 4.4.1, “Configuring Policies for the hyperv Provisioning Adapter,” on page 48](#)
- ♦ [Section 4.4.2, “Known Configuration Limitations for Hyper-V VMs,” on page 48](#)

4.4.1 Configuring Policies for the hyperv Provisioning Adapter

Before provisioning and managing the hyperv provisioning adapter, you must configure certain policies in the Development Client. The following table provides detailed information about the policies associated with the hyperv provisioning adapter that are used to manage the hyperv server hosts and VMs in the grid. The policy settings are applied to all the VMs in the grid.

Table 4-5 *Virtual Machine Management Policies for the Hyper-V Server*

Policy Name	Explanation	Additional Details
hyperv	Contains the constraints used to check whether the hyperv server host is registered to the Orchestrate Server, and that the host is up and running.	Do not edit the policy.

4.4.2 Known Configuration Limitations for Hyper-V VMs

The following list describes the known limitations you can encounter when configuring Hyper-V VMs in PlateSpin Orchestrate:

- ♦ If you are managing Windows VMs in a Hyper-V environment (clustered or non-clustered), the PlateSpin Orchestrate hyperv provisioning adapter fails to discover iSCSI target repositories in that environment if the VM is in a location other than `C:\Users\Public\Documents\Hyper-V\Virtual Hard Disks`.

To work around this issue:

1. From the Explorer tree in the Development Client, select the *Repositories* group to expand the list of Repository Objects, select *hyperv*, then select the storage object associated to the Hyper-V cluster to open the admin view.
2. In the Info/Groups page of the admin view, find the *Preferred Storage Path* field (the `repository.preferredpath` fact).
3. In the Preferred Storage Path field, change the value to the path where the VM resides. Remember that this field considers the information in the *Root Location* field (that is `repository.location`).

This is the location where PlateSpin Orchestrate searches for VM files for use in cloning and moving. Generally, it is a path like this:

C:\Users\Public\Documents\Hyper-V\Virtual Hard Disks

4. Click the Save icon  to save the new configuration

NOTE: If your Hyper-V environment is a Cluster Storage Volumes (CSV) environment, the VMs on the CSVs are automatically discovered by the hyperv provisioning adapter as separate repositories. Executing the *Discover VM Images* action on these repositories discovers the VMs residing there.

- ◆ If you create a VM in your Hyper-V environment, but the path to that VM was not configured as the default path in the Hyper-V Manager, the PlateSpin Orchestrate Server does not discover the VM.

To work around this issue, you can edit the preferred path for the discovered repository where the VM resides, or you can create a new repository in the Development Client with the preferred path to the Hyper-V VM.

- ◆ As with other VMs managed by Xen and VMware, the OS info for Hyper-V VMs is not always auto-discovered by PlateSpin Orchestrate.

The OS info for Hyper-V VMs is discovered only in the following circumstances:

- ◆ The Discover VM Images event triggers a Resync event for offline VMs to get OS info. However, OS info is not retrieved for the discovered templates.
- ◆ The Resync event on the VM retrieves the OS family and type if it is offline.
- ◆ The Resync event on the template sets the OS family to “Windows” because only Windows templates are supported in the hyperv provisioning adapter.
- ◆ The Create Template event tries to retrieve OS info before creating the template. Create Template succeeds only if the VM’s OS family is Windows.

To work around this issue, you can either enter this information manually, install the PlateSpin Orchestrate Client on the VM to enable discovery, or use the Key/Value Pair Exchange mechanism to support integration services installation.

- ◆ After adding a vDisk to a Hyper-V Linux VM, that vDisk is not visible inside the VM. To work around this issue, you need to install Microsoft Linux Integration Components for Linux. See the [Microsoft downloads site \(http://download.microsoft.com/download/4/2/7/4273D9CF-3FC3-4A91-8204-9E0D4DE2027C/Linux%20Integration%20Components%20Read%20Me.pdf\)](http://download.microsoft.com/download/4/2/7/4273D9CF-3FC3-4A91-8204-9E0D4DE2027C/Linux%20Integration%20Components%20Read%20Me.pdf) for more information.

Troubleshooting Provisioning Actions

5

The following sections provide solution to the problems you might encounter while performing provisioning actions:

- ♦ “After installing the Orchestrate Agent on VM, the VM is not displayed as a resource in the Orchestrate Development Client” on page 51
- ♦ “The VM is suspended when you try to revert the snapshot of a powered-on VM running on a Hyper-V host” on page 51
- ♦ “Unable to perform any provisioning adapter action after the Save Config action on the vSphere managed VM” on page 52
- ♦ “Provisioning of a Xen VM doesn’t work on the host server” on page 52
- ♦ “Multiple instances of the same Xen VM running when located on shared storage” on page 53

After installing the Orchestrate Agent on VM, the VM is not displayed as a resource in the Orchestrate Development Client

Source: The PlateSpin Orchestrate Development Client.

Action: Do the following:

- ♦ Ensure that the Orchestrate Agent is running on the VM.
- ♦ Ensure that no errors have been logged into the `agent.log` file.

The log file is located in the

`Orchestrate_Agent_installation_directory\novell\zos\agent\node.default` directory on Windows and in the `/opt/novell/zos/agent/node.default` directory on Linux.

- ♦ Ensure that the Orchestrate Server is registered to the DNS server.

The VM is suspended when you try to revert the snapshot of a powered-on VM running on a Hyper-V host

Source: The PlateSpin Orchestrate Development Client.

Explanation: If you try to revert the snapshot of a powered-on VM running on a Hyper-V host, the VM is suspended. This is a known behavior of VMs running on a Hyper-V host.

Action: Provision the suspended VM:

- 1 In the Orchestrate Development Client, right-click the suspended VM, then click *Provision*.

The Provision VM dialog box is displayed.

- 2 In the *Plan (Host/Repository)* drop-down list, select the appropriate Hyper-V host.
- 3 Click *OK*.

Unable to perform any provisioning adapter action after the Save Config action on the vSphere managed VM

Source: The PlateSpin Orchestrate Development Client.

Possible Cause: The VM UUID value of the vSphere managed VM is not a 128-bit hexadecimal value. Even though the Save Config action is successful and the VM is provisioned, the hypervisor automatically assigns a different UUID value. Subsequently, any provisioning adapter action performed on the VM fails.

Action: Specify a 128-bit hexadecimal value for the VM UUID.

- 1 In the Orchestrate Development Client, click *Resources* > the vSphere managed VM.

The *Info/Groups* tab is displayed by default.

- 2 In the Virtual Machine Configuration panel, set the value of *VM UUID* to a 128-bit hexadecimal value.

- 3 Right-click the vSphere managed VM, then click *Save Config*.

Provisioning of a Xen VM doesn't work on the host server

Source: The PlateSpin Orchestrate Development Client.

Explanation: When you try to provision a Xen VM, the job might fail with the following error message in the job log:

```
[c121] RuntimeError: vmprep: Autoprep of /var/lib/xen/images/  
min-tmpl-1-2/disk0  
failed with return code 1: vmprep: autoprep:  
/var/adm/mount/vmprep.3f96f60206a2439386d1d80436262d5e:  
Failed to mount vm  
image "/var/lib/xen/images/min-tmpl-1-2/disk0": vmmount: No  
root device found  
Job 'zosSystem.vmprep.76' terminated because of failure.  
Reason: Job failed
```

A VM host cannot provision a VM that has a different file system than the VM host. The currently supported file systems are ext2, ext3, reiserfs, jfs, xfs, vfat, and ntfs.

Action: To work around the issue, load the VM's file system Linux module on the VM host, or add it to the Linux kernel if a custom kernel is being used.

Typically, Linux kernels autoload the appropriate module to do the work.

You must manually load the proper kernel module on the VM host to support the VM's file system.

For example, if the VM host uses ext3 and the VM image uses reiserfs, load the proper kernel module onto the VM host to support the VM image's reiserfs file system. Then, on the VM host, run:

```
modprobe reiserfs
```

Next, provision the VM.

NOTE: Cloning with prep is limited to what the Virtual Center of VMware Server supports.

Multiple instances of the same Xen VM running when located on shared storage

Source: Shared storage for Xen VMs.

Explanation: The `xendConfig` job runs when a VM host is added to the PlateSpin Orchestrate Server. This job automates some of the configurations possible on a Xen VM host. With the default Xen configuration, it is possible to incorrectly start a running VM a second time from storage that is shared by and accessible to another Xen VM host.

Possible Cause: A running Xen VM can only be locked to a specific Xen VM host when the `xend` service is configured to share a VM domain lock file on a shared file system. By default, the `xend` service places these VM domain lock files in the `/var/lib/xend/domains` directory, which is usually not located on shared storage.

Action: You can configure Xen VM locks in PlateSpin Orchestrate by uncommenting certain facts in the policy file (search for `xend.xend-domain-lock`).

Uncomment these facts in `xendConfig.policy`:

```
<!--
<fact name="xend.xend-domain-lock"
  type="String"
  description="Create an external lock file when domains are started. Lock
    file is placed in xend-domains-lock-path/[domain_uid] on domain
    startup and removed when domain is stopped. Set to yes to enable
    lock file creation.

    Note that external locking mechanisms are no substitute for a cluster
    environment that protects shared resources, but may be useful in
    some circumstances nonetheless."
  value="yes" />
<fact name="xend.xend-domain-lock-path"
  type="String"
  description="Path where domain lock is stored if xend-domain-lock is enabled.
    Note: This path must be accessible to all VM Servers participating
    in domain locking, e.g. by specifying a shared mount point.
    Lock is placed in [xend-domain-lock-path]/[domain-uid].
    Default is /var/lib/xend/domains/"
  value="/var/lib/xend/domains" />
-->
```

To uncomment a section of code, remove the “<!--” (comment open) tag and the “-->” (comment close) tag. Edit the `xend-domain-lock-path` fact to set an alternate location on shared storage that is available to all VM hosts.

When you make the changes and save the file, the facts become active and the VM locking parameters of each newly joining VM host are adjusted accordingly.

You can also schedule an immediate run of the `xendConfig` job to adjust all configuration files of the Xen VM hosts that are already connected to the PlateSpin Orchestrate Server.

NOTE: Setting the lock path by using PlateSpin Orchestrate only supports the scenario where all VM hosts have the domain lock path connected to the same shared repository. For more complex setups, you need to use alternative methods to adjust the VM host lock configurations.

Autoprep and Sysprep

6

This section includes the following information:

- ♦ [Section 6.1, “Understanding Autoprep,” on page 55](#)
- ♦ [Section 6.2, “Understanding Sysprep,” on page 61](#)

6.1 Understanding Autoprep

In the PlateSpin Orchestrate Development Client, “autoprep” refers to the function of preparing unique network settings for a Linux VM so that VM can be provisioned by its provisioning adapter without creating network conflicts and without customizing other network-related settings.

As the administrator, you can set facts in the PlateSpin Orchestrate Development Client that can later be applied to a VM clone during a *Provision* or a *Clone* action from a VM template. You can also use the *Personalize* action to manually apply autoprep facts to an existing VM.

This section includes the following information:

- ♦ [Section 6.1.1, “How Autoprep Works,” on page 55](#)
- ♦ [Section 6.1.2, “Setting Autoprep Facts in the Development Client,” on page 56](#)
- ♦ [Section 6.1.3, “Applying Autoprep Facts,” on page 59](#)
- ♦ [Section 6.1.4, “Example Autoprep Scenarios,” on page 60](#)
- ♦ [Section 6.1.5, “Known Autoprep Limitations,” on page 60](#)

6.1.1 How Autoprep Works

The `vmprep` job always runs when you clone or provision from a VM template. The job prepares the root disk image of the VM with the defined autoprep settings. On a Linux system, the global autoprep settings for a VM are stored in various configuration files in the `/etc` directory. For example, the hostname is stored in `/etc/HOSTNAME`. Global network properties are stored in `/etc/sysconfig/network/config` and in `/etc/sysconfig/network/dhcp`. Per-NIC properties are written to the various `/etc/sysconfig/network/ifcfg.*` scripts, with one for each virtual NIC.

The `vmprep` job attempts to identify the disk image with the root partition, then mounts that partition and starts scanning the configuration files to make the necessary changes to the VM configuration file settings.

If the *Use Autoprep* check box in the VM Client or the *Use Autoprep* check box in the Development Client is not selected, the `vmprep` job still runs, but only to change the name of the Orchestrate Agent (if installed) on the VM.

If you want a full autoprep with system config changes when cloning or provisioning from template, you need to select the *Use Autoprep* check box in the VM Client or the *Use Autoprep* check box in the Development Client.

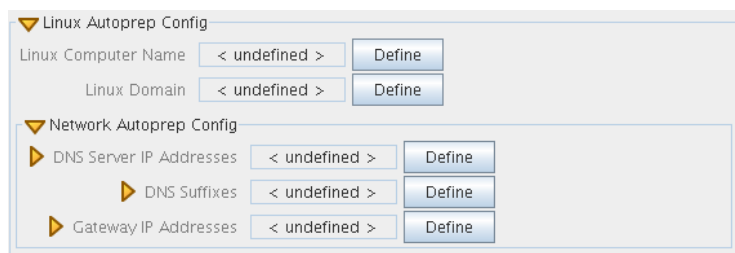
For Linux VMs, autoprep mounts the VM image's root disk image and edits the appropriate files in the `/etc/` directory to make the desired configuration changes. This might include adding network interface configurations to the network configuration scripts. The changes take effect when the VM starts again.

6.1.2 Setting Autoprep Facts in the Development Client

You can use the Development Client to configure the facts for autoprep of a VM. This section includes information about the Development Client interface where those facts are set.

When you select a Linux VM object in the Explorer tree of the Development Client, click the *Info/Groups* tab to open the Info Groups page, then scroll down to the *Provisioning Information* panel of this page. Open the *Linux Autoprep Config* panel and the *Network Autoprep Config* panels.

Figure 6-1 The Autoprep Sections of the Info/Groups Page of a VM Template Object



Linux VMs that you clone can be personalized and prepared for provisioning by configuring the facts in this panel. Click *Define* on each field if the value has not been previously configured.

NOTE: When you change any of the settings in this panel, you need to right-click the VM and select *Personalize* for the changes to take effect. This action is in contrast to right-clicking a template, which can apply these settings during a provision or clone operation.

This section also contains this information:

- ♦ [“Linux Autoprep Config” on page 56](#)
- ♦ [“Network Autoprep Config” on page 57](#)
- ♦ [“Defining Autoprep/Sysprep Network Adapter Facts on the vNIC Object” on page 57](#)

Linux Autoprep Config

The settings located in the *Linux Autoprep Config* panel are global to a configuration of a Linux VM and are not specific to a particular network adapter.

NOTE: It is not mandatory to define these facts. If they are left undefined, they are not applied to the “autoprepmed” VM.

- ♦ **Linux Computer Name:** The network host name of the new VM. If you specify an asterisk (*), the current Grid object ID (`resource.id`) of the new VM is used.

The Linux Computer Name should be the unqualified computer name without the DNS domain suffix, such as `webserver` instead of `webserver.acme.com`.

In the Fact Editor, this fact is listed as

```
resource.provisioner.autoprep.linuxglobal.ComputerName:
```

```
<fact name="resource.provisioner.autoprep.linuxglobal.ComputerName"
value="" type="String" />
```

- ♦ **Linux Domain:** The network domain name where the new VM is a member.

This field should contain the default DNS domain for the host, such as acme.com.

In the Fact Editor, this fact is listed as

```
resource.provisioner.autoprep.linuxglobal.Domain:
```

```
<fact name="resource.provisioner.autoprep.linuxglobal.Domain" value=""
type="String" />
```

Network Autoprep Config

This section includes the following fields:

- ♦ **DNS Server IP Addresses:** The list of DNS Servers for name for lookup. This setting is only for cloning/personalize actions. For Linux, it should be set only in the VM facts, not in the vNIC facts.

In the Fact Editor, this fact is listed as an array:

```
<fact name="resource.provisioner.autoprep.DNSServers">
  <array>
    <string></string>
  </array>
</fact>
```

- ♦ **DNS Suffixes:** The list of suffixes to append to a name for lookup. This setting is only for cloning/personalize actions. For Linux, it should be set only in the VM facts, not in the vNIC facts.

```
<fact name="resource.provisioner.autoprep.DNSSuffixes">
  <array type="String">
  </array>
</fact>
```

- ♦ **Gateway IP Addresses:** The list of Internet gateways available to this VM. This setting is only for cloning/personalize actions. For Linux, it should be set only in the VM facts, not in the vNIC facts.

In the Fact Editor, this fact is listed as an array:

```
<fact name="resource.provisioner.autoprep.Gateways">
  <array>
    <string></string>
  </array>
</fact>
```

Defining Autoprep/Sysprep Network Adapter Facts on the vNIC Object

VMs can be prepared for provisioning by configuring the facts in either the *Autoprep Network Adapter* subpanel (Windows VMs) of the vNIC *Info/Groups* panel or the *Sysprep Network Adapter* subpanel (Linux VMs). Click *Define* on each field if the value has not been previously configured.

NOTE: When you change any of the settings in this panel, you need to right-click the VM and select *Personalize* for the changes to take effect.

- ♦ **MAC Address:** The MAC address of the interface. Specify an asterisk (*) or specify no setting at all to generate a new MAC address. If the value is not set, the existing `vmnic.mac` is used.

IMPORTANT: An unset *MAC Address* fact generates a new MAC address. This is contrary to the current tooltip text.

In the Fact Editor, this fact is listed as `vmnic.provisioner.autoprep.MACAddress`:

```
<fact name="vmnic.provisioner.autoprep.MACAddress" value="" type="String" />
```

- ♦ **Use DHCP:** When this check box is selected (it has a value of true), the VM is configured to retrieve its network settings from a DHCP server. If the check box is not selected (it has value of false), you should make sure that the IP address, subnet mask, and gateway address facts are defined. In the Fact Editor, this fact is listed as `vmnic.provisioner.autoprep.UseDHCP`:

```
<fact name="vmnic.provisioner.autoprep.UseDHCP" value="false" type="Boolean" />
```

- ♦ **IP Address:** The IP address for the adapter.

In the Fact Editor, this fact is listed as `vmnic.provisioner.autoprep.IPAddress`:

```
<fact name="vmnic.provisioner.autoprep.IPAddress" value="" type="String" />
```

- ♦ **Subnet Mask:** The subnet mask for this adapter.


In the Fact Editor, this fact is listed as `vmnic.provisioner.autoprep.subnetMask`:

```
<fact name="vmnic.provisioner.autoprep.subnetMask" value="" type="String" />
```

- ♦ **Gateway IP Addresses:** (Windows only) A list of the gateway IP addresses available to the interface.

In the Fact Editor, this fact is listed as an array:

```
<fact name="vmnic.provisioner.autoprep.Gateways">
  <array type="String">
  </array>
</fact>
```

You can edit this array by clicking the  button to open an array editor. In this dialog box, you can add or remove the IP address or change its order in the array of element choices.

- ♦ **DNS from DHCP:** When this check box is selected (it has a value of true), the SUSE VM is configured to retrieve its DNS server settings from DHCP.

In the Fact Editor, this fact is listed as `vmnic.provisioner.autoprep.DNSFromDHCP`:

```
<fact name="vmnic.provisioner.autoprep.DNSFromDHCP" value="false" type="Boolean" />
```

- ♦ **DNS Server IP Addresses:** (Windows VM only) The adapter's list of DNS servers used for name lookup.

In the Fact Editor, this fact is listed as an array:

```
<fact name="vmnic.provisioner.autoprep.DNSServers">
  <array type="String">
  </array>
</fact>
```

- ♦ **DNS Domain:** (Windows VM only) The adapter's DNS domain name.

In the Fact Editor, this fact is listed as `vnics.provisioner.autoprep.DNSDomain`:

```
<fact name="vnics.provisioner.autoprep.DNSDomain" value="" type="String" />
```

- ♦ **Primary WINS Server:** (Windows VM only) The name of the adapter's primary WINS server.

In the Fact Editor, this fact is listed as `vnics.provisioner.autoprep.primaryWINS`:

```
<fact name="vnics.provisioner.autoprep.primaryWINS" value="" type="String" />
```

- ♦ **Secondary WINS Server:** (Windows VM only) The name of the adapter's secondary WINS server.

In the Fact Editor, this fact is listed as `vnics.provisioner.autoprep.secondaryWINS`:

```
<fact name="vnics.provisioner.autoprep.secondaryWINS" value="" type="String" />
```

- ♦ **DNS Suffixes:** Although it is possible to enter information in this field, it is unused.

- ♦ **NetBIOS:** (Windows VM only) The NetBIOS options for this VM. Options include:

- ♦ *EnableNetBIOSviaDhcp*
- ♦ *EnableNetBIOS*
- ♦ *DisableNetBIOS*

In the Fact Editor, this fact is listed as `vnics.provisioner.autoprep.netBIOS`:

```
<fact name="vnics.provisioner.autoprep.netBIOS" value="" type="String" />
```

NOTE: Although you can define individual static settings to be applied to these adapters, autoprep can be useful for provisioning multiple clones with unique, autogenerated MAC addresses and DHCP-defined IP addresses (even though the VM clones are copies of the same VM template OS image) by coupling the autoprep settings on the VM with the autoprep settings on the vNIC object associated with the VM, thus avoiding network conflicts. For more information about vNIC autoprep settings, see [“Defining Autoprep/Sysprep Network Adapter Facts on the vNIC Object” on page 57](#).

6.1.3 Applying Autoprep Facts

The PlateSpin Orchestrate Server applies the autoprep facts by launching the `vmprep` job when the facts are defined. This job runs automatically and applies the appropriate facts to a VM in the following situations:

- ♦ When a *Personalize* action is run on any non-template VM. (See [Table 2-2 on page 17](#)).

On VMs managed by Xen and vSphere, running the *Personalize* action on a templated VM is not supported. Running this action results in failure because it is not supported in the underlying system. When you clone or provision from a templated VM, select the *Use Autoprep* check box.

- ♦ When a VM clone is created by initiating the *Clone* action on a VM template.

Select the *Use Autoprep* check box in the VM Client or the *Use Autoprep* check box in the Development Client if autoprep facts are to be used when the *Clone* action is initiated.

- ◆ When a VM clone is created by initiating a *Provision* action on a VM template.
Select the *Use Autoprep* check box in the VM Client or the *Use Autoprep* check box in the Development Client if autoprep facts are to be used when the *Clone* action is initiated.

6.1.4 Example Autoprep Scenarios

Scenario 1: You want to create 25 dynamic VM instances to test job provisioning. You will never use these instances again.

You create a VM template by right-clicking a VM, then you select *Create Template*. When the VM Template is created in the Explorer Tree, you define its autoprep facts in the *Info/Groups* page of the vNIC object by specifying an asterisk in the *MAC Address* field, then you select the *Use DHCP* check box. This lets the Development Client autogenerate the MAC address and retrieve network data from the DHCP server. For information about setting autoprep facts on each vNIC, see “[The Virtual NIC Info Panel](#)” in the *PlateSpin Orchestrate 2.5 Development Client Reference*.

When the autoprep facts are defined, you provision this template. You right-click the template object and select *Provision*, then in the Provision VM dialog box, you specify that you want to provision (create) 25 new VM instances from this template. Provisioning automatically applies the autoprep facts from the template if the *Use Autoprep* check box is selected.

Scenario 2: You have created three VM clones in your grid and you want to provision those clones. You want to ensure that the MAC address and other key network information for each clone is unique, even though each clone is a copy of the same OS image. These clones are to be detached later and used for such things as mail servers and Web servers. When the clones were first created, autoprep facts were applied, but now you have changed those facts by adding static IP addresses, subnet masks, and gateway addresses for each. Each clone must be “personalized” because of this change to basic network identifiers.

To personalize, you select each Clone object, then define the adapter-specific settings on the *Info/Groups* page of each of the VM’s vNICs by entering IP addresses, DNS suffixes, and gateway addresses for each vNIC in the *Network Autoprep Config* subpanel. When you have defined the autoprep facts on each VM clone, you right-click each Clone object in turn and select *Personalize* to apply the new network configuration.

For more information, see “[Changing a Virtual Machine Template Clone to an Instance](#)” on page 22 and “[Personalize](#)” on page 23.

6.1.5 Known Autoprep Limitations

There are some limitations that you need to be aware of when you use autoprep:

- ◆ Currently, the *Gateway IP Addresses* setting in the *Info/Groups tab* for a VM object is available in a list box.

Because the Linux VM OS accepts only one default gateway, it accepts only the first setting in the list as the actual gateway IP address. The other settings are ignored.

- ◆ Before performing an autoprep operation on a vSphere VM, you must first configure the VM to have the VMware Tools guest OS package installed. The VMware administrator must also have configured the Virtual Center server to allow sysprep. For more information, see the [VMware vSphere Online Library \(http://pubs.vmware.com/vsp40/wwhelp/wwhimpl/js/html/wwhelp.htm#href=admin/c_installing_the_microsoft_sysprep_tools.html\)](http://pubs.vmware.com/vsp40/wwhelp/wwhimpl/js/html/wwhelp.htm#href=admin/c_installing_the_microsoft_sysprep_tools.html).

- ◆ When a VM is autoprepped in vSphere, it goes into a “customization pending” state that is not cleared until the VM successfully starts and the prep completes. This means that subsequent attempts to autoprep fail until the VM reboots and the previous prep completes. There is no way to directly reverse customization in vSphere.
- ◆ Customization of a vSphere VM for autoprep does the following with regard to the MAC address:
 - ◆ If the MAC address field in the Development Client autoprep subpanel is empty (or if an asterisk is entered), PlateSpin Orchestrate auto-generates a new MAC address.
 - ◆ If the MAC address value is in the range 00:50:56:00:00:00 → 00:50:56:3f:ff:ff, PlateSpin Orchestrate statically assigns the MAC address.
 - ◆ If the MAC address value is not empty and is outside the defined range, PlateSpin Orchestrate attempts to assign the MAC address, but it might be reassigned by vSphere if it is determined to be duplicate.


For more information, see VMware [Data Object - VirtualEthernetCard \(http://www.vmware.com/support/developer/vc-sdk/visdk400pubs/ReferenceGuide/vim.vm.device.VirtualEthernetCard.html\)](http://www.vmware.com/support/developer/vc-sdk/visdk400pubs/ReferenceGuide/vim.vm.device.VirtualEthernetCard.html) documentation.

- ◆ The Orchestrate vsphere provisioning adapter checks the setting for the host name. If the host name is not set, the setting defaults to the Orchestrate VM name.
- ◆ If a Hyper-V VM is running during the discover process, it fails to discover the `resource.os.family` fact. This prevents the Development Client from displaying the *Sysprep* and *Autoprep* options section on the *Info/Groups* tab for the VM.

NOTE: If the VM not running when the discovery occurs, the hyperv provisioning adapter discovers `resource.os.family` itself.

If you create a template from this VM, the `resource.os.family` fact is discovered and populated on the VM template admin view.

To display the sysprep/autoprep settings on a Hyper-V VM:

1. Shut down the Hyper-V VM that has the problem.
2. From the Explorer Tree, right-click the VM you shut down, then select *Resync State*.
3. In the Development Client, Shift+click the Refresh  icon or restart the Development Client to refresh all of the objects and their facts in the Resource admin view.

6.2 Understanding Sysprep

In the PlateSpin Orchestrate Development Client, sysprep refers to the function of preparing unique settings for Windows VMs on VM hosts so that those VMs can be provisioned by the provisioning adapter without creating conflicts and personalizing other information.

As the administrator, you can set facts in the PlateSpin Orchestrate Development Client that can later be automatically applied to a VM clone (by selecting the *Use Autoprep* check box) during a *Provision* or a *Clone* action from a VM template. The sysprep facts can also be manually applied to an existing VM by using the *Personalize* action.

NOTE: In PlateSpin Orchestrate 2.5, Windows VMs managed by the xen provisioning adapter and the hyperv provisioning adapter can now be “sysprepped” in the Development Client. This functionality was not supported in earlier releases of the product.

Make sure you have performed these prerequisites:

- ◆ Create a Windows VM by using the Xen hypervisor.
- ◆ Make sure the VM is discovered by PlateSpin Orchestrate.
- ◆ Configure the VM (or a template of the VM) by using the information in [Section 6.2.2, “Setting Sysprep Facts in the Development Client,” on page 63](#) and in [Section 6.2.3, “Using the Sysprep deploy.cab Files,” on page 72](#).

When the prerequisites are met, you can proceed to sysprep the xen VM by using the information in [Section 6.2.4, “Applying Sysprep Facts,” on page 74](#).

This section includes the following information:

- ◆ [Section 6.2.1, “How Sysprep Works,” on page 62](#)
- ◆ [Section 6.2.2, “Setting Sysprep Facts in the Development Client,” on page 63](#)
- ◆ [Section 6.2.3, “Using the Sysprep deploy.cab Files,” on page 72](#)
- ◆ [Section 6.2.4, “Applying Sysprep Facts,” on page 74](#)
- ◆ [Section 6.2.5, “Example Sysprep Scenarios,” on page 75](#)
- ◆ [Section 6.2.6, “Known Sysprep Limitations,” on page 75](#)

6.2.1 How Sysprep Works

This section includes the following information:

- ◆ [“Sysprep onVMware VMs” on page 62](#)
- ◆ [“Sysprep On Xen VMs” on page 62](#)

Sysprep onVMware VMs

Sysprep facts (see [Section 6.2.2, “Setting Sysprep Facts in the Development Client,” on page 63](#)) are translated by the PlateSpin Orchestrate vSphere provisioning adapter into a VMware image-customization specification. The provisioning adapter passes the specification to the vSphere client API, which does the customization work. The changes made are leveraged by the Windows sysprep facility to complete final reconfiguration.

The vsphere provisioning adapter requires the VMware Tools package installed and running on the target VM before sysprep can be done, because that is a requirement of the underlying VMware Image Customization functionality leveraged by PlateSpin Orchestrate.

Sysprep On Xen VMs

The xen provisioning adapter uses the settings specified in the sysprep facts to perform an “unattended mini-setup” to reconfigure the VMs’ Windows guest operating system. The provisioning adapter directly modifies the VM image without need for any agent, so you can configure sysprep on a “cold” VM image in the same way as you can run the *Install Agent* action on that image.

NOTE: You can perform the *Install Agent* and *Personalize* actions at the same time on a Xen Windows VM. The two operations cooperate and complete without conflict.

6.2.2 Setting Sysprep Facts in the Development Client

You can use the Development Client to configure the facts for sysprep of a VM. This section includes information about the Development Client interface where those facts are set.

When you select a Windows VM object in the Explorer tree of the Development Client, click the *Info/Groups* tab to open the Info Groups page, then scroll down to the *Provisioning Information* panel of this page. Open the *Windows Sysprep Config* subpanel and the *Network Sysprep Config* subpanel.

Figure 6-2 The Sysprep Sections of the Info/Groups Page of a VM Template Object

The screenshot shows two subpanels for configuring sysprep settings. The top subpanel, 'Windows Sysprep Config', contains 18 rows of settings, each with a text input field containing '< undefined >' and a 'Define' button. The settings are: Change SID, Delete Accounts, Admin Password, Admin Password Plaintext, Timezone, Autologon, Autologon Count, Fullname, Org Name, Computer Name, Product ID, Run Once Command, Workgroup, Domain, Domain Admin, Domain Admin Password, Domain Admin Password Plaintext, License File Automode, and License File Autousers. The bottom subpanel, 'Network Sysprep Config', contains three rows: 'DNS Server IP Addresses', 'DNS Suffixes', and 'Gateway IP Addresses', each with a text input field containing '< undefined >' and a 'Define' button.

Windows VMs that you clone can be personalized and prepared for provisioning by configuring the facts in this panel. Click *Define* on each field if the value has not been previously configured.

NOTE: You can trigger a sysprep for a Windows VM just by configuring the *Admin Password* field on this page. The Orchestrate provisioning adapter fills in the other required values with reasonable defaults if you don't specify them. For example, the value for the *Computer Name* field uses the PlateSpin Orchestrate VM name by default.

When you finish changing the settings in this panel, right-click the VM and select *Personalize* for the changes to take effect. This sets up a pending customization that does not take place until the VM is powered on (provisioned) again.

IMPORTANT: On VMs managed by Xen and vSphere, running the *Personalize* action on a templated VM is not supported. Running this action results in failure because it is not supported in the underlying system. When you clone or provision from a templated VM, select the *Use Autoprep* check box.

This section also contains the following information:

- ♦ [“Sysprep Facts” on page 64](#)
- ♦ [“Configuring vNIC Sysprep Facts” on page 72](#)

Sysprep Facts

The following table lists the facts that are either required or optional for configuring sysprep.

Table 6-1 *Required or Optional Facts for Sysprep Configuration*

String in Development Client UI	Fact Name	Required/Optional	Description and Information
<i>Change SID</i>	<code><fact name="resource.provisioner.autoprep.options.changeSID" value="false" type="Boolean" /></code>	Automatic default value provided by Orchestrate ¹	<p>The Windows Security ID. If this is marked as true, sysprep generates a new Security ID.</p> <p>If this is not selected, Orchestrate defaults the value to true, meaning a new SID is to be generated during sysprep.</p> <p>For newer (that is, unattended .xml-based) sysprep, unless this fact is defined and explicitly set to false, the SID is always changed. This is the desired behavior for cloning a Windows machine.</p>

String in Development Client UI	Fact Name	Required/Optional	Description and Information
<i>Delete Accounts</i>	<pre><fact name="resource.provisioner.autoprep.options.deleteAccounts" value="false" type="Boolean" /></pre>	Optional ²	<p>(Windows with vSphere VMs) When this check box is selected (it has a value of true), PlateSpin Orchestrate removes all user accounts from the destination VM; the Administrator account and other Windows "standard" accounts are left in place. If it is false, existing accounts from the source VM are retained.</p> <p>(Xen) This field is deprecated for Xen sysprep. Instead, ensure that the VM image has the required set of accounts from the beginning.</p> <p>No account deletion is done unless this fact is defined and set to true. Also, some versions of Windows sysprep do not support account deletion during sysprep, in which case this flag is ignored.</p>
<i>Admin Password</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.GuiUnattended.AdminPassword.value" value="" type="String" /></pre>	Required ³	<p>The Admin password for this VM.</p> <hr/> <p>NOTE: Only a plaintext admin password is currently supported. You should leave this field blank if <i>Admin Password Plaintext</i> is not selected.</p> <hr/> <p>This fact must be specified or the personalization fails. Windows sysprep requires that this be specified.</p>

String in Development Client UI	Fact Name	Required/Optional	Description and Information
<i>Admin Password Plaintext</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.GuiUnattended.AdminPassword.plainText" value="false" type="Boolean" /></pre>	Automatic default value provided by Orchestrate ¹	<p>When this check box is selected (it has a value of true) the <i>Admin Password</i> value is entered in plain text.</p> <p>If not set, this fact defaults to true, indicating that the AdminPassword fact is a plain text password.</p> <p>Orchestrate does not support automatic encryption of the password, so if you want to use an encrypted password, you need to know how to encrypt the password correctly for <code>sysprep.inf</code>, then enter it as <code>AdminPassword.value</code> with this flag set to false.</p>
<i>Timezone</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.GuiUnattended.TimeZone" value="" type="String" /></pre>	Automatic default value provided by Orchestrate ¹	<p>The time zone of the new VM. For sysprep on Windows versions prior to Vista, (that is, versions using <code>sysprep.inf</code>), numeric time zone codes are used. See the Microsoft sysprep documentation for values (for example, 04 indicates PST, 10 indicates MST, 20 indicates Central, 35 indicates EST as defined in the Windows sysprep documentation (http://technet.microsoft.com/en-us/library/cc749073.aspx)).</p> <hr/> <p>NOTE: Make sure that you use the exact text string listed under the Time Zone column heading in the table included in this Microsoft article.</p> <hr/> <p>For sysprep on Windows Vista and later (that is, versions using <code>unattend.xml</code>), full string time zone names are used. Refer to the Microsoft sysprep documentation for the relevant Windows version.</p> <p>If you do not set this fact, the default time zone for <code>sysprep.inf</code> (UTC or 85) is used.</p>

String in Development Client UI	Fact Name	Required/Optional	Description and Information
<i>Autologon</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.GuiUnattended.AutoLogon" value="false" type="Boolean" /></pre>	Optional ²	<p>When this check box is selected (it has a value of true) the VM automatically logs on to the Administrator account by using <i>AdminPassword</i>.</p> <p>If the value is false, logon is prompted.</p> <p>If no value is provided, the fact is set to false.</p>
<i>Autologon Count</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.GuiUnattended.AutoLogon" value="" type="Boolean" />></pre>	Optional ²	<p>The limit count for the VM to automatically log on with the Administrator account. <i>AutoLogon</i> must be true for this value to be accepted.</p> <p>If a value is not specified for this fact, but <i>Autologon</i> is set to true, the value defaults to 1.</p>
<i>Fullname</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.UserData.FullName" value="" type="String" /></pre>	Automatic default value provided by Orchestrate ¹	The user's full name required by the Windows OS installer during installation.
<i>Org Name</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.UserData.OrgName" value="" type="String" /></pre>	Automatic default value provided by Orchestrate ¹	<p>The organization name required by the Windows OS installer during installation.</p> <p>This fact is required by sysprep. If the value is not specified, the provided default is Organization Name.</p> <p>This fact is nonessential for Windows functionality.</p>

String in Development Client UI	Fact Name	Required/Optional	Description and Information
<i>Computer Name</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.UserData.ComputerName" value="" type="String" /></pre>	Automatic default value provided by Orchestrate ¹	<p>The VM's new host name. If you specify an asterisk (*) in this field, Orchestrate Services generates the name based on the source VM name.</p> <p>This fact is required by sysprep. If the value is not set, the default value is the name of the VM in Orchestrate Services.</p> <p>The name cannot be longer than 15 characters because of a Windows limitation on the length of the computer name. Values longer than 15 characters are not accepted.</p> <p>Because facts can be edited using methods other than the Admin view of the Server Console, be aware that there are other restrictions regarding the characters you can use for the Computer Name. The following characters are not allowed:</p> <pre>whitespace ` ! @ # \$ ^ & * () + = [] { } \ ; : ' " , < > / ?</pre> <p>Other methods you could use to edit the computer name fact might not enforce any restrictions or constraints on the naming. If the naming is invalid, the VM might hang during sysprep.</p>

String in Development Client UI	Fact Name	Required/Optional	Description and Information
<i>Product ID</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.UserData.ProductID" value="" type="String" /></pre>	Effectively required ⁴	<p>The Windows product key. The ID is obtained from the Windows MSDN CD or from Microsoft. The value is used when building a new VM.</p> <p>This fact is optional for PlateSpin Orchestrate, but if the value is not specified, the Windows VM might stop at a user prompt on its console waiting for the entry of the Product ID.</p> <p>Certain versions of Windows, such Windows Server 2008, might not require a product key at installation, but will eventually require it (or a valid license server setup for product activation).</p>
<i>Run Once Command</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.GuiRunOnce.Command" value="" type="String" /></pre>	Optional ²	<p>A list of commands separated by new line characters that run the first time a user logs on after the new VM is created. Commands are scheduled by using the HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce registry key.</p> <p>The value does not need to be specified. It is passed to the sysprep answer file only if one or more commands are specified in the list.</p>
<i>Workgroup</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.Identification.JoinWorkgroup" value="" type="String" /></pre>	Automatic default value provided by Orchestrate ¹	<p>The Windows workgroup name. If the VM is joining a domain, use JoinDomain.</p> <p>Sysprep requires either a domain or a workgroup to be joined. This fact is ignored if the Domain fact and related facts are set, because domain joining takes priority over Workgroup joining.</p> <p>If no domain is being joined, and this fact is not specified, the default value becomes WORKGROUP (default with Windows).</p>

String in Development Client UI	Fact Name	Required/Optional	Description and Information
<i>Domain</i>	<code><fact name="resource.provisioner.autoprep.sysprep.Identification.JoinDomain" value="" type="String" /></code>	Automatic default value provided by Orchestrate ¹	<p>The Windows domain name. If the VM is joining a workgroup, use <code>JoinWorkgroup</code>. For joining a domain, <code>DomainAdmin</code> and <code>DomainAdminPassword</code> must be defined.</p> <p>No default value is provided if this value is not set. Instead, a workgroup is joined with the default name <code>WORKGROUP</code> if no <code>Workgroup</code> fact was set.</p> <p>See also: Domain Admin Password (required if a value is set for this fact).</p>
<i>Domain Admin</i>	<code><fact name="resource.provisioner.autoprep.sysprep.Identification.DomainAdmin" value="" type="String" /></code>	Required ³	<p>Provide a value for this fact when the Domain fact has a value. Configuring this fact allows sufficient privileges to the Windows sysprep program to add the new server or workstation to the domain. Normally, this is the Administrator account for the domain.</p> <p>If the Domain fact does not have a value, this fact is ignored.</p>
<i>Domain Admin Password</i>	<code><fact name="resource.provisioner.autoprep.sysprep.Identification.DomainAdminPassword.value" value="" type="String" /></code>	Required ³	<p>Provide a value for this fact when the Domain fact has a value. Configuring this fact allows sufficient privileges to the Windows sysprep program to add the new server or workstation to the domain. Normally, this is the Administrator password for the domain.</p> <p>If the Domain fact does not have a value, this fact is ignored.</p>

String in Development Client UI	Fact Name	Required/Optional	Description and Information
<i>Domain Admin Password Plaintext</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.Identification.DomainAdminPassword.plainText" value="false" type="Boolean" /></pre>	Automatic default value provided by Orchestrate ¹	When this check box is selected (it has a value of true), DomainAdminPassword is in plaintext. The value defaults to true if it is not set. The value is currently ignored for sysprep, because there are no fields to accept this flag in <code>sysprep.inf</code> or <code>unattend.xml</code> .
<i>License File Automode</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.LicenseFilePrintData.AutoMode" value="" type="String" /></pre>	Automatic default value provided by Orchestrate ¹	The value in this field is either PerServer or PerSeat. If it is set to PerServer, the AutoUsers fact must be set. A value for this fact is required for sysprep on Windows Server products. The provided default is "PerServer."
<i>License File Autousers</i>	<pre><fact name="resource.provisioner.autoprep.sysprep.LicenseFilePrintData.AutoUsers" value="200" type="Integer" /></pre>	Automatic default value provided by Orchestrate ¹	Specify value between 1 and 9999, representing the number of client licenses per seat. A value for this fact is required for sysprep on Windows Server products. The provided default is 5.

¹ Facts with automatic default values must be set in the `sysprep.inf` or `unattend.xml` answer file, but the `vmprep` job provides useful “generic” defaults if any of these facts are not specified. In general, a VM can be successfully personalized using only an Admin Password and Product ID. Some versions of Windows do not require the Product ID if their activation timeout for Windows Product Activation has not yet expired.

² Optional facts are never required. They are not placed in `sysprep.inf` or `unattend.xml` if a value is not specified. For the purpose of this documentation, “optional” also means that sysprep continues to function if these facts are not specified; however, optional facts might be needed as part of a Domain join or a similar function.

³ There is no way to provide default values for required facts, and sysprep fails ungracefully if they are not specified. The `vmprep` job fails a Personalization action if these facts are not set by the user.

⁴ For sysprep, this is the only fact that is “effectively” required. Some versions of Windows can be installed or sysprepped without this value, but most versions stop during sysprep and await user interaction at a Product Key prompt if this value is not specified in `sysprep.inf` or `unattend.xml`.

Configuring vNIC Sysprep Facts

VMs can be prepared for provisioning by configuring the facts in either the *Autoprep Network Adapter* subpanel (Windows VMs) of the vNIC *Info/Groups* panel or the *Sysprep Network Adapter* subpanel (Linux VMs). For more information, see [“Defining Autoprep/Sysprep Network Adapter Facts on the vNIC Object” on page 57](#).

Virtual NIC sysprep facts are always optional. If they are not specified, Windows chooses “typical system” values, including setting vNICs to use DHCP.

6.2.3 Using the Sysprep `deploy.cab` Files

By default, Microsoft does not include the `sysprep.exe` or `setupcl.exe` utilities needed for sysprep on Windows Server 2003, Windows XP, or any previous version. To provide sysprep functionality for VMs running these Windows versions, PlateSpin Orchestrate must have access to compatible `deploy.cab` files from Microsoft. These files can usually be copied directly from the Windows installation CD, or they can be [downloaded from Microsoft](#).

For Windows Vista, Windows Server 2008, and later releases, Microsoft includes the needed sysprep tools on the OS installation and uses a different “answer file” format and utility syntax. For these newer versions of Windows, there is no need to download additional files from Microsoft, or to copy them from the installation CD.

The following instructions apply only if you want to perform sysprep-based personalization on Windows 2003 VMs, Windows XP VMs, or earlier versions of Windows VMs. The instructions include the following:

- ♦ [“`.cab` File Installation Locations” on page 72](#)
- ♦ [“Detailed Instructions for Downloading `.cab` Files From Microsoft” on page 73](#)
- ♦ [“Detailed instructions for Copying `deploy.cab` from the Windows Installation CD or DVD” on page 74](#)

`.cab` File Installation Locations

Assuming a normal install of the Orchestrate Server, the server’s datagrid file tree is located in the `/var/opt/novell/zenworks/zos/server/datagrid/` directory. Copy the `.cab` files to one of the following locations (leaving off the fully qualified portion of the path before `/datagrid`) as appropriate for the Windows server operating system:

```
dataGrid/files/sysprep/winserver2003_sp1/x86/deploy.cab
dataGrid/files/sysprep/winserver2003_r2/x86/deploy.cab
dataGrid/files/sysprep/winserver2003/x86/deploy.cab
dataGrid/files/sysprep/windowsxp/x86/deploy.cab
dataGrid/files/sysprep/windowsxpx64/x86_64/deploy.cab
dataGrid/files/sysprep/winserver2003x64/x86_64/deploy.cab
dataGrid/files/sysprep/winserver2003x64_r2/x86_64/deploy.cab
```

Notice that the files are named according to the `resource.os.type` fact, the `resource.os.arch` fact, and (optionally) whether the VM’s operating system is SP1, R2, or something similar. The file tree in the list above covers all of the common releases of Windows. The Orchestrate sysprep job looks for the datagrid file in the following path:

```
grid:///sysprep/<resource.os.type>_<servicepack>/<resource.os.arch>/
deploy.cab
```


If Orchestrator cannot find the `.cab` file in this path, it looks for the `datagrid` file in the following path:

```
grid:///sysprep/<resource.os.type>/<resource.os.arch>/deploy.cab
```

If you want to install the precise version of `deploy.cab` from your Windows CD, use the above convention to copy it to the `/datagrid` directory.

Through testing, Novell has determined that only two unique `.cab` files are required to support the most common Windows versions. See the Download Instructions exceptions on the Microsoft site for details. This method works because Orchestrator uses only the `sysprep.exe` and `setupcl.exe` executables from the `.cab` files. The other utilities are not used because their purpose is to manually build `sysprep.inf`. Orchestrator automatically builds its own answer file as part of the VM personalization process.

Detailed Instructions for Downloading .cab Files From Microsoft

Use the following steps to download the `.cab` files from Microsoft that you need for sysprep for Xen and Hyper-V VMs.

TIP: You can deploy the `.cab` files using any user account that has admin rights.

- 1** (Conditional) Download the Windows 2003 `.exe` file containing `deploy.cab`.
 - 1a** From a Web browser, navigate to the Microsoft Download Center page entitled *System Preparation tool for Windows Server 2003 Service Pack 2 Deployment* (<http://www.microsoft.com/downloads/en/details.aspx?FamilyID=93f20bb1-97aa-4356-8b43-9584b7e72556&displaylang=en>), then download the Windows 2003 sysprep tool, `WindowsServer2003-KB926028-v2-x86-ENU.exe`.
 - 1b** Copy the `.exe` file to a suitable location on a Windows physical or virtual machine, then run the executable with the `/x` flag (which specifies file extraction only) to extract `deploy.cab` from the executable bundle.
 - 1c** Navigate to the extracted directory where `deploy.cab` is located.
 - 1d** Copy `deploy.cab` to the [appropriate locations](#) on the Orchestrator Server.
- 2** (Conditional) Download the Windows XP `.cab` file:
 - 2a** From a Web browser, navigate to the Microsoft Download Center page entitled *Windows XP Service Pack 2 Deployment Tools* (<http://www.microsoft.com/downloads/en/details.aspx?FamilyId=3E90DC91-AC56-4665-949B-BEDA3080E0F6&displaylang=en>), then download the Windows XP sysprep tool, `WindowsXP-KB838080-SP2-DeployTools-ENU.cab`.
 - 2b** Run the `.cab` file on a Windows physical or virtual machine to extract `deploy.cab`.
 - 2c** Navigate to the extracted directory where `deploy.cab` is located.
 - 2d** Copy `deploy.cab` to the [appropriate locations](#) on the Orchestrator Server.

VM personalization testing using the two versions of `deploy.cab` files listed above has determined that they are suitable for common versions of Windows. When you have placed copies of the `deploy.cab` file in the proper directories of the Orchestrator Server machine, you can perform sysprep personalization on pre-Vista versions of Windows.

You can download other, potentially newer, `deploy.cab` files from Microsoft, but be sure you are familiar with how to use the Microsoft sysprep tools and that the version you download matches the version of your VMs. Make sure you use the file and directory naming conventions explained in this section, so that the personalization system uses the correct `deploy.cab` for the VM being personalized.

Detailed instructions for Copying `deploy.cab` from the Windows Installation CD or DVD

If the version of `deploy.cab` you download from Microsoft is not suitable for the Windows version on your Windows VMs, you can copy `deploy.cab` for the version of Windows server you need directly from the Microsoft installation CD or DVD. The file is normally located in the following path relative to the CD's root directory:

```
support/tools/deploy.cab
```

Copy `deploy.cab` to the correct location in the datagrid file tree of the Orchestrate Server according your Windows version (see [“.cab File Installation Locations” on page 72](#)). For example, if your CD is the `x86_64` version of Windows 2003 Server SP2, copy it to the following location on the Orchestrate Server machine:

```
dataGrid/files/sysprep/winserver2003x64_sp2/x86_64/deploy.cab
```

You can also copy `deploy.cab` to the alternate location used for fall back:

```
dataGrid/files/sysprep/winserver2003x64/x86_64/deploy.cab
```

NOTE: If the `.cab` file you download from Microsoft (see [“Detailed Instructions for Downloading .cab Files From Microsoft” on page 73](#)) causes problems with sysprep on your VM images, using the method of copying `deploy.cab` described in this section might correct compatibility problems.

6.2.4 Applying Sysprep Facts

For vSphere VMs, the `vmprep.job` is not used. Instead, Orchestrate uses VMware Image Customization through the Orchestrate vsphere provisioning adapter's vi-client facility. For more information, see [“Sysprep on VMware VMs” on page 62](#).

For Xen VMs, the PlateSpin Orchestrate Server applies the sysprep facts by launching the `vmprep` job when the facts are defined. This job runs automatically and applies the appropriate facts to a VM in the following situations:

- ◆ When you run a *Personalize* action on any non-templated VM. (See [Table 2-2 on page 17](#)).

On VMs managed by Xen and vSphere, running the *Personalize* action on a templated VM is not supported. Running this action results in failure because it is not supported in the underlying system. When you clone or provision from a templated VM, select the *Use Autoprep* check box.

- ◆ When you create a VM clone by initiating the *Clone* action on a VM template.

You must select the *Use Autoprep* check box in the VM Client or the *Use Autoprep* check box in the Development Client if autoprep facts are to be used when the *Clone* action is initiated.

You need need to make sure that the VM template is set up according to your needs before you clone or provision it, so that the resulting clone meets your needs.

6.2.5 Example Sysprep Scenarios

Scenario 1: You want to create 25 dynamic VM instances to test job provisioning. You will never use these instances again, so you will not personalize them.

You create a VM template by right-clicking a VM, then you select *Create Template*. When the VM Template is created in the Explorer Tree, you define its sysprep facts in the *Info/Groups* page by specifying an asterisk in the *MAC Address* field, then you select the *Use DHCP* check box. This lets the Development Client autogenerate the MAC address and retrieve network data from the DHCP server. For information about setting sysprep facts on each vNIC, see “[The Virtual NIC Info Panel](#)” in the *PlateSpin Orchestrate 2.5 Development Client Reference*.

When the sysprep facts are defined, you provision this template. You right-click the template object and select *Provision*, then in the Provision VM dialog box, you specify that you want to provision (create) 25 new VM instances from this template. Provisioning automatically applies the sysprep facts from the template.

Scenario 2: You have created three VM clones in your grid and you want to provision those clones. You want to ensure that the MAC address and other key network information for each clone is unique, even though each clone is a copy of the same OS image. These clones are to be detached later and used for such things as mail servers and Web servers. When the clones were first created, sysprep facts were applied, but now you have changed those facts by adding static IP addresses, subnet masks, and gateway addresses for each. Each clone must be “personalized” because of this change to basic network identifiers.

To personalize, you select each Clone object, then define the adapter-specific settings on the *Info/Groups* page by entering IP addresses, subnet masks, and gateway addresses for each adapter. When you have defined the sysprep facts on each VM clone, you right-click each Clone object in turn and select *Personalize* to apply the new network configuration.

For more information, see “[Changing a Virtual Machine Template Clone to an Instance](#)” on page 22 and “[Personalize](#)” on page 23.

6.2.6 Known Sysprep Limitations

There are some limitations that you need to be aware of when you use sysprep on VMs:

- ♦ When a VM is sysprepped on vSphere, two reboots are required after the VM is first started in order for Windows to apply VMware and sysprep reconfigurations. Windows VMs managed by Xen and sysprepped also require two reboots.
- ♦ Before performing a sysprep operation on a vSphere VM, you must first configure the VM to have the VMware Tools guest OS package installed. The VMware administrator must also have configured the Virtual Center server to allow sysprep. For more information, see the [VMware vSphere Online Library \(http://pubs.vmware.com/vsp40/wwhelp/wwhimpl/js/html/wwhelp.htm#href=admin/c_installing_the_microsoft_sysprep_tools.html\)](http://pubs.vmware.com/vsp40/wwhelp/wwhimpl/js/html/wwhelp.htm#href=admin/c_installing_the_microsoft_sysprep_tools.html).
- ♦ When a VM is sysprepped in vSphere, it goes into a “customization pending” state that is not cleared until the VM successfully starts and the prep is complete. This means that subsequent attempts to sysprep fail until the VM reboots and the previous prep completes. There is no way to directly reverse customization in vSphere.

- ◆ When you create a template of a VM with the Windows Server 2008 R2 OS, make sure that you configure the sysprep settings for all of its network interfaces using a DHCP connection, not an IP address. This is necessary because of a problem in Windows sysprep that does not remove old IP addresses from the template. The guest OS must not have an IP configured when it is sysprepped.

When the template has been prepared to use DHCP, subsequent syspreps of the clones of that template can use an IP address.

- ◆ Testing has shown that personalizing VMs that have pre-Vista Windows operating systems does not properly configure some network settings. This is a sysprep limitation. The issue is manifest when vNIC-specific settings from sysprep facts on the VM's vNICs are not configured in the VM after personalization.


To work around this issue, personalize by using DHCP settings. You can do this by leaving the fields blank. DHCP is the default for network settings.

- ◆ If a Hyper-V VM is running during the discover process, it fails to discover the `resource.os.family` fact. This prevents the Development Client from displaying the *Sysprep* and *Autoprep* options section on the *Info/Groups* tab for the VM.

NOTE: If the VM not running when the discovery occurs, the hyperv provisioning adapter discovers `resource.os.family` itself.

If you create a template from this VM, the `resource.os.family` fact is discovered and populated on the VM template admin view.

To display the sysprep/autoprep settings on a Hyper-V VM use the following steps:

1. Shut down the Hyper-V VM that has the problem.
 2. From the Explorer Tree, right-click the VM you have shut down, then select *Resync State*.
 3. In the Development Client, Shift+click the Refresh  icon or restart the Development Client to refresh all of the objects and their facts in the Resource admin view.
- ◆ Sysprep does not work on Windows VMs until you set a value for the `Admin Password` fact: `resource.provisioner.autoprep.sysprep.GuiUnattended.AdminPassword.value`. For information about this fact, see [“Admin Password” on page 65](#).
 - ◆ If you clone a Windows Server 2003 VM template originally created in the hypervisor environment, the administrator password for the VM template base image must be blank (no value), or the original VM administrator password is retained and you cannot log in to the cloned VM with the new password.

Attempting to change an old password value by using the `AdminPassword` entry in `sysprep.inf` does not work, but if the original password value was blank, you can use the `AdminPassword` entry in `sysprep.inf` to provide the password value and log in with that password. The value is applied from the

`resource.provisioner.autoprep.sysprep.GuiUnattended.AdminPassword.value` sysprep fact when you select the *Use Autoprep* check box while creating a clone.

- ◆ Some sysprep problems have been noted with Windows Product Activation (WPA) functionality, particularly with versions of Windows that require product activation by the end user.

On Windows Server 2003 SP1, if you have a VM that has passed the initial activation deadline, and you sysprep it, sysprep is applied correctly, but that VM immediately changes to “limited functionality” mode and requires user intervention to reactivate it. Sysprep seems to remove activation and require that the VM be reactivated.

Further, although there is a Boolean value (AutoActivate) that you can set in the “unattended” section of `sysprep.inf`, setting the value does not always result in auto activation of a VM.

To avoid this situation, we recommend that you consider volume licensing or similar licensing solutions available from Microsoft that don’t require manual activation by an actual user. For versions of Windows prior to Windows Vista, this would be a VLK (Volume License Key). For Windows Vista or later, Microsoft has license server-based solutions available to handle volume licensing.

In using sysprep on Windows VMs managed by Xen, no special agent or tools are needed for sysprep because of the method used by the xen provisioning adapter. See [“Sysprep On Xen VMs” on page 62](#) for more information.

Virtual Machine Technologies and Actions

A

The tables in this section contain information about the different PlateSpin Orchestrate VM Management technologies, PlateSpin Orchestrate provisioning adapters, and actions you can perform on and with a VM.

This section contains the following:

- ♦ [Section A.1, “Virtual Machine Technologies,” on page 79](#)
- ♦ [Section A.2, “Xen Hypervisor on SLES 10 SP3 or SLES 11 Host,” on page 81](#)
- ♦ [Section A.3, “VMware vSphere,” on page 83](#)
- ♦ [Section A.4, “Microsoft Hyper-V Hypervisor,” on page 85](#)
- ♦ [Section A.5, “Amazon EC2,” on page 87](#)

A.1 Virtual Machine Technologies

The following table lists the available actions using VMs with PlateSpin Orchestrate:

Table A-1 VM Technologies with Host Operating System, Guest Operating System, and Provisioning Adapter

VM Technology	Host Operating System	Guest Operating System	PlateSpin Orchestrate Provisioning Adapter
Xen 3.3	SLES 11	<ul style="list-style-type: none">♦ SLES 9♦ SLES 10♦ SLES 11♦ RHEL 4♦ RHEL 5♦ Other Linux♦ Windows Server 2003¹♦ Windows XP¹♦ Windows Server 2008¹	xen

VM Technology	Host Operating System	Guest Operating System	PlateSpin Orchestrate Provisioning Adapter
VMware vSphere 4 ²	Subject to the VMware support matrix	<ul style="list-style-type: none"> ◆ SLES 8 ◆ SLES 9 ◆ SLES 10 ◆ RHEL 4 ◆ RHEL 5 ◆ Other Linux ◆ Windows Server 2003 ◆ Windows XP 	vsphere
Microsoft Hyper-V ³	Windows Server 2008 with Hyper-V enabled	<ul style="list-style-type: none"> ◆ Windows 2008 R2 ◆ Windows 2008 (32-bit and 64-bit) ◆ Windows Server 2003 SP2 (32-bit and 64-bit) ◆ Windows 2000 ◆ Windows 7 (32-bit and 64-bit) ◆ Windows Vista (32-bit and 64-bit) ◆ Windows XP Professional (32-bit and 64-bit) ◆ SLES 10 SP1 (32-bit and 64-bit) ◆ SLES 10 SP2 (32-bit and 64-bit) ◆ RHEL 5.x (32-bit and 64-bit) 	hyperv

VM Technology	Host Operating System	Guest Operating System	PlateSpin Orchestrate Provisioning Adapter
Amazon EC2	“Cloud” Web service in Amazon data center.	<p>Instances currently supported are subject to Amazon support matrix. See the EC2 Web site (http://aws.amazon.com/ec2/) for more information.</p> <p>Currently, the Amazon Machine Images (AMIs) that PlateSpin Orchestrate discovers include:</p> <ul style="list-style-type: none"> ◆ Red Hat Enterprise Linux ◆ OpenSolaris ◆ Fedora ◆ Windows Server 2003/2008 ◆ openSUSE Linux Enterprise Server ◆ Gentoo Linux ◆ Oracle enterprise Linux ◆ Ubuntu Linux ◆ Debian 	ec2

¹ Windows VMs running on the Xen hypervisor require a VM host CPU with the Intel VT or AMD-V technology available and enabled.

² The vsphere provisioning adapter does not support VMware Virtual Center 1.x.

³ A complete listing of guest OS support for the Hyper-V hypervisor is available at the [Microsoft TechNet Web site \(http://technet.microsoft.com/en-us/library/cc794868\(WS.10\).aspx\)](http://technet.microsoft.com/en-us/library/cc794868(WS.10).aspx) and at the [Windows Server 2008 Hyper-V product page \(http://www.microsoft.com/windowsserver2008/en-us/hyperv-supported-guest-os.aspx\)](http://www.microsoft.com/windowsserver2008/en-us/hyperv-supported-guest-os.aspx).

A.2 Xen Hypervisor on SLES 10 SP3 or SLES 11 Host

The Xen hypervisor runs on SUSE Linux Enterprise Server (SLES) 10 SP3 and SLES 11 host machines with the PlateSpin Orchestrate Xen Provisioning Adapter (xen). The following table lists the PlateSpin Orchestrate VM actions and whether or not PlateSpin Orchestrate can perform the action on the guest operating system.

Table A-2 VM Actions Supported for SLES 10 SP3 or SLES 11 Host Using the Xen Hypervisor

PlateSpin Orchestrate Managed VM Action	SLES 9 Guest	SLES 10 Guest	RHEL 4 Guest	RHEL 5 Guest	Other Linux Guest	Windows Guest
Provision	X	X	X	X	X	X
Clone	X	X	X	X	X	X
Shutdown	X	X	X	X	X	X
Destroy	X	X	X	X	X	X
Suspend	X	X	X	X	X	X
Pause	X	X	X	X	X	X
Resume	X	X	X	X	X	X
Create Template	X	X	X	X	X	X
Move Disk Image ¹	X	X	X	X	X	X
Hot Migrate ²	X	X	X	X	X	X
Checkpoint	X	X	X	X	X	X
Restore	X	X	X	X	X	X
Install Orchestrate Agent	X	X	X	X		X
Make Standalone	X	X	X	X	X	X
Check Status	X	X	X	X	X	X
Personalize	X	X	X	X		
Save Config	X	X	X	X	X	X
Cancel Action	X	X	X	X	X	X
Check Host Assignment	X	X	X	X	X	X
Build	?	?	?	?	?	?
Launch Remote Desktop	X	X	X	?	X	X

¹ A “move” is the relocation of VM disk images between two storage devices when the VM is in a not running state (this includes VMs that are suspended with a checkpoint file). This action does not require shared storage; the move is between separate repositories.

² A “hot migrate” (also called a “live migrate”) is the migration of a running VM to another host and starting it there with minimal resulting downtime (measured in milliseconds). This function requires shared storage.

A.2.1 Additional Xen Provisioning Adapter Information

Other behaviors of the xen provisioning adapter that you should be aware of include the following:

- ◆ The VM Builder running on a SLES 11 VM host now supports three additional VM OS types:
 - ◆ openSUSE 11

- ◆ SLED 11
- ◆ SLES 11

These OS types are not available for build when you use the VM Builder running on a SLES 10 SP2 VM host. These OS types can only be built and provisioned to a SLES 11 VM host.

- ◆ VMs that were supported on SLES 10 SP2 are supported on SLES 11; that is, if the VM was built on SLES 10 SP2, the provisioning adapter supported for that VM on SLES 10 SP2 can also be supported when the VM runs on SLES 11.
- ◆ Although VM migration from a SLES 11 VM host to a SLES 10 SP2 VM host is not supported, VM migration is supported in the following scenarios:
 - ◆ VM on a SLES 10 SP2 VM host migrating to a SLES 11 VM host
 - ◆ VM on a SLES 10 SP2 VM host migrating to a SLES 10 SP2 VM host
 - ◆ VM on a SLES 11 VM host migrating to a SLES 11 VM host
- ◆ Virtual machines built on SLES 10 SP2 can be provisioned on either SLES 10 SP2 or SLES 11 VM hosts.
- ◆ Virtual machines (regardless of OS type) built on SLES 11 cannot be provisioned on a SLES 10 SP2 VM host.
- ◆ RHEL 5 VMs managed by the xen provisioning adapter must have the following characteristics:
 - ◆ They must not use LVM
 - or
 - ◆ If LVM is used on the VM, its volume groups (VGs) must have unique (that is, non-default) names.

To illustrate what can happen when you use a default LVM configuration, consider the following example:

You create two RHEL 5 VMs. During this process, the default disk configuration (which incorporates LVM) is utilized, so the two VMs have identical VG names. If these VMs are located on the same Xen host, a naming conflict occurs when those VMs are concurrently discovered by PlateSpin Orchestrate. This results in one of the VMs not being properly discovered.

TIP: As a general rule, we do not recommend using LVM for VM disks.

For information about the VMs provisioned by this provisioning adapter, see [Section 4.2, “Configuring the Xen Provisioning Adapter and Xen VMs,”](#) on page 44.

A.3 VMware vSphere

The following table lists the PlateSpin Orchestrate VM actions and whether or not PlateSpin Orchestrate can perform that action on the guest operating system.

IMPORTANT: The PlateSpin Orchestrate Server supports only one VMware vSphere server per grid.

Table A-3 PlateSpin Orchestrate VM Actions and Supported Guest Operating Systems Using VMware Virtual Center

PlateSpin Orchestrate Managed VM Action	SLES 8	SLES 9	SLES 10	RHEL 4	RHEL 5	Other Linux	Windows
Provision	X	X	X	X	X	X	X
Clone	X	X	X	X	X	X	X
Shutdown	X	X	X	X	X	X	X
Destroy	X	X	X	X	X	X	X
Suspend	X	X	X	X	X	X	X
Pause							
Resume							
Create Template	X	X	X	X	X	X	X
Move Disk Image ¹	X	X	X	X	X	X	X
Hot Migrate ²	X	X	X	X	X	X	X
Warm Migrate ³	X	X	X	X	X	X	X
Checkpoint	X	X	X	X	X	X	X
Restore	X	X	X	X	X	X	X
Install Orchestrate Agent	X	X	X	X	X	X	X
Make Standalone	X	X	X	X	X	X	X
Check Status	X	X	X	X	X	X	X
Save Config	X	X	X	X	X	X	X
Cancel Action	X	X	X	X	X	X	X
Check Host Assignment	X	X	X	X	X	X	X
Launch Remote Desktop	X	X	X	X	X	X	X

NOTE: Host operating systems are dependent on the VMware vSphere support matrix.

In the 2.5 release of PlateSpin Orchestrate, the VMware vSphere provisioning adapter supports only VMware Virtual Center 2.x. Virtual Center 1.x is not supported in this release.

¹ A “move” is the relocation of VM disk images between two storage devices when the VM is in a not running state (including VMs that are suspended with a checkpoint file). This action does not require shared storage; the move is between separate repositories.

² A “hot migrate” (also called a “live migrate”) is the migration of a running VM to another host and starting it there with minimal resulting downtime (measured in milliseconds). This action requires shared storage.

³ A “warm migrate” is the migration of a suspended VM to another host and starting it there with brief resulting downtime (measured in seconds). This action requires shared storage.

A.3.1 Additional VMware vSphere Provisioning Adapter Information

The vsphereDiscovery.policy file includes fixed paths the to vpxd.exe file:

```
<fact name="vcenter.path.vpxd">
  <list>
    <listelement value="C:\Program
Files\VMware\Infrastructure\VirtualCenter Server\vpxd.exe" />
    <listelement value="C:\Program Files\VMware\VMware VirtualCenter
2.0\vpxd.exe" />
    <listelement value="C:\Program Files\VMware\VMware
VirtualCenter\vpxd.exe" />
  </list>
</fact>
```

If you have installed vSphere in a location other than those listed as default in the policy file, you can copy one of the three existing listelement values, modify the path to point to their installation location for the vpxd.exe process, and then append this new item to the end of the vcenter.path.vpxd list.

For example, if vpxd.exe is installed at D:\foo\bar\VMware\VMware VirtualCenter2.0\vpxd.exe, you could modify the XML policy file as follows:

```
<fact name="vcenter.path.vpxd">
  <list>
    <listelement value="C:\Program
Files\VMware\Infrastructure\VirtualCenter Server\vpxd.exe" />
    <listelement value="C:\Program Files\VMware\VMware VirtualCenter
2.0\vpxd.exe" />
    <listelement value="C:\Program Files\VMware\VMware
VirtualCenter\vpxd.exe" />
    <listelement value="D:\foo\bar\VMware\Infrastructure\VirtualCenter
Server\vpxd.exe" />
  </list>
</fact>
```

A.4 Microsoft Hyper-V Hypervisor

The following table lists the PlateSpin Orchestrate VM actions and whether or not PlateSpin Orchestrate can perform that action on the guest operating system.

Table A-4 PlateSpin Orchestrate VM Actions Supported on Guest Operating Systems Managed by Hyper-V

PlateSpin Orchestrate Managed VM Action	SLES 10	Windows 2000	Windows Server 2003	Windows Server 2008
Provision	X	X	X	X
Pause	X	X	X	X
Resume	X	X	X	X
Suspend	X	X	X	X
Shutdown	(X) ¹	X	X	X

PlateSpin Orchestrate Managed VM Action	SLES 10	Windows 2000	Windows Server 2003	Windows Server 2008
Shutdown Agent	X	X	X	X
Restart	X	X	X	X
Hot Migrate ²	X	X	X	X
Warm Migrate ³				
Move Disk Images ⁴	X	X	X	X
Resync State ⁵	X	X	X	X
Save Config	X	X	X	X
Apply Config		X	X	X
Create Template		X	X	X
Delete / Destroy Resource ⁶	X	X	X	X
Checkpoint	X	X	X	X
Restore	X	X	X	X
Remove Template Dependency		X	X	X
Install Agent		X	X	X
Personalize		X	X	X
Cancel Action	X	X	X	X
Check Host Assignment	X	X	X	X
Launch Remote Desktop		X	X	X
Create Virtual Disk	X	X	X	X
Create Virtual NIC	X	X	X	X
Delete Virtual Disk	X	X	X	X
Delete Virtual NIC	X	X	X	X

¹ Shutting down the Orchestrate Agent on a SLES 10 VM (guest OS) is possible if the agent is installed manually in the VM.

² A “hot migrate” (also called a “live migrate”) is the migration of a running VM to another host and starting it there with minimal resulting downtime (measured in milliseconds). This action requires shared storage.

³ A “warm migrate” is the migration of a suspended VM to another host and starting it there with brief resulting downtime (measured in seconds). This action requires shared storage.

⁴ A “move” is the relocation of VM disk images between two storage devices when the VM is in a not running state (including VMs that are suspended with a checkpoint file). This action does not require shared storage; the move is between separate repositories.

⁵ In addition to updating the VM’s status, Resync State collects information about the present VM host, even if the VM is migrated to some other host by an external action executed in a Hyper-V clustered environment. It also discovers the `osfamily` and `ostype` fact values.

⁶ The Delete/Destroy Resource action deletes all VM files (including snapshot files) except ISOs that are attached to the image. The action might not destroy the VM file folders if the VM was created in a non-default location.

A.5 Amazon EC2

The Amazon Elastic Compute Cloud (EC2) provides a library of VM images (“machine images” and “machine instances” in Amazon terminology) that you can use to execute work. The PlateSpin Orchestrate ec2 provisioning adapter lets you provision and terminate an Amazon template, also called an Amazon Machine Image or “AMI.” The ec2 provisioning adapter also lets you start, stop, restart, and terminate clones (“instances” in Amazon terminology) of these templates in the EC2 cloud. The provisioning adapter can also manage VMs from multiple EC2 accounts simultaneously.

The following table lists the PlateSpin Orchestrate VM actions and whether or not PlateSpin Orchestrate can perform that action on the guest operating system.

Table A-5 PlateSpin Orchestrate VM Actions Supported on Guest Operating Systems Managed by Amazon EC2

PlateSpin Orchestrate Managed VM Action	Linux Server AMIs	Windows Server 2003/2008 AMIs
Provision (AMI instances)	X	X
Pause		
Resume		
Suspend (Stop)	X	X
Shutdown (Terminate) ¹	X	X
Shutdown Agent		
Restart	X	X
Hot Migrate ²		
Warm Migrate ³		
Move Disk Images ⁴		
Resync State		
Save Config		
Apply Config		
Create Template		
Delete / Destroy Existing Template	X	X
Checkpoint		
Restore		
Remove Template Dependency		

PlateSpin Orchestrate Managed VM Action	Linux Server AMIs	Windows Server 2003/2008 AMIs
---	-------------------	-------------------------------

Install Agent

Personalize

Cancel Action

Check Host Assignment

Launch Remote Desktop

Create Virtual Disk

Create Virtual NIC

Delete Virtual Disk

Delete Virtual NIC

¹ If you try to stop an instance that is backed by an instance store, the instance is stopped with a terminate, rather than a suspend.

² A “hot migrate” (also called a “live migrate”) is the migration of a running VM to another host and starting it there with minimal resulting downtime (measured in milliseconds). This action requires shared storage.

³ A “warm migrate” is the migration of a suspended VM to another host and starting it there with brief resulting downtime (measured in seconds). This action requires shared storage.

⁴ A “move” is the relocation of VM disk images between two storage devices when the VM is in a not running state (including VMs that are suspended with a checkpoint file). This action does not require shared storage; the move is between separate repositories.

Some actions available in Amazon EC2 but not in the PlateSpin Orchestrate ec2 provisioning adapter include:

- ◆ Template creation
- ◆ Reserved and spot instances
- ◆ Key pair and security group management
- ◆ EBS storage management
- ◆ VPN management
- ◆ Cost calculation

For information about configuring the Amazon EC2 provisioning adapter, see [Section 4.3, “Configuring the Amazon EC2 Provisioning Adapter,”](#) on page 45.

Tuning the vsphere Provisioning Adapter for Large Scale VM Provisioning

B

If the vSphere virtual machine environment you manage services a large number of concurrent requests, such as multiple (for example, more than 30) concurrent Business Service Requests for workloads from Novell Cloud Manager, you should tune the vsphere provisioning adapter in PlateSpin Orchestrate 2.5 to accommodate the load.

We recommend that you adjust these policies for large scale concurrent VM provisioning operations.

vsphere policy

Use the following steps to adjust critical performance facts in the policy.

- 1 In the Explorer view of the Orchestrate Development Client, select the vsphere policy to open the policy editor.
- 2 On line 77, change the default value for the `actionTimeout` fact from 2400 to 7200.

```
<fact name="actionTimeout"
      type="Integer"
      value="2400"
      description="Timeout limit for this job to process a PA action
in seconds"/>
```

- 3 Click the *Save* icon.

vspheredaemon policy

Use the following steps to adjust critical facts in the policy.

- 1 In the Explorer view of the Orchestrate Development Client, select the vspheredaemon policy to open the policy editor.
- 2 On line 26, change the default value for the `maxConcurrentTimeout` fact from 3600 to 7200.

```
<fact name="maxConcurrentTimeout"
      type="Integer"
      value="3600"
      description="Timeout limit to wait in seconds for a operation
to be accepted. Wait is due to reaching the maxConcurrentOperation limit"/>
```

- 3 On line 46, change the default value for the `javaArgs` fact from 256 to 1024.

```
<fact name="javaArgs"
      type="String"
      value="-Xmx256m"
      description="JVM args for vsphere client"/>
```

- 4 Click the *Save* icon.

Discovering or Dynamically Creating vSphere Port Groups

C

If you manage a vSphere hypervisor environment, you can configure virtual switches (vSwitches) and distributed virtual switches (dvSwitches) with virtual ports to allow virtual machines on the same ESX Server host to communicate on the network. Your VMware environment also requires port groups, which, according to the [VMware Infrastructure Library \(http://pubs.vmware.com/vi3/serverconfig/wwhelp/wwhimpl/common/html/wwhelp.htm?context=serverconfig&file=sc_networking.5.3.html\)](http://pubs.vmware.com/vi3/serverconfig/wwhelp/wwhimpl/common/html/wwhelp.htm?context=serverconfig&file=sc_networking.5.3.html) “define how a connection is made through the vSwitch to the (physical) network construct.”

You can use PlateSpin Orchestrate to monitor the discovered VMs in your vSphere environment (provided that you have configured the vSwitches or dvSwitches on those VMs), dynamically deleting unused port groups from the virtual switches to free up resources on the vSwitch or dvSwitch so that it can be used elsewhere in the virtual network. Then, if you later decide to provision a VM that requires the port group, Orchestrate can be configured to automatically re-create the required port group. This allows VMs to be moved or migrated from one host to another without reconfiguring the networking information on either the VM or the host.

This section includes the following information:

- ♦ [Section C.1, “How Orchestrate Models a vSphere Port Group and Exposes vSwitch Settings,” on page 91](#)
- ♦ [Section C.2, “Dynamically Creating Port Groups,” on page 91](#)

C.1 How Orchestrate Models a vSphere Port Group and Exposes vSwitch Settings

When PlateSpin Orchestrate discovers a vSphere VM, it maps port group configuration settings (including its associated vSwitch or dvSwitch configuration settings) to a virtual bridge (vBridge) grid object in Orchestrate. As the virtual networking settings (including the VLAN ID) of these resources are determined during VM and VM host discovery, Orchestrate also automatically creates a Network group object to group similarly associated vBridge objects. All necessary information required for re-creating a port group is preserved on the Network object and the vBridge object.

C.2 Dynamically Creating Port Groups

You can instruct PlateSpin Orchestrate to dynamically create a port group in the vSphere environment if a VM that you are provisioning requires it or you can delete an existing port group when the last VM utilizing that port group is shut down. Port groups are dynamically created on the host/dvSwitch only when a VM requiring a port group is provisioned. A port group is deleted only when there are no active VMs utilizing it and if it is not configured as a management network interface. In order for dynamic creation or deletion to function properly, you need to configure the dvSwitches/vSwitches in the vSphere environment.

The vsphere provisioning adapter utilizes information it gathers during discovery and stores on the Network objects to determine how to correctly create a required port group and to attach it to the correct dvSwitches or vSwitches. You can modify this information on the discovered Network objects, or you can add it to Network objects that you customize in the Orchestrate Development Client.

This section includes the following information:

- ♦ [Section C.2.1, “Network Configuration Facts,” on page 92](#)
- ♦ [Section C.2.2, “Enabling Dynamic Creation of Port Groups,” on page 92](#)

C.2.1 Network Configuration Facts

PlateSpin Orchestrate uses the following facts when it creates a port group:

- ♦ **group.dvswitch:** A Boolean fact that specifies if the port group is to be attached to a dvSwitch or to a standard host vSwitch. The value is set to true for a dvSwitch or false if set for a vSwitch.
- ♦ **group.switch:** The name of the dvSwitch or vSwitch that the port group is to be attached to. This fact can be a regular expression to match against different dvSwitches or vSwitches on different hosts
- ♦ **group.vsphere.portgroupsettings:** An XML string that describes the settings on the port group (for example, Security, Traffic Shaping, or Teaming policies). Use the vSphere tools to configure these settings, so that Orchestrate can properly populate the fact upon discovery.
- ♦ **group.vlanid:** The vLAN ID that is associated with the port group. A value of zero (0) is equivalent to no vLAN.

Orchestrate also uses the `group.vbridge.matchname.fact`. This is a regular expression used during discovery of Networks to correlate port groups that are functionally identical (and should therefore be grouped in the same Network object in Orchestrate) but are not named the same. So, if Host A contains a port group named HostA-InternetConnection and Host B contains a port group named HostB-InternetConnection, modifying the `group.vBridge.matchname` fact to “Host?-InternetConnection” results in creating a vBridge object for each host under a single Network object.

C.2.2 Enabling Dynamic Creation of Port Groups

Complete the following steps to enable the dynamic creation of port groups described in [Section C.2, “Dynamically Creating Port Groups,” on page 91](#).

1. Assign the `vsphere_ignoreNetwork.policy` to all vSphere hosts where you want to enable port group creation. This lets the host be considered as a provisioning target, even if the port groups do not currently exist on the host’s dvSwitches or vSwitches.
2. Associate the `vsphere_ignoreNetwork.policy` with all VMs where you want to enable port group creation. This enables the computed facts that are necessary to provide proper ordering of provisioning plans.
3. In the `vsphere.policy`, update the `vsphere.vcenters` fact by setting the `auto_portgroups_creation` setting to true.

Modifying the `auto_portgroups_deletion` and `auto_portgroups_disconnection` settings is not necessary to enable dynamic creation of port groups, but is necessary for automatically reclaiming the port group resources when they are not in use.

Documentation Updates

D

This section contains information about documentation content changes that were made in this *PlateSpin Orchestrate Development Client Reference* after the initial release of PlateSpin Orchestrate 2.5. The changes are listed according to the date they were published.

The documentation for this product is provided on the Web in two formats: HTML and PDF. The HTML and PDF documentation are both kept up-to-date with the changes listed in this section.

If you need to know whether a copy of the PDF documentation that you are using is the most recent, the PDF document includes a publication date on the title page.

The documentation was updated on the following dates:

- ♦ [Section D.1, “March 11, 2011,” on page 93](#)
- ♦ [Section D.2, “November 12, 2010,” on page 93](#)
- ♦ [Section D.3, “October 15, 2010,” on page 93](#)

D.1 March 11, 2011

Updates were made to the following sections:

Location	Update Description
Appendix C, “Discovering or Dynamically Creating vSphere Port Groups,” on page 91	New content.

D.2 November 12, 2010

Updates were made to the following sections:

Location	Update Description
Section 2.2, “Provisioning Actions and History,” on page 14.	Moved this content from an appendix in the <i>PlateSpin Orchestrate 2.5 Development Client Reference</i> .

D.3 October 15, 2010

Updates were made to the following sections:

Location	Update Description
“Detailed Instructions for Downloading .cab Files From Microsoft” on page 73	Added detail to this section to distinguish between Windows XP and Windows 2003 deployment procedures.
Section A.2.1, “Additional Xen Provisioning Adapter Information,” on page 82	Added information cautioning against using XEN VMs created with LVM, in particular a RHEL 5 VM.

Location	Update Description
Appendix B, "Tuning the vsphere Provisioning Adapter for Large Scale VM Provisioning," on page 89	New documentation.
