



Red Hat CloudForms 4.6

Provisioning Virtual Machines and Hosts

Provisioning, workload management, and orchestration for Red Hat CloudForms

Red Hat CloudForms 4.6 Provisioning Virtual Machines and Hosts

Provisioning, workload management, and orchestration for Red Hat CloudForms

Red Hat CloudForms Documentation Team
cloudforms-docs@redhat.com

Legal Notice

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This guide provides instructions for provisioning, service creation, and automation in Red Hat CloudForms. If you have a suggestion for improving this guide or have found an error, please submit a Bugzilla report at <http://bugzilla.redhat.com> against Red Hat CloudForms Management Engine for the Documentation component. Please provide specific details, such as the section number, guide name, and CloudForms version so we can easily locate the content.

Table of Contents

CHAPTER 1. RED HAT CLOUDFORMS LIFECYCLE	5
1.1. PROVISIONING	5
CHAPTER 2. PROVISIONING REQUESTS	6
2.1. REQUIREMENTS FOR PROVISIONING VIRTUAL MACHINES AND INSTANCES	6
2.2. REQUIREMENTS FOR PROVISIONING VIRTUAL MACHINES FROM RED HAT VIRTUALIZATION MANAGER	6
2.3. PXE PROVISIONING	7
2.3.1. Connecting to a PXE Server	8
2.3.2. Creating System Image Types for PXE	9
2.3.3. Setting the PXE Image Type for a PXE Image	10
2.4. ISO PROVISIONING	11
2.4.1. Adding an ISO Datastore	11
2.4.2. Refreshing an ISO Datastore	11
2.4.3. Creating System Image Types for ISO	12
2.4.4. Setting the Image Type for an ISO Image	12
2.5. CUSTOMIZATION TEMPLATES FOR VIRTUAL MACHINE AND INSTANCE PROVISIONING	13
2.6. CUSTOMIZATION SCRIPT ADDITIONS FOR VIRTUAL MACHINE AND INSTANCE PROVISIONING	14
2.7. ADDING A CUSTOMIZATION TEMPLATE	14
2.8. PROVISIONING VIRTUAL MACHINES	15
2.8.1. Provisioning a Virtual Machine from a Template	15
2.8.2. Cloning a Virtual Machine	21
2.8.3. Publishing a Virtual Machine to a Template (VMware Virtual Machines Only)	21
2.9. PROVISIONING INSTANCES	22
2.9.1. Provisioning an EC2 Instance from an Image	22
2.9.2. Provisioning an OpenStack Instance from an Image	23
2.9.3. Provisioning a Google Compute Engine Instance from an Image	25
2.9.4. Requirements for Provisioning a Host	26
2.9.4.1. IPMI Hosts	27
2.9.4.1.1. Discovering the Management Interface for an IPMI Host	27
2.9.4.1.2. Adding IPMI Credentials to a Discovered Host	27
2.9.4.1.3. Adding the Management Interface for an IPMI Host	28
2.9.4.2. Customization Templates for Host Provisioning	28
2.9.4.3. Customization Script Additions	28
2.9.4.3.1. Adding a Customization Template	30
2.9.5. Provisioning a Host	30
2.9.6. Customizing Provisioning Dialogs	33
2.9.6.1. Adding a Provision Dialog for All Users	34
2.9.6.2. Creating a Custom Provision Dialog	34
2.9.7. Provisioning Profiles	35
2.9.7.1. Creating a Provisioning Profile Instance	36
2.9.7.2. Setting Provisioning Scope Tags	37
2.9.8. Managing Key Pairs	38
CHAPTER 3. WORKING WITH REQUESTS	39
3.1. PROVISIONING REQUEST APPROVAL METHODS	39
3.2. WORKING WITH PROVISIONING REQUESTS	39
3.2.1. Reloading the Status of Provisioning Requests	39
3.2.2. Approving a Provisioning Request	39
3.2.3. Denying a Provisioning Request	39
3.2.4. Copying a Provisioning Request	40
3.2.5. Editing a Provisioning Request	40

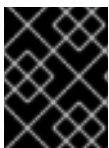
3.2.6. Deleting a Provisioning Request	40
3.2.7. Automatically Approving Requests	40
3.2.7.1. Enabling Global Defaults for Automatic Approval	40
3.2.7.2. Template Specific Approval Defaults	42
3.2.7.3. Assigning Tags to a Template for Auto Approval	42
3.2.7.4. Setting Provisioning Notification Email Addresses	42
CHAPTER 4. FULFILLING REQUESTS	44
4.1. FULFILLING A REQUEST	44
4.2. DEFAULT EXECUTION STEPS IN STATES INSTANCE	44
4.2.1. Quotas	44
4.2.1.1. Applying User or Group Quotas	45
4.2.1.2. Using Tags for Owner and Group Quotas	45
4.2.1.3. Applying a Tag to a User or User Group	46
4.2.1.4. State Machines	46
4.2.1.5. Customizing Provisioning States	48
4.2.1.6. Editing the Default State Instance	49
4.2.1.7. Viewing the Status of a Provisioning Request	50
4.2.1.8. Viewing a Provisioned Virtual Machine or Instance	50
4.2.1.9. Viewing a Virtual Machine or Instance Summary	50
CHAPTER 5. CATALOGS AND SERVICES	51
5.1. GENERIC OBJECTS	51
5.1.1. Viewing Generic Objects Classes	52
5.1.2. Creating Generic Objects Classes	52
5.1.3. Editing Generic Object Classes	52
5.1.4. Removing Generic Objects Classes	53
5.2. SERVICE DIALOGS	53
5.2.1. Adding a Service Dialog	53
5.2.2. Adding a Service Dialog Using Dialog Editor	55
5.2.3. Creating a Service Dialog from a Container Template	57
5.2.4. Importing Service Dialogs	58
5.2.5. Exporting Service Dialogs	58
5.3. METHODS	58
5.3.1. Creating a Method to Associate with the Dialog	58
5.3.2. Creating a Method in the Service Class	61
5.3.3. Creating an Instance in the Service Class	61
5.3.4. Associating a Method with an Automate Instance	62
5.4. CATALOGS	63
5.4.1. Creating a Catalog Bundle	63
5.4.2. Creating a Catalog Item	64
5.4.3. Creating a Generic Catalog Item	65
5.4.4. Creating an Ansible Playbook Service Catalog Item	66
5.4.5. Creating an Ansible Tower Service Catalog Item	68
5.4.6. Creating an Amazon Service Catalog Item	69
5.4.7. Creating an Azure Service Catalog Item	69
5.4.8. Creating an OpenShift Template Catalog Item	70
5.4.9. Creating an Orchestration Catalog Item	70
5.4.10. Provisioning a Service	71
5.5. ORCHESTRATION STACKS	71
5.5.1. Creating an Orchestration Template	72
5.5.2. Editing Orchestration Templates	72
5.5.3. Copying Orchestration Templates	72

5.5.4. Deleting Orchestration Templates	73
CHAPTER 6. RETIREMENT	74
6.1. RETIRING VIRTUAL MACHINES	74
6.1.1. Retiring Virtual Machines and Instances	74
6.1.2. Using the Console to Retire a Virtual Machine	74
6.1.3. Retiring a Virtual Machine Immediately	74
6.1.4. Setting a Retirement Date and Time for a Virtual Machine or Instance	74
6.1.5. Removing a Retirement Date for a Virtual Machine or Instance	75
6.2. SETTING RETIREMENT IN A PROVISION REQUEST	75
6.2.1. Scheduling Retirement in a Provision Request	75
6.3. EXTENDING RETIREMENT DATES	76
6.3.1. Creating a Custom Button to Extend Retirement	76
6.3.2. Changing the Number of Days to Extend Retirement	76

CHAPTER 1. RED HAT CLOUDFORMS LIFECYCLE

This guide discusses lifecycle activities such as provisioning and retirement that are part of the Red Hat CloudForms Automate component. Red Hat CloudForms Automate enables real-time, bidirectional process integration and adaptive automation for management events and administrative or operational activities.

- Operations Management with service level resource enforcement.
- Resource Management including datastore cleanup, snapshot aging and enforcement, and virtual machine or instance aging and retirement.
- Configuration and Change Management including enforced closed loop change management.
- Lifecycle Management such as provisioning, customization, reconfiguration, approval, CMDB updates, and retirement.

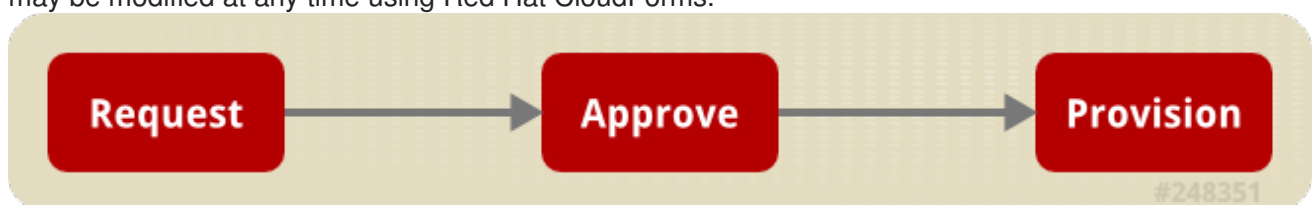


IMPORTANT

Provisioning requires the **Automation Engine** server role enabled. Check your server role settings in the settings menu, **Configuration** → **Server** → **Server Control**.

1.1. PROVISIONING

When a virtual machine or cloud instance is provisioned, it goes through multiple phases. First, the request must be made. The request includes ownership information, tags, virtual hardware requirements, the operating system, and any customization of the request. Second, the request must go through an approval phase, either automatic or manual. Finally, the request is executed. This part of provisioning consists of pre-processing and post-processing. Pre-processing acquires IP addresses for the user, creates CMDB instances, and creates the virtual machine or instance based on information in the request. Post-processing activates the CMDB instance and emails the user. The steps for provisioning may be modified at any time using Red Hat CloudForms.



CHAPTER 2. PROVISIONING REQUESTS

The following options are available when making provisioning requests:

- Set an owner (User can do this using LDAP lookup)
- Assign a purpose (tag)
- Select a template or image from which to create a new virtual machine or instance respectively
- Choose placement
- Set hardware requirements
- Specify the vLan
- Customize the guest operating system
- Schedule the provisioning



2.1. REQUIREMENTS FOR PROVISIONING VIRTUAL MACHINES AND INSTANCES

Red Hat CloudForms supports the provisioning of VMware ESX hosts/hypervisors. To provision a virtual machine from VMware providers, you must have an appliance with the Automation Engine role enabled.

If you are using a Windows template, the following configuration is required:

- To customize settings that are inside the operating system, Sysprep must be copied to the appropriate directory on your vCenter computer. Usually this location is: **C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\sysprep**. Copy the Sysprep tools to the relevant operating system subdirectory. If you are running a standard Win2008 operating system, this step is unnecessary as Sysprep is included as standard.
- The Windows template must have the latest version of VMware tools for its ESX Server. Check the VMware Site for more information. If you are creating a new password for the Administrator account, the Administrator's password must be blank on the template. This is a limitation of Microsoft Sysprep.

See the VMware documentation for a complete list of customization requirements.

2.2. REQUIREMENTS FOR PROVISIONING VIRTUAL MACHINES FROM RED HAT VIRTUALIZATION MANAGER

Item	Requirements
------	--------------

Item	Requirements
Red Hat Virtualization Manager version 3.0 or higher	Red Hat Virtualization Manager properly installed with API in default location https://server:8443/api
Red Hat Virtualization Manager History Database	<p>Red Hat Virtualization Manager Data Warehouse (DWH) properly installed with access to the PostgreSQL database on the Red Hat Virtualization Manager server. Port 5432 open in iptables.</p> <p>md5 authentication allowed to Red Hat CloudForms appliances in pg_hba.conf.</p> <p>PostgreSQL set to listen for connections on *:5432 in postgresql.conf.</p> <p>Credentials provided during database setup to be used in Red Hat CloudForms UI.</p>
Storage Supported for Red Hat CloudForms Virtual Machine Analysis	<p>NFS - Red Hat CloudForms server must be able to mount NFS storage domain.</p> <p>iSCSI / FCP - Cluster must use full Red Hat Enterprise Linux (not Red Hat Virtualization Hypervisor) Hosts.</p> <p>DirectLUN Hook installed on each host and registered to Red Hat Virtualization Managers.</p> <p>Must have Red Hat CloudForms appliance in each Cluster with this storage type.</p> <p>Red Hat CloudForms appliance virtual machine container must have DirectLUN attribute set.</p> <p>Local storage - Not yet supported (Red Hat does not recommend due to single point of failure).</p>

2.3. PXE PROVISIONING

PXE is a boot method that allows you to load files from across a network link. Red Hat CloudForms uses it for files required for provisioning virtual machines. PXE can be used for provisioning for either Red Hat Virtualization Manager or VMware.

Procedure Overview

1. Connect to the **PXE Server**.
2. Create a **System Image Type**.
3. Associate each **PXE** image with an image type.
4. Create a customization template.

Requirements for PXE Provisioning

- DHCP server configured with required PXE implementation
- PXE implementation for Linux virtual machine provisioning
- NFS or SAMBA read and write access to create and modify files on the PXE server
- Red Hat CloudForms Server uses NFS mount to read and write the response files
- HTTP read access to the NFS share location as virtual machines use this URL to access PXE images and Kickstart or Cloud-Init configuration files
- Operating system installation media available to be streamed from PXE server
- Images configured for desired operating systems
- Kickstart or Cloud-Init templates to configure operating systems with desired packages

Additional Requirements for Provisioning Linux Virtual Machines



- Linux distribution kernel and ramdisk available over HTTP
- Linux sources available over HTTP
- Sample PXE menu item that boots this kernel

Additional Requirements for Provisioning Windows Virtual Machines

- WinPE ISO built with rhev-agent-tools (for RHEV-M environments) and configured to mount shares for Windows source files and Sysprep files and configured to run customization script
- Windows based WIM file with operating system installed and configured with Sysprep
- Sample Sysprep unattend file to be used with the operating system
- Sample PXE menu item that downloads WinPE ISO, mount in memdisk and boot into WinPE environment

2.3.1. Connecting to a PXE Server



The following procedure connects to a PXE server and adds its details to Red Hat CloudForms.

1. Navigate to **Compute** → **Infrastructure** → **PXE**.
2. Click  (**Configuration**), then  (**Add a New PXE Server**).

3. In **Basic Information**, type a **Name** that will be meaningful in your environment.

Basic Information	
Name	Mahwah
Depot Type	Network File System
URI	nfs:// 192.168.252.60/srv/httpboot/ipx
Access URL	http://192.168.252.60/ipxe
PXE Directory	mac
Windows Images Directory	sources/microsoft
Customization Directory	customization

PXE Image Menus	
Filename	menu.php

4. For **Depot Type**, select either **Network File System** (NFS) or **Samba**. The fields to enter in the dialog depend on the **Depot Type**.
- For NFS, type in the **URI**, **Access URL**, **PXE Directory**, **Windows Images Directory**, and **Customization Directory**. When you provision, Red Hat CloudForms writes a text file to the **PXE Directory**. The file is named after the MAC address of the NIC that is assigned to the virtual machine. It contains where to get the kernel and initrd image. This file is removed after a successful provision. The **Windows Images Directory** is where the files are located on your NFS for the provisioning of Windows operating systems. The **Customization Directory** is where your Kickstart and Sysprep files are located.
 - If using a **Depot Type** of **Samba**, you will not need **Access URL**, but you will need a **User ID**, and **Password**, in addition to the items required for NFS.
5. For **PXE Image Menus**, type the **Filename** for the PXE Boot menu.
6. Click **Add**.
7. Select the new PXE server from the tree on the left, and click  (**Configuration**), then  (**Refresh**) to see your existing images.

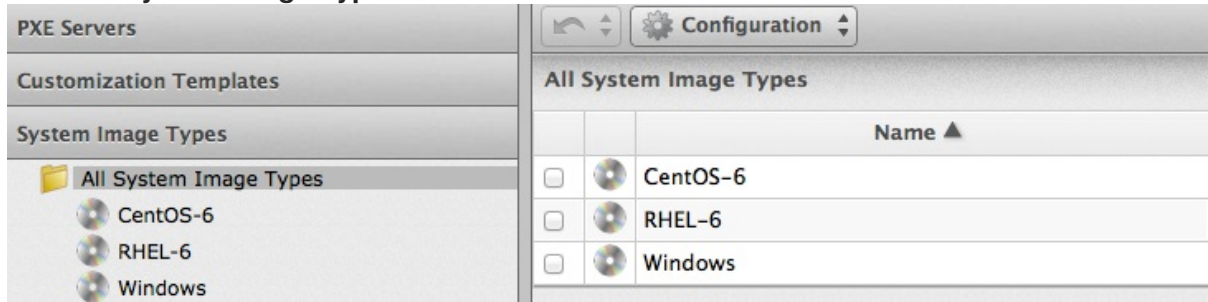
Next, create PXE Image types to associate with the customization templates and to specify if the image type is for a virtual machine, a host, or both.

2.3.2. Creating System Image Types for PXE

The following procedure creates a system image type for PXE servers.

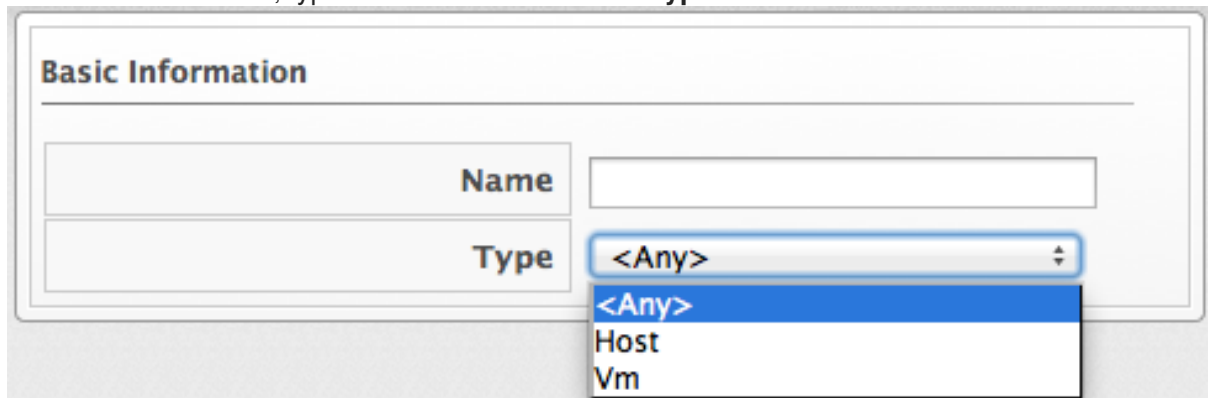
1. Navigate to **Compute** → **Infrastructure** → **PXE**.

2. Click the **System Image Types** accordion.



3. Click  (**Configuration**), then  (**Add a new System Image Type**).

4. In **Basic Information**, type in a **Name** and select a **Type**.



- Use **Host** if you want this image type to only apply to hosts.
- Use **Vm** if you want this image type to only apply to virtual machines.
- Use **Any** if this image type can be used for either hosts or virtual machines.

5. Click **Add**.



After creating the System Image Types, assign the types to each image on your PXE servers. To do this, you will select each image on the PXE server and identify its type.

2.3.3. Setting the PXE Image Type for a PXE Image

The following procedure sets the image type for a chosen PXE image.

1. Navigate to **Compute** → **Infrastructure** → **PXE**.

2. Click the **PXE Servers** accordion and select the image that you want to set a type for.

3. Click  (**Configuration**), then  (**Edit this PXE Image**).

4. From the **Basic Information** area, select the correct type. If this PXE image will be used as the **Windows Boot Environment**, check **Windows Boot Environment**. At the time of this writing, only one PXE Image can be identified as the **Windows Boot Environment**. Therefore, checking one as the **Windows Boot Environment**, will remove that from any other PXE image with that check.

Basic Information

Type	Windows
Windows Boot Environment	<input checked="" type="checkbox"/> * Checking this box will remove this setting from all other PXE Images on this PXE Server

Click **Save**.

PXE Servers

- All PXE Servers
 - PXE Server
 - PXE Images
 - centos63
 - rhel63server
 - winpex64
 - Windows Images
 - win2008r2

PXE Image "winpex64"

✓ PXE Image "winpex64" was saved

Basic Information

Name	winpex64
Description	WindowsPE_amd64
Type	Windows
Kernel	http://\$[next-server]/ipxe/sources/misc/memdisk
Windows Boot Environment	Yes

2.4. ISO PROVISIONING

Red Hat CloudForms also allows ISO provisioning from Red Hat Virtualization Manager datastores. To use this feature, you will need to do the following before creating a provision request.

1. Add the **ISO Datastore**. The Red Hat Virtualization Manager system must have already been discovered or added into the VMDB. For more information, see [Adding a Red Hat Enterprise Virtualization Manager Provider](#) in *Managing Providers*.
2. Refresh the **ISO Datastore**.
3. Create a **System Image Type**.
4. Set the **ISO Image Type**.
5. **Create** a customization template.

2.4.1. Adding an ISO Datastore

The following procedure adds an ISO Datastore from your Red Hat Virtualization environment.

1. Navigate to **Compute** → **Infrastructure** → **PXE**.
2. Click the **ISO Datastores** accordion.
3. Click  (**Configuration**),  (**Add a new ISO Datastore**).
4. Select the Cloud or Infrastructure provider hosting the ISO Datastore.
5. Click **Add**.



The ISO datastore is added to Red Hat CloudForms.

2.4.2. Refreshing an ISO Datastore

The following procedure refreshes the chosen ISO datastore and updates Red Hat CloudForms with available ISOs.



1. Navigate to **Compute** → **Infrastructure** → **PXE**.

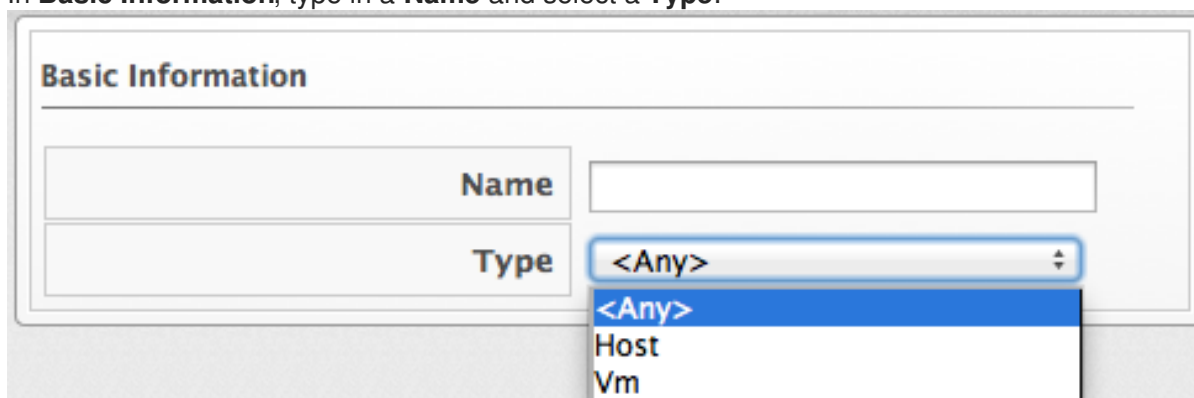
- Click the **ISO Datastores** accordion, and select an ISO datastore.

- Click  (**Configuration**), then click  (**Refresh Relationships**).

2.4.3. Creating System Image Types for ISO

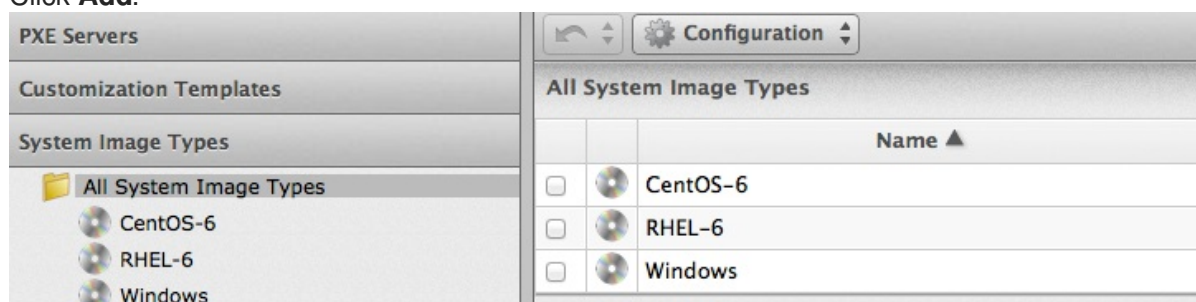
The following procedure creates a system image type for ISO Servers.




- Navigate to **Compute** → **Infrastructure** → **PXE**.
- Click the **System Image Types** accordion.
- Click  (**Configuration**), then  (**Add a new System Image Type**).
- In **Basic Information**, type in a **Name** and select a **Type**.



- Use **Host** if you want this image type to only apply to hosts.
- Use **Vm** if you want this image type to only apply to virtual machines.
- Use **Any** if this image type can be used for either hosts or virtual machines.

- Click **Add**.




All System Image Types		
		Name ▲
<input type="checkbox"/>		CentOS-6
<input type="checkbox"/>		RHEL-6
<input type="checkbox"/>		Windows

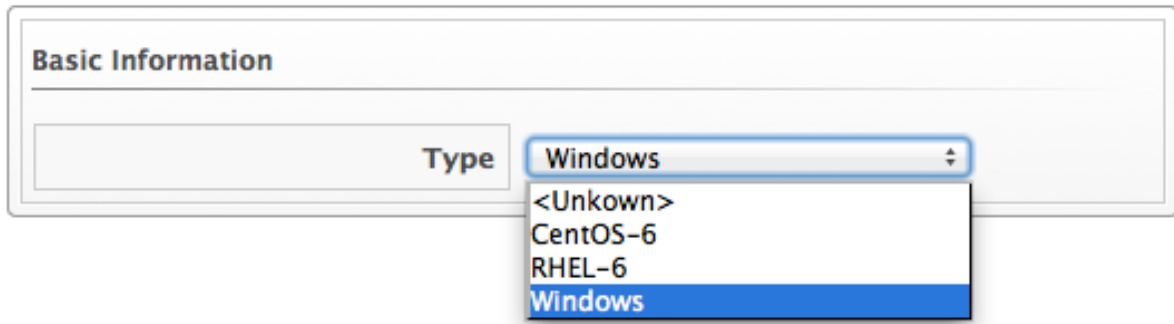
After creating the system image types, assign the types to each image on your ISO servers. To do this, you will select each image on the ISO server and identify its type.

2.4.4. Setting the Image Type for an ISO Image

The following procedure sets the image type for an ISO image.

- Navigate to **Compute** → **Infrastructure** → **PXE**.
- Click the **ISO Datastores** accordion, and select the image that you want to set a type for.

3. Click  (**Configuration**), then  (**Edit this ISO Image**).
4. From the **Basic Information** area, select the correct **Type**.



The screenshot shows a 'Basic Information' configuration window. Inside, there is a 'Type' dropdown menu. The dropdown is open, displaying a list of operating system types: '<Unkown>', 'CentOS-6', 'RHEL-6', and 'Windows'. The 'Windows' option is currently selected and highlighted with a blue background.

5. Click **Save**.

2.5. CUSTOMIZATION TEMPLATES FOR VIRTUAL MACHINE AND INSTANCE PROVISIONING

Add a customization template to provide **Kickstart**, **Cloud-Init**, or **Sysprep** files for the initial loading of the operating system.

Cloud-Init Requirements

- When creating a template using Red Hat Virtualization, install the **cloud-init** package on the source virtual machine. This enables Cloud-Init to source configuration scripts when a virtual machine built on that template boots.
- See [Using Cloud-Init to Automate the Configuration of Virtual Machines](#) in the Red Hat Virtualization *Administration Guide* for more information on using Cloud-Init in a Red Hat Virtualization environment.
- See the [Cloud-Init Documentation](#) web site for example scripts.

Kickstart Requirements for ISO Provisioning

- The **Kickstart** file must be named **ks.cfg**.
- Set the new virtual machine to power down after provisioning is complete.
- CloudForms must use the virtual machine payload feature of Red Hat Virtualization to create a floppy disk containing the data from the selected customization template.
- Customize the installer to include the data written to the floppy disk payload.

Example 2.1. RHEL ISO with the following modifications:

- **isolinux.cfg** – add **ks=cdrom** to the **append** line
- **ks.cfg** – which must minimally include:

```
### Pre Install Scripts
%pre

# Mount the floppy drive
```



```
modprobe floppy
mkdir /tmp/floppy
mount /dev/floppy /tmp/floppy
%end
```

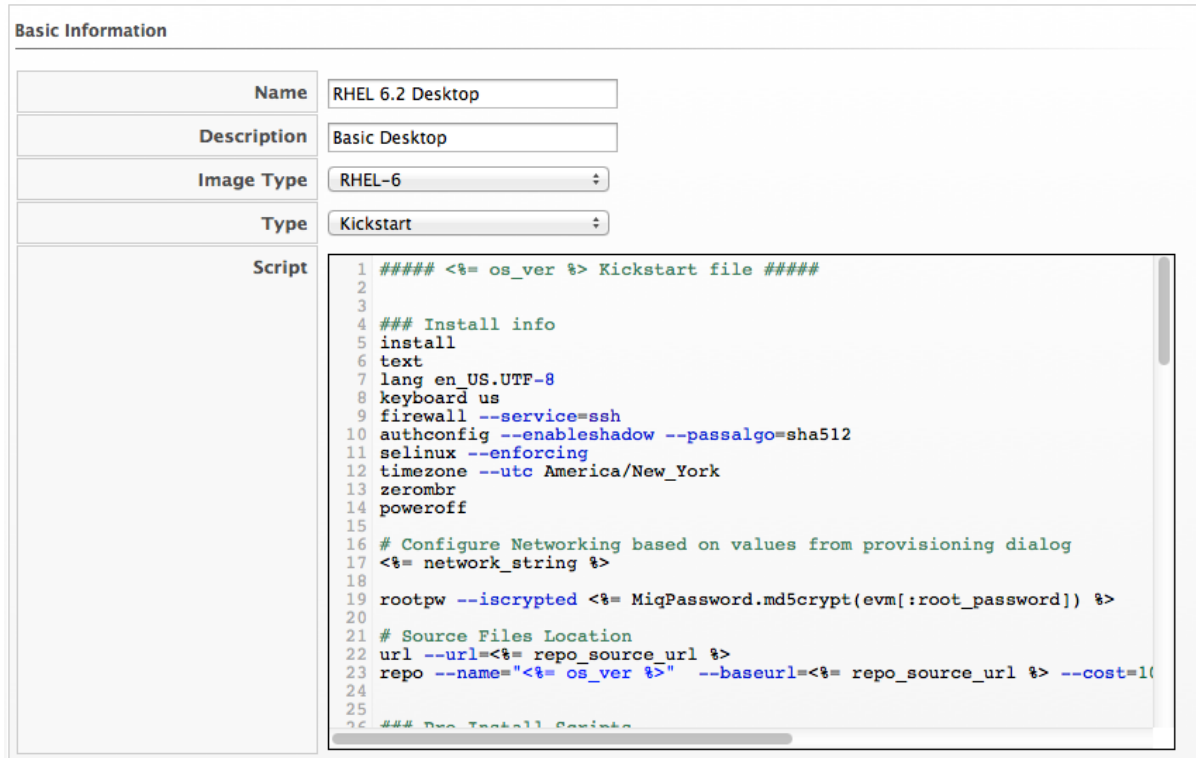
```
# Include ks.cfg file from the floppy (written by CFME based on selected
customization template)
%include /tmp/floppy/ks.cfg
```

2.6. CUSTOMIZATION SCRIPT ADDITIONS FOR VIRTUAL MACHINE AND INSTANCE PROVISIONING

Customization Type	Reason to Include	Script entries
Kickstart	Takes the values from the Customize tab in Provisioning Dialog and substitutes them into the script.	Configure Networking based on values from provisioning dialog <% if evm[:addr_mode].first == 'static' %> <% network_string = "network --onboot yes --device=eth0 --bootproto=static --noipv6" %> <% ["ip", :ip_addr, "netmask", :subnet_mask, "gateway", :gateway, "hostname", :hostname, "nameserver", :dns_servers].each_slice(2) do ks_key, evm_key %> <% network_string << " -{ks_key} #{evm[evm_key]}" unless evm[evm_key].blank? %> <% end %> <%= network_string %> <% else %> network --device=eth0 --bootproto=dhcp <% end %>
Kickstart	Encrypts the root password from the Customize tab in the Provisioning Dialog .	rootpw --iscrypted <%= MiqPassword.md5crypt(evm[:root_password]) %>
Kickstart	Sends status of the provision back to Red Hat CloudForms Server for display in the Red Hat CloudForms Console.	
Sysprep	Encrypts the root password from the Customize tab in the Provisioning Dialog . The value for the AdministratorPassword line must be inserted to use the password from the Provisioning Dialog and encrypt it.	<UserAccounts> <AdministratorPassword> <Value><%= MiqPassword.sysprep_crypt(evm[:root_password]) %></Value> <PlainText>false</PlainText> </AdministratorPassword> </UserAccounts>

2.7. ADDING A CUSTOMIZATION TEMPLATE

1. Navigate to **Compute** → **Infrastructure** → **PXE**.
2. Click the **Customization Templates** accordion.
3. Click  (**Configuration**),  (**Add a new Customization Template**).
4. In **Basic Information**, type in a **Name** and **Description**.



Basic Information

Name	RHEL 6.2 Desktop
Description	Basic Desktop
Image Type	RHEL-6
Type	Kickstart
Script	<pre> 1 ##### <%= os_ver %> Kickstart file ##### 2 3 4 ### Install info 5 install 6 text 7 lang en_US.UTF-8 8 keyboard us 9 firewall --service=ssh 10 authconfig --enablesshadow --passalgo=sha512 11 selinux --enforcing 12 timezone --utc America/New_York 13 zerombr 14 poweroff 15 16 # Configure Networking based on values from provisioning dialog 17 <%= network_string %> 18 19 rootpw --iscrypted <%= MiqPassword.md5crypt(evm[:root_password]) %> 20 21 # Source Files Location 22 url --url=<%= repo_source_url %> 23 repo --name="<%= os_ver %>" --baseurl=<%= repo_source_url %> --cost=10 24 25 26 ### Post Install Scripts </pre>

5. Select the **Image Type**. This list should include the PXE image types you created.
6. In **Type**, select **Kickstart** or **CloudInit** for Linux based systems, and **Sysprep** for Windows based system.
7. In the **Script** area, either paste the script from another source or type the script directly into the Red Hat CloudForms interface.
8. Click **Add**.

The default dialogs show all possible parameters for provisioning. To limit the options shown, see [Section 2.9.6, “Customizing Provisioning Dialogs”](#).

2.8. PROVISIONING VIRTUAL MACHINES

There are three types of provisioning requests available in Red Hat CloudForms:

1. Provision a new virtual machine from a template
2. Clone a virtual machine
3. Publish a virtual machine to a template

2.8.1. Provisioning a Virtual Machine from a Template



You can provision virtual machines through various methods. One method is to provision a virtual machine directly from a template stored on a provider.

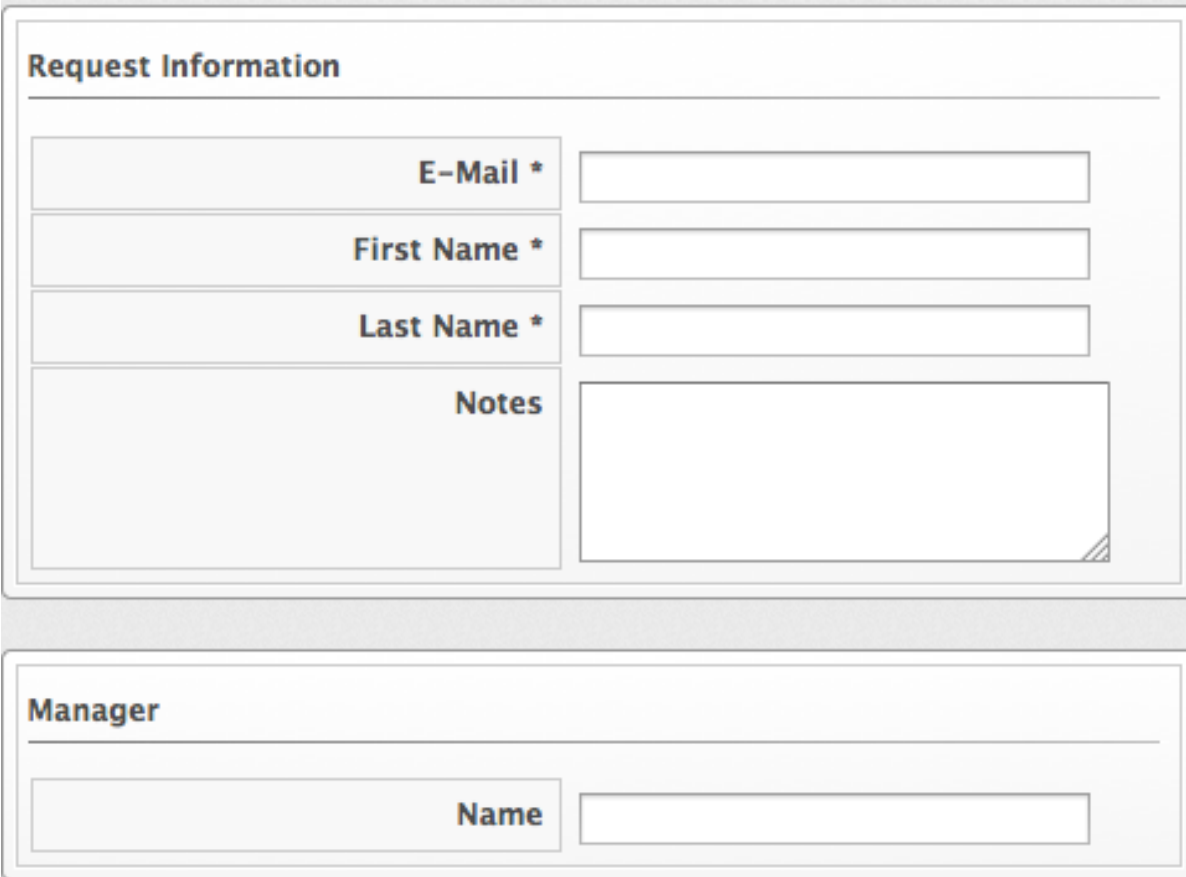


IMPORTANT

To provision a virtual machine, you must have the "Automation Engine" role enabled.

To provision a virtual machine from a template:

1. Navigate to **Compute** → **Infrastructure** → **Virtual Machines**.
2. Click  (**Lifecycle**), and then  (**Provision VMs**).
3. Select a template from the list.
4. Click **Continue**.
5. On the **Request** tab, enter information about this provisioning request.



The screenshot shows a web form for provisioning a virtual machine. It is divided into two main sections: "Request Information" and "Manager".

Request Information

E-Mail *	<input type="text"/>
First Name *	<input type="text"/>
Last Name *	<input type="text"/>
Notes	<input type="text"/>

Manager

Name	<input type="text"/>
-------------	----------------------

In **Request Information**, type in at least a **First Name** and **Last Name** and an email address. This email is used to send the requester status emails during the provisioning process for items such as auto-approval, quota, provision complete, retirement, request pending approval, and request denied. The other information is optional. If the Red Hat CloudForms server is configured to use LDAP, you can use the **Look Up** button to populate the other fields based on the email address.



NOTE

Parameters with a * next to the label are required to submit the provisioning request. To change the required parameters, see [Section 2.9.6, “Customizing Provisioning Dialogs”](#).

6. Click the **Purpose** tab to select the appropriate tags for the provisioned virtual machines.
7. Click the **Catalog** tab to select the template to provision from. This tab is context sensitive based on provider.
8. For templates on VMware providers:

The screenshot shows the VMware provisioning interface. It is divided into three main sections: **Select**, **Number of VMs**, and **Naming**.

Select Section:

- Filter:** A dropdown menu set to "<ALL>".
- Name *:** A search input field.
- Table:** A table with columns: Name, Operating System, Platform, CPUs, Memory, Disk Size, Provider, and Snapshots.

Name	Operating System	Platform	CPUs	Memory	Disk Size	Provider	Snapshots
XD-Win7 Template	Microsoft Windows 7 (64-bit)	windows	1	2 GB	30 GB	Virtual Center (10.17.70.14)	0
- Provision Type *:** A dropdown menu set to "VMware".
- Linked Clone:** A checkbox labeled "VM requires a snapshot" which is currently unchecked.

Number of VMs Section:

- Count:** A dropdown menu set to "1".

Naming Section:

- VM Name *:** An input field.
- VM Description (0 / 100):** A text area.

- a. For **Provision Type**, select **VMware** or **PXE**.
 - i. If **VMware** is selected, select **Linked Clone** to create a linked clone to the virtual machine instead of a full clone. Since a snapshot is required to create a linked clone, this box is only enabled if a snapshot is present. Select the snapshot you want to use for the linked clone.
 - ii. If **PXE** is selected, select a PXE **Server** and **Image** to use for provisioning.
 - b. Under **Count**, select the number of virtual machines to create in this request.
 - c. Use **Naming** to specify a virtual machine name and virtual machine description. When provisioning multiple virtual machines, a number will be appended to the virtual machine name.
9. For templates on Red Hat providers:
 - a. Select the **Name** of a template to use.
 - b. For **Provision Type**, select either **ISO**, **PXE**, or **Native Clone**. You must select **Native Clone** in order to use a Cloud-Init template.
 - i. If **Native Clone** is selected, select **Linked Clone** to create a linked clone to the virtual machine instead of a full clone. This is equivalent to *Thin Template Provisioning* in Red Hat Virtualization. Since a snapshot is required to create a linked clone, this box is only

enabled if a snapshot is present. Select the snapshot to use for the linked clone.

- ii. If **ISO** is selected, select an ISO **Image** to use for provisioning.
 - iii. If **PXE** is selected, select a PXE **Server** and **Image** to use for provisioning.
 - c. Under **Count**, select the number of virtual machines you want to create in this request.
 - d. Use **Naming** to specify a **VM Name** and **VM Description**. When provisioning multiple virtual machines, a number will be appended to the **VM Name**.
10. Click the **Environment** tab to decide where you want the new virtual machines to reside.
- a. If provisioning from a template on VMware, you can either let Red Hat CloudForms decide for you by checking **Choose Automatically**, or select a specific cluster, resource pool, folder, host, and datastore. VMware virtual machines can also be provisioned to a clustered datastore by selecting it under **Datastore**. Additionally, you can assign a storage profile to a VMware virtual machine under **Datastore** to configure the virtual machine to operate using a storage profile from that datastore.
Note, read-only datastores are excluded when provisioning a virtual machine.
 - b. If provisioning from a template on Red Hat, you can either let Red Hat CloudForms decide for you by checking **Choose Automatically**, or select a datacenter, cluster, host and datastore.
11. Click the **Hardware** tab to set hardware options.

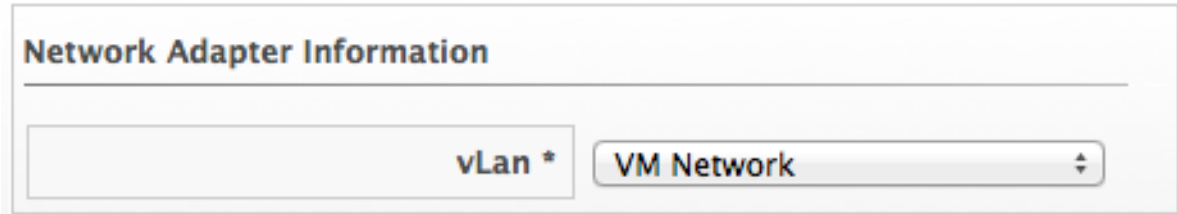
Hardware	
Number of Sockets	<input type="text" value="1"/>
Cores per Socket	<input type="text" value="1"/>
Memory (MB)	<input type="text" value="1024"/>
Disk Format	<input type="radio"/> Thick <input type="radio"/> Thin <input checked="" type="radio"/> Default

VM Limits	
CPU (MHz)	<input type="text" value="-1"/> (-1 = Unlimited)
Memory (MB)	<input type="text" value="-1"/> (-1 = Unlimited)

VM Reservations	
CPU (MHz)	<input type="text" value="0"/>
Memory (MB)	<input type="text" value="0"/>

- a. In **Hardware**, set the number of sockets, cores per socket, memory in MB, and disk format: thin, pre-allocated/thick or same as the provisioning template (default).
- b. For VMware provisioning, set the **VM Limits** of CPU and memory the virtual machine can use.
- c. For VMware provisioning, set the **VM Reservation** amount of CPU and memory.

12. Click **Network** to set the vLan adapter. Additional networking settings that are internal to the operating system appear on the **Customize** tab.



Network Adapter Information

vLan * VM Network

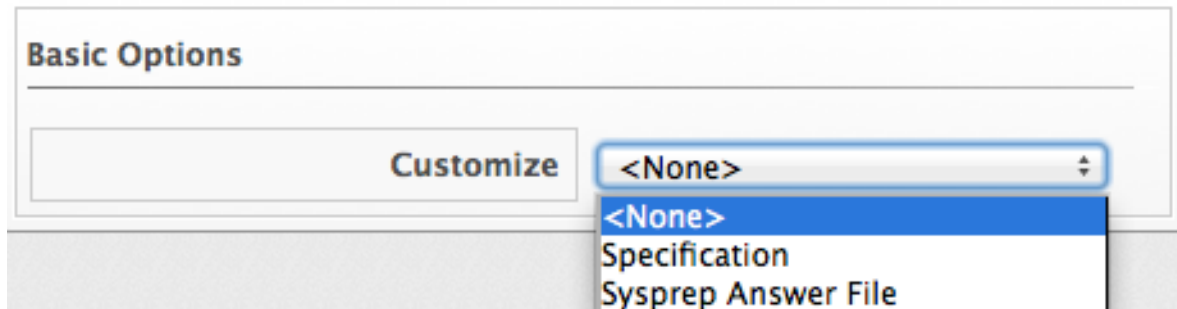
- a. In **Network Adapter Information**, select the **vLan**.



NOTE

A VMware virtual machine can be provisioned to a DVPortgroup by selecting it from the **vLan** list. Prior to provisioning a virtual machine, the DVPortgroup must be created on a vSphere Distributed Switch (VDS) in VMware vCenter in order for Red Hat CloudForms to list the DVPortgroup under **vLan**.

13. Click **Customize** to customize the operating system of the new virtual machine. These options vary based on the operating system of the template.



Basic Options

Customize <None>

<None>
Specification
Sysprep Answer File

14. For Windows provisioning:
 - a. To use a custom specification from the provider, click **Specification**. To select an appropriate template, choose from the list in the custom specification area. The values that are honored by Red Hat CloudForms display.



NOTE

Any values in the specification that do not show in the Red Hat CloudForms console's request dialogs are not used by Red Hat CloudForms. For example, for Windows operating systems, if you have any run once values in the specification, they are not used in creating the new virtual machines. Currently, for a Windows operating system, Red Hat CloudForms honors the unattended GUI, identification, workgroup information, user data, windows options, and server license. If more than one network card is specified, only the first is used.

Basic Options

Customize Specification

Custom Specification

Name	Name	Description	Last Updated
	<None>		
	GM Win2k3 - 0 nics		2012-04-13 15:47:13 UTC
	GM Win2k3 - 1 nics		2012-04-13 13:15:43 UTC
	GM Win2k3 - 2 nics		2012-04-12 15:12:35 UTC
	Win2k8Template		2011-05-17 15:54:37 UTC

Override Specification Values ☐

To modify the specification, select **Override Specification Values**.

- b. Select **Sysprep Answer File**, to upload a Sysprep file or use one that exists for a custom specification on the Provider where the template resides. To upload a file, click **Browse** to find the file, and then upload. To use an answer file in **Customization Specification**, click on the item. The answer file will automatically upload for viewing. You cannot make modifications to it.
15. For Linux provisioning:
 - a. Under **Credentials**, enter a **Root Password** for the **root** user to access the instance.
 - b. Enter a **IP Address Information** for the instance. Leave as **DHCP** for automatic IP assignment from the provider.
 - c. Enter any **DNS** information for the instance if necessary.
 - d. Select **Customize Template** for additional instance configuration. Select from the Kickstart or Cloud-Init customization templates stored on your appliance.
 16. Click the **Schedule** tab to select when provisioning begins.
 - a. In **Schedule Info**, select when to start provisioning. If you select **Schedule**, you will be prompted to enter a date and time. Select **Stateless** if you do not want the files deleted after the provision completes. A stateless provision does not write to the disk so it requires the PXE files on the next boot.
 - b. In **Lifespan**, select to power on the virtual machines after they are created, and to set a retirement date. If you select a retirement period, you will be prompted for when you want a retirement warning.

Provision Virtual Machines

Request Purpose Catalog Environment Hardware Network Schedule

Schedule Info

When to Provision

☐ Schedule
 ☒ Immediately on Approval

Stateless

☐

Lifespan

Power on virtual machines after creation

☒

Time until Retirement

Indefinite

17. Click **Submit**.

The provisioning request is sent for approval. For the provisioning to begin, a user with the administrator, approver, or super administrator account role must approve the request. The administrator and super administrator roles can also edit, delete, and deny the requests. You will be able to see all provisioning requests where you are either the requester or the approver.



After submission, the appliance assigns each provision request a **Request ID**. If an error occurs during the approval or provisioning process, use this ID to locate the request in the appliance logs. The Request ID consists of the region associated with the request followed by the request number. As regions define a range of one trillion database IDs, this number can be several digits long.

Request ID Format



Request 99 in region 123 results in Request ID 123000000000099.

2.8.2. Cloning a Virtual Machine

Virtual machines can be cloned in other providers as well.

1. Navigate to **Compute** → **Infrastructure** → **Virtual Machines**, and check the virtual machine you want to clone.
2. Click  (**Lifecycle**), and then  (**Clone selected item**).
3. Enter the requested information in the dialogs. Be sure to check the **Catalog** tab.
4. Schedule the request on the **Schedule** tab.
5. Click **Submit**.



2.8.3. Publishing a Virtual Machine to a Template (VMware Virtual Machines Only)

1. Navigate to **Compute** → **Infrastructure** → **Virtual Machines**, and check the virtual machine you want to publish as a template.
2. Click  (**Lifecycle**), and then  (**Publish selected VM to a Template**).
3. Enter the requested information in the dialogs. Be sure to check the **Catalog** tab.
4. Schedule the request on the **Schedule** tab.
5. Click **Submit**.

2.9. PROVISIONING INSTANCES

Cloud instances follow the same process (Request, Approval, Deployment) as a standard virtual machine from virtualization infrastructure. First, a user makes a request for instances and specifies the image, tags, availability zone and hardware profile flavor. Second, the request goes through the approval phase. Finally, Red Hat CloudForms executes the request.

2.9.1. Provisioning an EC2 Instance from an Image

1. Navigate to **Compute** → **Clouds** → **Instances**.
2. Click  (**Lifecycle**), then click  (**Provision Instances**).
3. Select an image from the list presented.
4. Click **Continue**.
5. On the **Request** tab, enter information about this provisioning request. In **Request Information**, type in at least a first and last name and an email address. This email is used to send the requester status emails during the provisioning process for items such as auto-approval, quota, provision complete, retirement, request pending approval, and request denied. The other information is optional. If the Red Hat CloudForms Server is configured to use LDAP, you can use the **Look Up** button to populate the other fields based on the email address.



NOTE

Parameters with a * next to the label are required to submit the provisioning request. To change the required parameters, see [Section 2.9.6, “Customizing Provisioning Dialogs”](#).

6. Click the **Purpose** tab to select the appropriate tags for the provisioned instance.
7. Click the **Catalog** tab for basic instance options.
 - a. To change the image to use as a basis for the instance, select it from the list of images.
 - b. Select the **Number of VMs** to provision.
 - c. Type a **VM Name** and **VM Description**.
8. Click the **Environment** tab to select the instance's **Availability Zone**, **Virtual Private Cloud**, **Cloud Subnet**, **Security Groups**, and **Elastic IP Address**. If no specific availability zone is required, select the **Choose Automatically** checkbox.

9. Click the **Properties** tab to set provider options such as hardware flavor and security settings.
 - a. Select a flavor from the **Instance Type** list.
 - b. Select a **Guest Access Key Pair** for access to the instance.
 - c. Select the **CloudWatch** monitoring level. Leave as **Basic** for the default EC2 monitoring.
10. Click the **Customize** tab to set additional instance options.
 - a. Under **Credentials**, enter a **Root Password** for the **root** user access to the instance.
 - b. Enter a **IP Address Information** for the instance. Leave as **DHCP** for automatic IP assignment from the provider.
 - c. Enter any **DNS** information for the instance if necessary.
 - d. Select a **Customize Template** for additional instance configuration. Select from the Cloud-Init scripts stored on your appliance.
11. Click the **Schedule** tab to set the provisioning and retirement date and time.
 - a. In **Schedule Info**, choose whether the provisioning begins upon approval, or at a specific time. If you select **Schedule**, you will be prompted to enter a date and time.
 - b. In **Lifespan**, select whether to power on the instances after they are created, and whether to set a retirement date. If you select a retirement period, you will be prompted for when to receive a retirement warning.
12. Click **Submit**.



The provisioning request is sent for approval. For the provisioning to begin, a user with the admin, approver, or super admin account role must approve the request. The admin and super admin roles can also edit, delete, and deny the requests. You will be able to see all provisioning requests where you are either the requester or the approver.

After submission, the appliance assigns each provision request a **Request ID**. If an error occurs during the approval or provisioning process, use this ID to locate the request in the appliance logs. The Request ID consists of the region associated with the request followed by the request number. As regions define a range of one trillion database IDs, this number can be several digits long.

Request ID Format

Request 99 in region 123 results in Request ID 123000000000099.

2.9.2. Provisioning an OpenStack Instance from an Image

1. Navigate to **Compute** → **Clouds** → **Instances**.
2. Click  (**Lifecycle**), then click  (**Provision Instances**).
3. Select an OpenStack image from the list presented. These images must be available on your OpenStack provider.
4. Click **Continue**.
5. On the **Request** tab, enter information about this provisioning request. In **Request Information**,

type in at least a first and last name and an email address. This email is used to send the requester status emails during the provisioning process for items such as auto-approval, quota, provision complete, retirement, request pending approval, and request denied. The other information is optional. If the Red Hat CloudForms Server is configured to use LDAP, you can use the **Look Up** button to populate the other fields based on the email address.



NOTE

Parameters with a * next to the label are required to submit the provisioning request. To change the required parameters, see [Section 2.9.6, “Customizing Provisioning Dialogs”](#).

6. Click the **Purpose** tab to select the appropriate tags for the provisioned instance.
7. Click the **Catalog** tab for basic instance options.
 - a. To change the image to use as a basis for the instance, select it from the list of images.
 - b. Select the **Number of Instances** to provision.
 - c. Type a **Instance Name** and **Instance Description**.
8. Click the **Environment** tab to select the instance's **Tenant**, **Availability Zones**, **Cloud Network**, **Security Groups**, and **Public IP Address**. If no specific Tenant is required, select the **Choose Automatically** checkbox.
9. Click the **Properties** tab to set provider options such as flavors and security settings.
 - a. Select a flavor from the **Instance Type** list.
 - b. Select a **Guest Access Key Pair** for access to the instance. For more information about key pairs, see [Section 2.9.8, “Managing Key Pairs”](#).
10. Click the **Volumes** tab to provision any volumes with the instance. Volumes are useful for augmenting ephemeral storage of instances with persistent, general-purpose block storage:
 - a. Fill in the **Volume Name** and **Size (gigabytes)** fields.
 - b. If you want the volume to be deleted once the instance terminates (thereby making it non-persistent), check **Delete on Instance Terminate**.
 - c. To provision and add multiple volumes to the instance, click **Add Volume**. Doing so will add new fields you can fill in.
For more information about persistent storage in OpenStack, see the Red Hat OpenStack Platform *Storage Guide*.
11. Click the **Customize** tab to set additional instance options.
 - a. Under **Credentials**, enter a **Root Password** for the **root** user access to the instance.
 - b. Enter a **IP Address Information** for the instance. Leave as **DHCP** for automatic IP assignment from the provider.
 - c. Enter any **DNS** information for the instance if necessary.
 - d. Select a **Customize Template** for additional instance configuration. Select from the Cloud-Init scripts stored on your appliance.

12. Click the **Schedule** tab to set the provisioning and retirement date and time.
 - a. In **Schedule Info**, choose whether the provisioning begins upon approval, or at a specific time. If you select **Schedule**, you will be prompted to enter a date and time.
 - b. In **Lifespan**, select whether to power on the instances after they are created, and whether to set a retirement date. If you select a retirement period, you will be prompted for when to receive a retirement warning.
13. Click **Submit**.

The provisioning request is sent for approval. For the provisioning to begin, a user with the admin, approver, or super admin account role must approve the request. The admin and super admin roles can also edit, delete, and deny the requests. You will be able to see all provisioning requests where you are either the requester or the approver.



After submission, the appliance assigns each provision request a **Request ID**. If an error occurs during the approval or provisioning process, use this ID to locate the request in the appliance logs. The Request ID consists of the region associated with the request followed by the request number. As regions define a range of one trillion database IDs, this number can be several digits long.

Request ID Format

Request 99 in region 123 results in Request ID 123000000000099.

2.9.3. Provisioning a Google Compute Engine Instance from an Image

From Red Hat Red Hat CloudForms 4.2, a Google Compute Engine instance can be provisioned. Some functionality available in other cloud providers is currently unavailable when provisioning from Google Compute Engine.

1. Navigate to **Compute** → **Clouds** → **Instances**.
2. Click  (**Lifecycle**), then click  (**Provision Instances**).
3. Select an image from the list presented.
4. Click **Continue**.
5. On the **Request** tab, enter information about this provisioning request. In **Request Information**, type in at least a first and last name and an email address. This email is used to send the requester status emails during the provisioning process for items such as auto-approval, quota, provision complete, retirement, request pending approval, and request denied. The other information is optional. If the Red Hat CloudForms Server is configured to use LDAP, you can use the **Look Up** button to populate the other fields based on the email address.



NOTE

Parameters with a * next to the label are required to submit the provisioning request. To change the required parameters, see [Section 2.9.6, “Customizing Provisioning Dialogs”](#).

6. Click the **Purpose** tab to select the appropriate tags for the provisioned instance.
7. Click the **Catalog** tab for basic instance options.

- a. To change the image to use as a basis for the instance, select it from the list of images.
 - b. Select the **Number of Instances** to provision.
 - c. Type a **Instance Name** and **Instance Description**. For Google Compute Engine instances, the **Instance Name** cannot use upper case letters. Use a combination of lower case letters and numbers to create the name.
8. Click the **Environment** tab to select the instance's **Availability Zone** and **Cloud Network**. If no specific availability zone is required, select the **Choose Automatically** checkbox.
 9. Click the **Properties** tab to set provider options such as hardware flavor and boot disk settings.
 - a. Select a flavor from the **Instance Type** list.
 - b. Select a **Boot Disk Size** from the list.
 10. Click the **Schedule** tab to set the provisioning and retirement date and time.
 - a. In **Schedule Info**, choose whether the provisioning begins upon approval, or at a specific time. If you select **Schedule**, you will be prompted to enter a date and time.
 - b. In **Lifespan**, select and whether to set a retirement date. If you select a retirement period, you will be prompted for when to receive a retirement warning.
 11. Click **Submit**.

The provisioning request is sent for approval. For the provisioning to begin, a user with the admin, approver, or super admin account role must approve the request. The admin and super admin roles can also edit, delete, and deny the requests. You will be able to see all provisioning requests where you are either the requester or the approver.

After submission, the appliance assigns each provision request a **Request ID**. If an error occurs during the approval or provisioning process, use this ID to locate the request in the appliance logs. The Request ID consists of the region associated with the request followed by the request number. As regions define a range of one trillion database IDs, this number can be several digits long.

Request ID Format

Request 99 in region 123 results in Request ID 123000000000099.

2.9.4. Requirements for Provisioning a Host

Red Hat CloudForms can provision hosts using PXE and Intelligent Platform Management Interface (IPMI) technologies. Before you provision your first host, configure the following prerequisites:



- Provisioning requires the **Automation Engine** server role be enabled. Confirm your server role settings in the settings menu, then **Configuration** → **Server** → **Server Control**.
- Make a PXE server accessible to the Red Hat CloudForms server.
- Create a customization template for hosts. This customization template must contain host-specific additions, documented in the Customization Templates for Host Provisioning section.
- Create system image types for the host.
- Associate images with the image types.

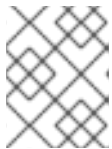
- Enable IPMI on provisioning hosts and add them to the Red Hat CloudForms Infrastructure.

2.9.4.1. IPMI Hosts

There are two ways to get the Intelligent Platform Management Interface (IPMI) Host into the VMDB. You can either use the Red Hat CloudForms's discovery process or add the host using its IP address and credentials.

2.9.4.1.1. Discovering the Management Interface for an IPMI Host

1. Navigate to **Compute** → **Infrastructure** → **Hosts**.
2. Click  (**Configuration**), then  (**Discover Items**).
3. In **Discover**, check **IPMI**.
4. Optionally, in **IPMI Credentials**, type in a **User ID** and **Password**.



NOTE

You can also add IPMI credentials after the host has been discovered. See section [Section 2.9.4.1.2, “Adding IPMI Credentials to a Discovered Host”](#).

5. In **Subnet Range**, type in a range of IP addresses. For quickest results, use the actual IP address in both fields.
6. Click **Start**.





NOTE



After the host is discovered, you can add credentials for IPMI.

2.9.4.1.2. Adding IPMI Credentials to a Discovered Host

After discovering an IPMI host, add the credentials using the following procedure.

1. Navigate to **Compute** → **Infrastructure** → **Hosts**.
2. Click on the host you want to edit.
3. Click  (**Configuration**), and then  (**Edit this Host**).
4. In the **Credentials** area, **IPMI** tab, type in the IPMI credentials
 - a. Use **User ID** to specify a login ID.
 - b. Use **Password** to specify the password for the user ID.
 - c. Use **Verify Password** to confirm the password.
5. Click **Validate** to test the credentials.
6. Click **Save**.

2.9.4.1.3. Adding the Management Interface for an IPMI Host

- 1. Navigate to **Compute** → **Infrastructure** → **Hosts**.
- 2. Click  (**Configuration**), then  (**Add a New Item**).
- 3. In **Basic Information**, type in a **Name** and the **IPMI IP address**.
- 4. In the **Credentials** area, under **IPMI** tab, type in the IPMI credentials
 - a. Use **User ID** to specify a login ID.
 - b. Use **Password** to specify the password for the User ID.
 - c. Use **Verify Password** to confirm the password.
- 5. Click **Validate** to test the credentials.
- 6. Click **Add**.

The IPMI host is added to the Red Hat CloudForms environment; an operating system can now be provisioned onto it.

2.9.4.2. Customization Templates for Host Provisioning



Add a customization template to provide Kickstart files for the initial loading of the operating system. There are certain sections to use to allow for interactions with the provisioning dialogs provided by Red Hat CloudForms.

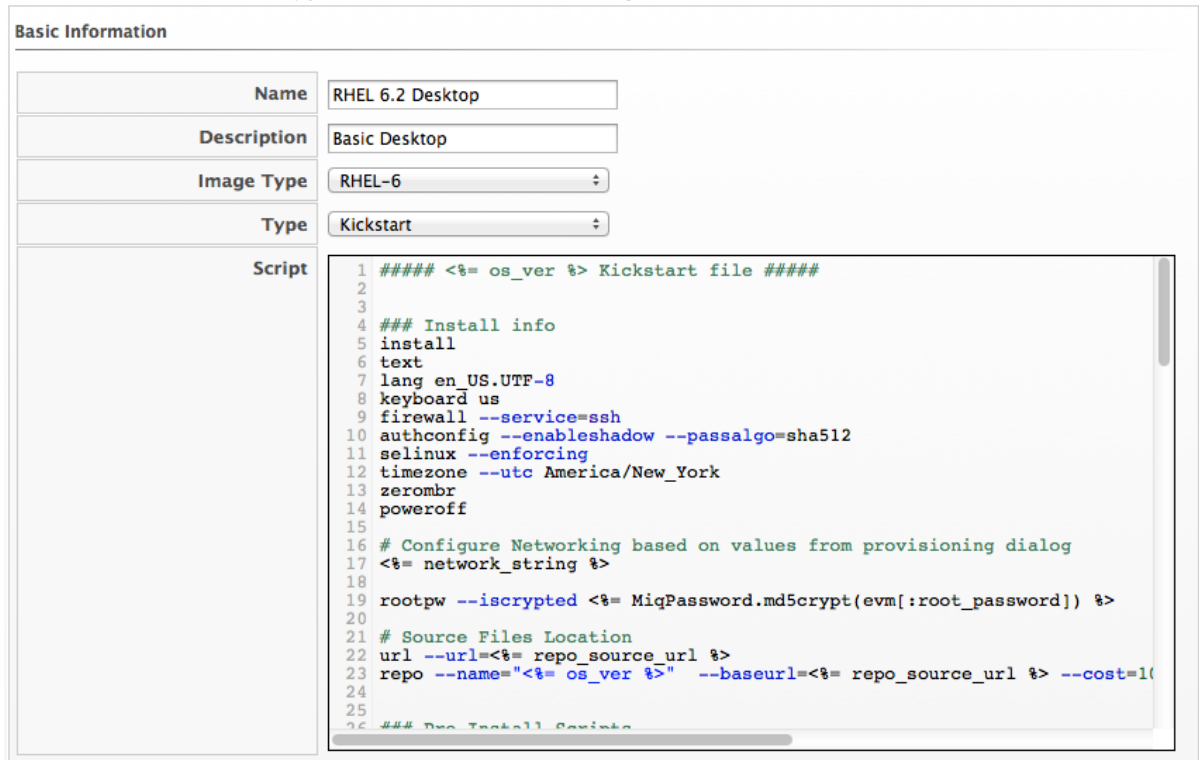
2.9.4.3. Customization Script Additions

Customization Type	Reason to Include	Script entries
--------------------	-------------------	----------------

Customization Type	Reason to Include	Script entries
Kickstart	Takes the values from the Customize tab in the Provisioning Dialog and substitutes them into the script.	<pre>#Configure Networking based on values from provisioning dialog <% if evm[:addr_mode].first == 'static' %> <% network_string = "network --onboot yes --device=eth0 --bootproto=static -- noipv6" %> <% ["ip", :ip_addr, "netmask", :subnet_mask, "gateway", :gateway, "hostname", :hostname, "nameserver", :dns_servers].each_slice(2) do ks_key, evm_key %> <% network_string << " --#{ks_key} # {evm[evm_key]}" unless evm[evm_key].blank? %> <% end %> <%= network_string %> <% else %> network --device=eth0 --bootproto=dhcp <% end %></pre>
Kickstart	Encrypts the root password from the Customize tab in the Provisioning Dialog .	<pre>rootpw --iscrypted <%= MiqPassword.md5crypt(evm[:root_password]) %></pre>
Kickstart	Sends status of the provision back to Red Hat CloudForms Server for display in the Red Hat CloudForms Console.	<pre># Callback to CFME during post-install wget --no-check-certificate <%= evm[:post_install_callback_url] %></pre>
Sysprep	Encrypts the root password from the Customize tab in the Provisioning Dialog . The value for the AdministratorPassword line must be inserted to use the password from the Provision Dialog and encrypt it.	<pre><UserAccounts> <AdministratorPassword> <Value><%= MiqPassword.sysprep_crypt(evm[:root_passw ord]) %></Value> <PlainText>false</PlainText> </AdministratorPassword> </UserAccounts></pre>

2.9.4.3.1. Adding a Customization Template

1. Navigate to **Compute** → **Infrastructure** → **PXE**.
2. Click the **Customization Templates** accordion.
3. Click  (**Configuration**), then  (**Add a New Customization Template**).
4. In **Basic Information**, type in a **Name** and **Description**.



Basic Information

Name	RHEL 6.2 Desktop
Description	Basic Desktop
Image Type	RHEL-6
Type	Kickstart
Script	<pre> 1 ##### <%= os_ver %> Kickstart file ##### 2 3 4 ### Install info 5 install 6 text 7 lang en_US.UTF-8 8 keyboard us 9 firewall --service=ssh 10 authconfig --enableshadow --passalgo=sha512 11 selinux --enforcing 12 timezone --utc America/New_York 13 zerombr 14 poweroff 15 16 # Configure Networking based on values from provisioning dialog 17 <%= network_string %> 18 19 rootpw --iscrypted <%= MiqPassword.md5crypt(evm[:root_password]) %> 20 21 # Source Files Location 22 url --url=<%= repo_source_url %> 23 repo --name="<%= os_ver %>" --baseurl=<%= repo_source_url %> --cost=10 24 25 26 ### Post Install Scripts </pre>

5. Select the **Image Type** list. This list includes the PXE image types you created.
6. In **Type**, select **Kickstart** or **Cloudinit** for Linux based systems, and **Sysprep** for Windows based system.
7. In the **Script** area, either paste the script from another source or type the script directly into the Red Hat CloudForms interface.
8. Click **Add**.

The default dialogs show all possible parameters for provisioning. To limit the options shown, see [Section 2.9.6, “Customizing Provisioning Dialogs”](#).



2.9.5. Provisioning a Host

After setting up the IPMI and PXE environments, you are ready to provision a host. Currently, you can only provision in the cluster where the template is located or you can create a template in each cluster and let a Red Hat CloudForms Automate method automatically switch the selected template in the provision object.



IMPORTANT

A customization template with host-specific script additions is required. Ensure especially that the customization template contains the post-installation callback to enable discovery in Red Hat CloudForms.

1. Navigate to **Compute** → **Infrastructure** → **Hosts**.
2. Select a host with IPMI enabled.
3. Click  (**Lifecycle**), then  (**Provision Hosts**).
4. In **Request Information**, type in at least a **First Name** and **Last Name** and an email address. This email is used to send the requester status emails during the provisioning. The other information is optional. If the Red Hat CloudForms server is configured to use LDAP, you can use the **Look Up** button to populate the other fields based on the email address.

Request Information

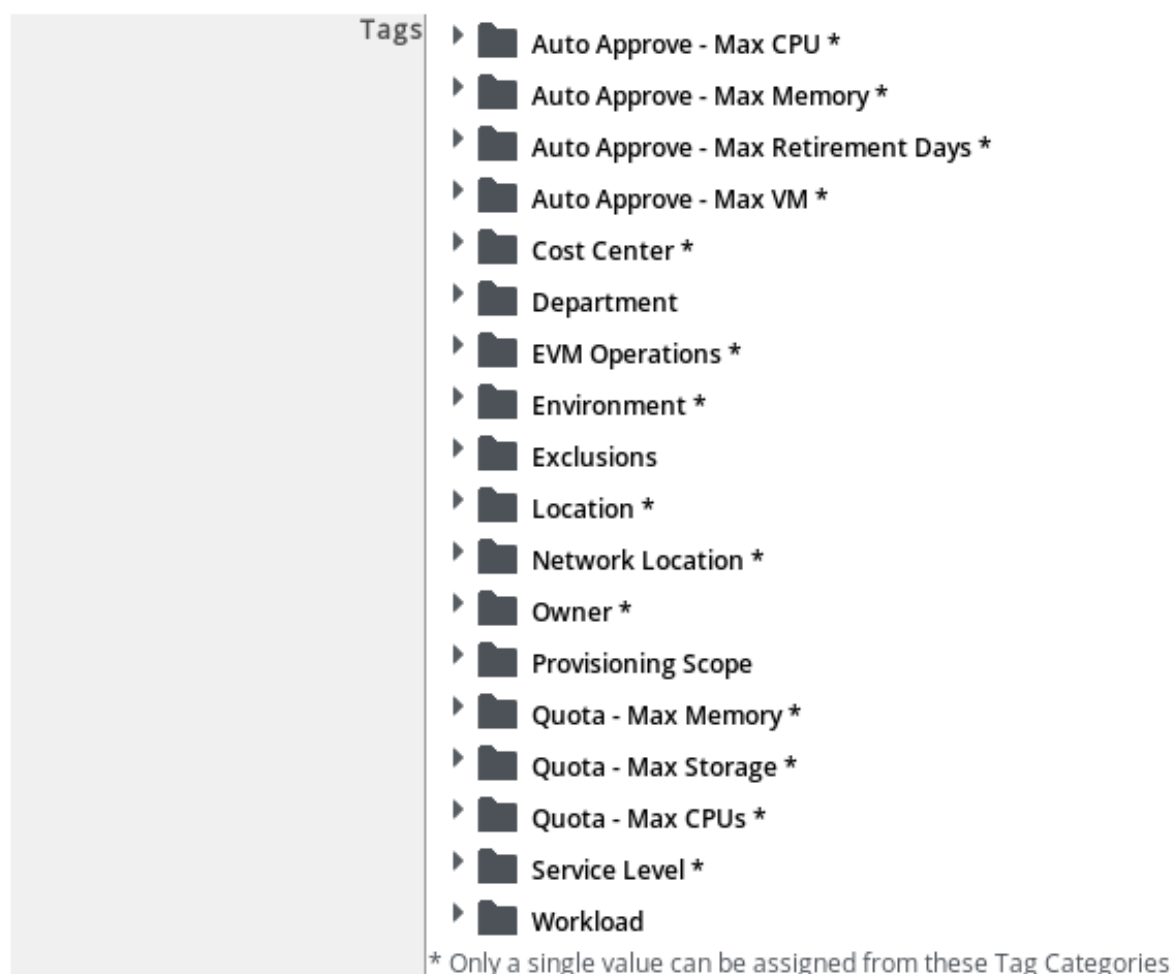
E-Mail *	<input type="text"/>
First Name *	<input type="text"/>
Last Name *	<input type="text"/>
Notes	<input type="text"/>

Manager

Name	<input type="text"/>
------	----------------------

5. On the **Purpose** tab, select the desired tags for the provisioned host.

Select Tags to apply



6. On the **Catalog** tab, select the hosts to provision.
- In the **Host** area, select the hosts to provision.
 - In the **PXE** area, select the PXE server and image.
7. On the **Customize** tab, you can select customizations for the operating system of the new host. These options vary based on the operating system to be provisioned.
- Use **Credentials** to type in a root password
 - In the **IP Address** area, select either **Static** or **DHCP** and enter any other address information you need. If needed, type in DNS specifications.
 - Under **Customize Template**, select a script.

- On the **Schedule** tab, select when to start the provisioning process.

When to Provision	<input checked="" type="radio"/> Schedule <input type="radio"/> Immediately on Approval
Provision on	<input type="text" value="10/23/2014"/>
Starting Time (UTC)	<input type="text" value="0"/> <input type="text" value="0"/>
Stateless	<input type="checkbox"/>

- In **Schedule Info**, select when to start the provisioning process. If you select **Schedule**, enter a date and time.
- Select **Stateless** to retain files after the provision completes. A stateless provision does not write to the disk so it requires the PXE files on the next boot.
- Click **Submit**.

The provisioning request is sent for approval. For the provisioning to begin, a user with the admin, approver, or super admin account role must approve the request. The admin and super admin roles can also edit, delete, and deny the requests. You will be able to see all provisioning requests where you are either the requester or the approver.

After submission, the appliance assigns each provision request a **Request ID**. If an error occurs during the approval or provisioning process, use this ID to locate the request in the appliance logs. The Request ID consists of the region associated with the request followed by the request number. As regions define a range of one trillion database IDs, this number can be several digits long.

Request ID Format

Request 99 in region 123 results in Request ID 123000000000099.

2.9.6. Customizing Provisioning Dialogs



The default set of provisioning dialogs shows all possible options. However, Red Hat CloudForms also provides the ability to customize which tabs and fields are shown. You can decide what fields are required to submit the provisioning request or set default values.

For each type of provisioning, there is a dialog that can be created to adjust what options are presented. While samples are provided containing all possible fields for provisioning, you can remove what fields are shown but cannot add new fields or tabs.

Edit the dialogs to:

- Hide or show provisioning tabs.
- Hide or show fields. If you hide an attribute, the default will be used, unless you specify otherwise.
- Set default values for a field.
- Specify if a field is required to submit the request.
- Create custom dialogs for specific users.

2.9.6.1. Adding a Provision Dialog for All Users

1. Login to the Red Hat CloudForms console for the Red Hat CloudForms server where you want to change the dialog.
2. Navigate to **Automate** → **Customization**.
3. Click the **Provisioning Dialogs** accordion.
4. Click the type of dialog you want to create: **Host Provision**, **VM Provision** or **VM Migrate**.
5. Select one of the default dialogs.
6. Click  (**Configuration**), and then  (**Copy this Dialog**).
7. Type a new **Name** and **Description** for the dialog.
8. In the **Content** field,
 - To remove a tab from display, change its display value to ignore. By choosing ignore, you not only hide the tab, but also skip any fields on that tab that were required. To show the tab, change the display value to show.
 - To hide a field, change its **:display:** value from **:edit** to **:hide**. To display fields of most data types, use **:edit**. To display a button, use **:show**. To set a default value for a field, add **:default: defaultvalue** to the list of parameters for the field. Set the **:required:** parameter to either **true** or **false** based on your needs.





NOTE

If you set **:required:** to **true**, the field must have a value for the provision request to be submitted.

9. Click **Add**.

If you are using **Provisioning Profiles**, you can specify a specific file that holds the customizations. To do this, you must create an instance mapping to this file in the Red Hat CloudForms **Applications/provisioning/profile/VM provisioning by group** class. By default, if you are using provisioning profiles and the group does not have a defined instance, the appropriate default dialog file will be used based on the type of provisioning selected.

2.9.6.2. Creating a Custom Provision Dialog

1. Navigate to **Automate** → **Customization**.
2. Click on the **Provisioning Dialogs** accordion.
3. Click on the type of dialog you want to create, **Host Provision**, **VM Provision** or **VM Migrate**.
4. Select one of the default dialogs.
5. Click  (**Configuration**), and then  (**Copy this Dialog**).
6. Rename the dialog as shown in the examples below.

Type of Provision	Dialog Name
Provision Virtual Machine from a template	miq_provision_dialogs_groupname_template Example: miq_provision_dialogs_EvmGroup-user_self_service_template
Clone a Virtual Machine	miq_provision_dialogs_groupname_clone_to_vm Example: miq_provision_dialogs_EvmGroup-user_self_service_clone_to_vm
Publish a Virtual Machine to a template	miq_provision_dialogs_groupname_clone_to_template Example: miq_provision_dialogs_EvmGroup-user_self_service_clone_to_template

7. Make any changes you need.

8. In the **Content** field,

- To remove a tab from display, change its display value to ignore. By choosing ignore, you not only hide the tab, but also skip any fields on that tab that were required. To show the tab, change the display value to show.
- To hide a field, change its **:display:** value from **:edit** to **:hide**. To ensure the field does not get turned back on by a workflow model, use **:display_override: :hide**. To display fields of most data types, use **:edit**. To display a button, use **:show**. To set a default value for a field, add **:default: defaultvalue** to the list of parameters for the field. Set the **:required:** parameter to either **true** or **false** based on your needs.



NOTE

If you set **:required:** to **true**, the field must have a value for the provision request to be submitted.

9. Click **Add**.

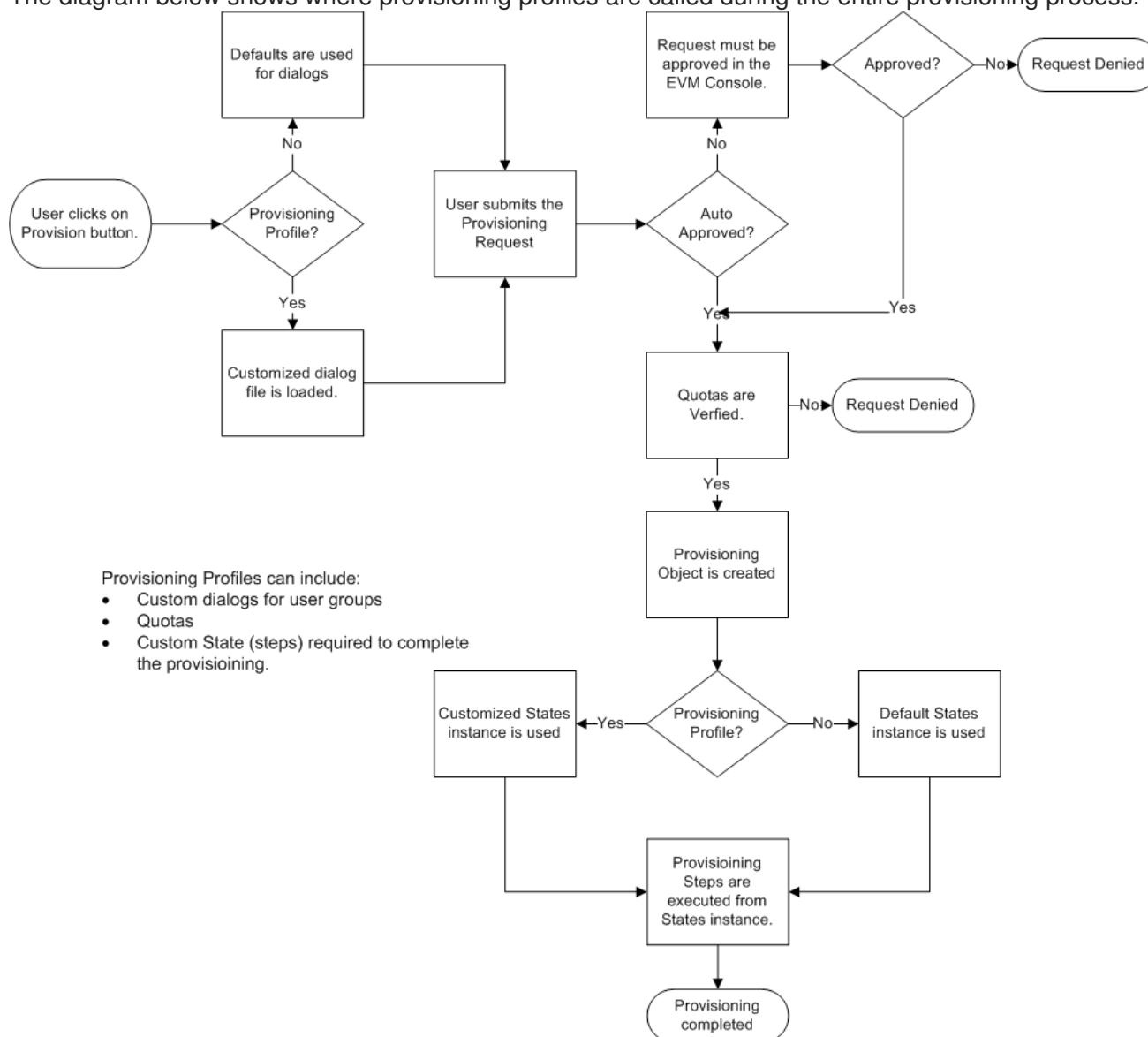
Enter the name of the new dialog into the dialog name field in the appropriate Red Hat CloudForms **Applications/provisioning/profile instance**. This dialog can now be referred to in an instance in the Provisioning Profiles class so that it can be used for groups of users.

2.9.7. Provisioning Profiles

Provisioning profiles can be used to customize the dialogs and the state machine (steps used to provision the machine). Profiles can be created for LDAP or Red Hat CloudForms groups. To use provisioning profiles:

- Create a **Provisioning Profile** instance for the LDAP or Red Hat CloudForms group. If no instance exists, then default settings will be used.
- If customizing dialogs, create a custom dialog file, and specify the name of that file in the provisioning profile instance. If customizing the states for provisioning, create a state instance and set the name of the state instance in the provisioning profile instance.

The diagram below shows where provisioning profiles are called during the entire provisioning process.



2.9.7.1. Creating a Provisioning Profile Instance



1. Navigate to **Automate** → **Explorer**.
2. Using the tree located in the accordion, click **DOMAIN** → **Cloud** → **VM** → **Provisioning** → **Profile**.



NOTE

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.









This example uses the **Cloud** Namespace, but can also use the **Infrastructure** namespace.

3. Click  (**Configuration**),  (**Add a New Instance**).
4. Make the name of the tag identical to the name of the LDAP or Red Hat CloudForms group you are creating the instance for, replacing spaces in the group name with underscores. For

example, change **Red Hat CloudForms-test group** to **Red Hat CloudForms-test_group**.

Name	CloudForms-test_group
Display Name	
Description	

Fields

Name	Value	On Entry	On Exit	On Error	Collect
 (vmname)	/Cloud/VM/Provisioning/Na	<input type="text"/>	<input type="text"/>	<input type="text"/>	/vmname = vmname
 (pre_dialog_name)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	/dialog_name = pre_dialog
 (get_dialog_name)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
 (vm_dialog_name_prefix)	vm_dialog_name_prefix	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
 (dialog_name)	miq_provision_dialogs-Clou	<input type="text"/>	<input type="text"/>	<input type="text"/>	/dialog_name = dialog_nam
 (auto_approval_state_machine)	ProvisionRequestApproval	<input type="text"/>	<input type="text"/>	<input type="text"/>	/state_machine = auto_app
 (quota_state_machine)	ProvisionRequestQuotaVer	<input type="text"/>	<input type="text"/>	<input type="text"/>	/state_machine = quota_st
 (state_machine)	VMProvision_\$(/miq_prov	<input type="text"/>	<input type="text"/>	<input type="text"/>	/state_machine = state_ma

- In the dialog name field, enter the name of the customized dialog file. This file must reside on the Red Hat CloudForms appliance in the `/var/www/miq/vmdb/db/fixtures` directory. Red Hat recommends naming the file in the format **miq_provision_dialogs-groupname.rb** and copying this file to all Red Hat CloudForms appliances. For instructions on creating a custom dialog file, see [Section 2.9.6, “Customizing Provisioning Dialogs”](#).



NOTE

Be sure that the custom dialog file exists. If it does not, an error will appear when the user clicks on the **Provisioning** button in the Red Hat CloudForms console.



- Click **Add**.

2.9.7.2. Setting Provisioning Scope Tags

Some non-default placement methods, for example the **redhat_best_placement_with_scope** or **vmware_best_fit_with_scope** methods, may require you to set **Provisioning Scope** tags for a host and a datastore.



To enable these resources for all groups, set the scope to **All**. To limit access to a select group, create a tag in the **Provisioning Scope** category with the exact name of the user group and set this tag on the desired resources. See [Tags](#) in *General Configuration* for information on creating tags.

To set the scope for a host:

- Navigate to **Compute** → **Infrastructure** → **Hosts**.
- Select the host to set the provisioning scope for.
- Click  (**Policy**), and then  (**Edit Tags**).
- From the **Select a customer tag to assign** drop down, select **Provisioning Scope** and then a value for the tag from the next drop down menu.
- Click **Save**.

To set the scope for a datastore:

- Navigate to **Compute** → **Infrastructure** → **Datastores**.
- Select the datastore to set the provisioning scope for.



3. Click  (**Policy**), and then  (**Edit Tags**).
4. From the **Select a customer tag to assign** drop down, select **Provisioning Scope** and then a value for the tag from the next drop down menu.
5. Click **Save**.

2.9.8. Managing Key Pairs

Key pairs allow you to manage SSH access between a user and provisioned instance. For more information about key pairs in OpenStack, see [Manage Key Pairs](#) in the *Instances and Images Guide*.

To manage key pairs, navigate to **Compute** → **Clouds** → **Key Pairs**. From there, you can view a list of available key pairs. Click on a key pair to view its details.

To create a new key pair:

1. Navigate to **Compute** → **Clouds** → **Key Pairs**.
2. Click  (**Configuration**),  (**Add a new Key Pair**).
3. Enter a **Name** for the key pair.
4. If you want to use a public key, copy its contents into the **Public Key (optional)** field.
5. Select which cloud provider on which to create the key pair. The key pair will then be available for use by instances in that provider.
6. Click **Add**.

CHAPTER 3. WORKING WITH REQUESTS

3.1. PROVISIONING REQUEST APPROVAL METHODS

In this chapter, you will learn about the different approval methods. The request can be approved manually in the Red Hat CloudForms console, set for automatic approval by setting options in the **Automate Explorer**, or by using an external method.



When using an external method, the approval actually takes place on the external system and is sent directly for execution. This chapter discusses how to view and edit requests in the Red Hat CloudForms Console, how to approve a request, and how to set automatic approval parameters.

3.2. WORKING WITH PROVISIONING REQUESTS


After a provisioning request is sent, if you have proper authority, you can copy, edit, delete, approve, or deny a request.

After submission, the appliance assigns each provision request a **Request ID**. If an error occurs during the approval or provisioning process, use this ID to locate the request in the appliance logs. The **Request ID** consists of the region associated with the request followed by the request number. As regions define a range of one trillion database IDs, this number can be several digits long.

Request ID Format


Request 99 in **region 123** results in **Request ID 123000000000099**.

3.2.1. Reloading the Status of Provisioning Requests

1. Navigate to **Services** → **Requests**.
2. Click  (**Reload the current display**).


3.2.2. Approving a Provisioning Request

After a user creates provisioning request, administrators have the ability to approve the request and allow Red Hat CloudForms to complete virtual machine or instance creation.


1. Navigate to **Services** → **Requests**.
2. Click on the request you want to approve.
3. Type in a **Reason** for the approval.
4. Click  (**Approve this request**).

3.2.3. Denying a Provisioning Request

1. Navigate to **Services** → **Requests**.


2. Click on the request you want to deny.
3. Type in a **Reason** for the denial.
4. Click  (**Deny this request**).

3.2.4. Copying a Provisioning Request


1. Navigate to **Services** → **Requests**.
2. Click on the request you want to copy.
3. Click  (**Copy original provision request**).
4. Make changes to the request.
5. Click **Submit**.

If the logged in user is not same as the requester or the request has been already approved or denied, you cannot edit or delete the request.

3.2.5. Editing a Provisioning Request

1. Navigate to **Services** → **Requests**.
2. Click on the request you want to edit.
3. Click  (**Edit the original provision request**).
4. Make changes to the request.
5. Click **Submit**.

3.2.6. Deleting a Provisioning Request

1. Navigate to **Services** → **Requests**.
2. Click on the request you want to delete.
3. Click  (**Delete this request**).
4. Click **OK** to confirm.

3.2.7. Automatically Approving Requests

You can set thresholds for automatic approval of provisioning requests and, therefore, remove the requirement to manually approve the request. You can do this either as a global default or on a per template basis.

3.2.7.1. Enabling Global Defaults for Automatic Approval

To enable a global set of default approval values, edit the defaults instance by navigating to **Automate** → **Explorer**, then **DOMAIN** → **Cloud|Infrastructure** → **VM** → **Provisioning** → **StateMachines** →

ProvisionRequestApproval in the accordion menu. The parameters in this instance are used by the methods in that same class. By default, the maximum number of virtual machines or instances that can be automatically approved for provisioning is 1. To skip the check for the maximum number of virtual machines, set this field to 0. Set this field to -1 to force manual approval. At a minimum, you must change this parameter for all others to be validated.



1. Navigate to **Automate** → **Explorer**.
2. From the tree in the accordion menu, select **DOMAIN** → **Cloud** → **VM** → **Provisioning** → **StateMachines** → **ProvisionRequestApproval Class**.



NOTE

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.

This example uses the **Cloud** Namespace but can also use the **Infrastructure** namespace.






3. Click  (**Configuration**), then  (**Edit this instance**).

Fields

Main Info

Fully Qualified Name	ManageIQ-Test / Cloud / VM / Provisioning / StateMachines / ProvisionRequestApproval / Default
Name	Default
Display Name	
Description	

Fields

Name	Value	On Entry	On Exit	On Error	Collect
 (max_cpus)					
 (max_vms)	1				
 (max_memory)					
 (max_retirement_days)					
 (ValidateRequest)		validate_request		pending_request	



NOTE

Do not change any values other than those listed below. Doing so may prevent the automatic approval process from running.

- Use **max_cpus** to set the number of CPUs allowed to approve automatically the provisioning request.
- Use **max_vms** to set the maximum number of virtual machines or instances that are allowed to be provisioned automatically approve the request. If this is set to blank, no requests will be automatically approved.
- Use **max_memory** to set the maximum memory allowed to approve automatically the provisioning request.
- Use **max_retirement_days** to set the maximum number of days until the virtual machine or instance is retired to automatically approve this request.

- If a value is blank or **0**, the parameter is ignored.

4. Click **Save**.

The thresholds for automatic approval are set. The next time a provision request is created these thresholds will be checked. If the requirements are met, the provisioning request will be approved with no user intervention.

3.2.7.2. Template Specific Approval Defaults

Red Hat CloudForms provides tags that can be used to set default automatic approval values on a per template or image basis. These values **supersede** those in the **Automate** model. Use these tags to eliminate the need for manual approval for all provisioning requests. To enable automatic approval, assign the tags directly to templates or images.

Category Display Name (Name)	Use (Sample values)
Auto Approve Max CPU (prov_max_cpus)	Sets the maximum number of CPUs that can be automatically approved in a single provisioning request. Sample Values: 1, 2, 3, 4, 5
Auto Approve Max Memory (prov_max_memory)	Sets the maximum number of memory that can be automatically approved in a single provisioning request. Sample Values: 1, 2, 4, 8 (in GB)
Auto Approve Max Retirement Days (prov_max_retirement_days)	Sets the maximum number of days until retirement that can be automatically approved in a single provisioning request. Sample Values: 30, 60, 90, 180 (in days)
Auto Approve Max VM (prov_max_vms)	Sets the maximum number of virtual machines or instances that can be automatically approved in a single provisioning request. Sample Values: 1, 2, 3, 4, 5

3.2.7.3. Assigning Tags to a Template for Auto Approval

1. Navigate to **Compute** → **Infrastructure** → **Virtual Machines**.
2. Click the **Templates** accordion, and select the templates that you want to tag.
3. Click  (**Policy**), and then  (**Edit Tags**).
4. Select a customer tag from the first dropdown, and then a value for the tag.

The thresholds for automatic approval for a specific template are set. The next time a provision request is created for this template these thresholds will be checked. If the requirements are met, the provisioning request will be approved with no user intervention.

3.2.7.4. Setting Provisioning Notification Email Addresses

Red Hat CloudForms contains a set of Automate instances for provisioning. These Automate instances also include email fields to set the sender and recipient of provisioning notifications, such as requests. These fields are set to **evmadmin@company.com** as a default.



1. Navigate to **Automate** → **Explorer**.
2. Choose the following Namespace: **DOMAIN** → **Cloud** → **VM** → **Provisioning** → **Email**.



NOTE

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.


This example uses the **Cloud** Namespace, but can also use the **Infrastructure** namespace.

3. Select an instance within the chosen class.
4. Click  (**Configuration**), then  (**Edit this instance**).
5. Type the desired email addresses in the **to_email_address** and **from_email_address** fields.
6. Click **Save**.

Main Info

Fully Qualified Name	ManageIQ-Test / Cloud / VM / Provisioning / Email / MiqProvisionRequest_Approved
Name	MiqProvisionRequest_Approved
Display Name	
Description	

Fields

Name	Value	On Entry	On Exit	On Error	Collect
 (to_email_address)	evmadmin@company.c				
 (from_email_address)	evmadmin@company.c				
 (signature)	Virtualization Infrastru				
 (vm_retire_extend_days)					
 (method1)	MiqProvisionRequest_A				

CHAPTER 4. FULFILLING REQUESTS

4.1. FULFILLING A REQUEST

After a request has been approved, Red Hat CloudForms then goes through the steps required to complete the request. The steps followed for a regular provision from a virtual machine to a virtual machine (not to a template) are found by navigating to **Automate** → **Explorer**, then listed under **DOMAIN** → **Cloud|Infrastructure** → **VM** → **Provisioning** → **VMProvision_VM** → **Provision VM from Template (template)**. The value for each state shows where the instance resides in the **Datastore** accordion. The default set of execution steps is shown below. For more information on state machines, see [Section 4.2.1.4, “State Machines”](#).

4.2. DEFAULT EXECUTION STEPS IN STATES INSTANCE

Step	Description
Customize Request	Apply customizations.
Acquire IP Address	Integrates with IPAM (IP Address Management) to get an IP Address.
Acquire MAC Address	Integrates with IPAM to get a MAC Address.
Register DNS	Integrates with IPAM to register with DNS.
Register CMDB	Integrates with CMDB (Configuration Management Database) to register with the CMDB.
Register AD	Integrates with IPAM to register with active directory.
PreProvision	Pre-provisioning steps.
Provision	Create the virtual machine or instance.
CheckProvisioned	Check that the new virtual machine or instance is in the VMDB.
PostProvision	Post-provisioning steps.
Register DHCP	Integrate with IPAM to register the IP address with DHCP Server.
Activate CMDB	Integrate with IPAM to activate the virtual machine or instance in the CMDB.
Email owner	Send email to owner that the virtual machine or instance has been provisioned.

4.2.1. Quotas

Quotas allow you to establish maximum usage thresholds for an user, group, or tenant for provisioned virtual machines or instances and are integrated into provisioning profiles. These maximums are

checked after the approval but before the actual provision request is started. The quota is set for the tenant or group as a whole.


4.2.1.1. Applying User or Group Quotas

1. Log in as a user with administrator or super administrator rights to the Red Hat CloudForms console.
2. Navigate to **Automate** → **Explorer**.
3. Copy the **ManageIQ** → **System** → **CommonMethods** → **QuotaStateMachine** → **quota** instance to a custom DOMAIN.
4. From the accordion menu, click **DOMAIN** → **System** → **CommonMethods** → **QuotaStateMachine** → **quota**.



NOTE

By default, quotas are applied to tenants and do not require any change in **Automate** → **Explorer**.

5. Click  (**Configuration**),  (**Edit this instance**).
Editing Automate Instance "quota"

Fields

Main Info







Fully Qualified Name / AcmeQuota / System / CommonMethods / QuotaStateMachine / quota

Name quota

Display Name

Description

Fields

Name	Value	On Entry	On Exit	On Error	Collect
 Quota Source Type (quota_source_type)	group				
 VM Warning Count (warn_vms)	10				
 VM Maximum Count (max_vms)	20				
 Storage Warning Limit (warn_storage)	1024				
 Storage Maximum Limit (max_storage)	2048				
 CPU Warning Count (warn_cpu)	1				

- a. Set the value for **Quota Source Type** to *user* or *group*.



IMPORTANT

A user creating a provisioning request must have an email address saved in their profile, or provisioning may fail. See [Creating a User](#) in *General Configuration* for details on configuring users.

- b. Set the values for **VM Warning Count**, **VM Maximum Count**, **Storage Warning Limit**, **Storage Maximum Limit**, **CPU Warning Count**, **CPU Maximum Count**, **Memory Warning Limit**, or **Memory Maximum Limit** to be the maximums for a specific user or group.
6. Click **Save**.



4.2.1.2. Using Tags for Owner and Group Quotas

Red Hat CloudForms provides tags for enforcing quotas for the owners of virtual machines or instances. Ownership of a virtual machine or instance can be set either during the provisioning process or by using the **Configuration Set Ownership** button. If a virtual machine or instance has an owner, the value is displayed in the **Lifecycle** section of the virtual machine or instance summary page.

Quota tags can be assigned directly to **either** a group or owner **not** to a configuration item. The table below shows the tags for use in quotas.

Category Display Name (Name)	Use
Quota Max CPU (quota_max_cpu)	Sets the maximum number of CPUs summed over all virtual machines and instances owned by the group or user. Sample Values: 1, 2, 3, 4, 5, 10, 20, 30, 40, 50
Quota Max Memory (quota_max_memory)	Sets the maximum memory summed over all virtual machines and instances owned by the group or user. Sample Values: 1024, 2048, 4096, 8192, 10240, 20480, 40960, 81920 (in MB)
Quota Max Storage (quota_max_storage)	Sets the maximum storage summed over virtual machines and instances owned by the group or user. Sample Values: 10, 100, 1000, 20, 200, 40, 400 (in GB)

4.2.1.3. Applying a Tag to a User or User Group

1. From the settings menu, select **Configuration**.
2. Click the **Access Control** accordion, and select the user or group that you want to tag.
3. Click  (**Policy**), then click  (**Edit Tags**).
4. Select the appropriate customer tag to assign, then the value.
5. Click **Save**.



NOTE

When quotas are applied by both automate instance and tagging, the tagged values will have higher precedence.

4.2.1.4. State Machines

The automate state machine processes an ordered list of states. It can ensure the successful completion of a step before the next step is run, permit steps to be retried, allow setting a maximum time to retry the state before exiting, and number of retries before exiting the state. Before each state is executed, the **On_Entry** method is executed and after the state ends the **On_Exit** or **On_Error** method is executed based on how the state ends.

The following components make up a Red Hat CloudForms automate state machine:

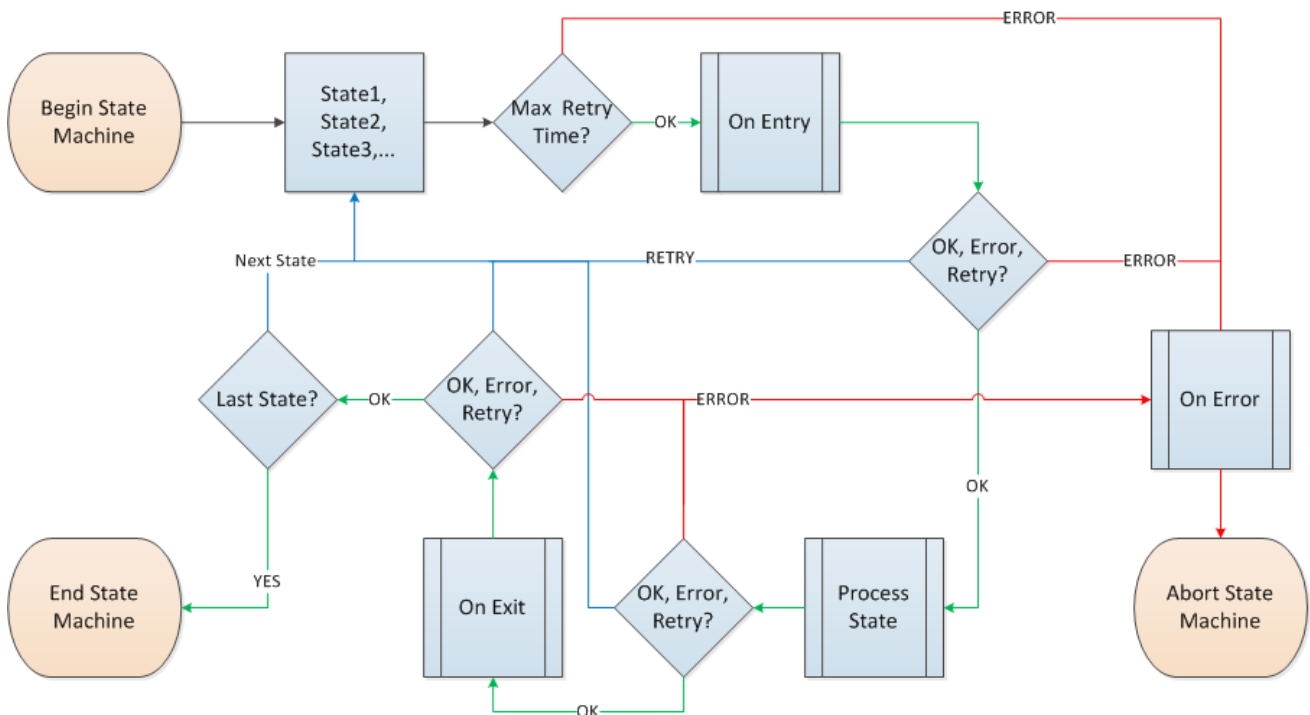
Component	Description
On_Entry	Method to run when entering the state. It enables you to execute an automate method to do some pre-processing before the state of the state machine is processed.
On_Exit	Method to run when exiting the state.
On_Error	Method to run if an error is encountered when running the state. It enables you to execute an automate method to do some final processing before the state machine finally exits (MIQ_ABORT) due to the error.
Default Value	Runs after the On_Entry method completes (The actual state being processed).
Max Retries	Maximum number of times to retry the state before exiting.
Max Time	Maximum time in seconds to retry the state before exiting.

In the diagram below, you can see how these components combine to create a state machine workflow.



NOTE

The retry logic, **On_Entry** and **On_Error** are distinct cases in the program flow.



Code snippet demonstrating the state machine retry logic:

```
# Get current provisioning status
task = $evm.root['service_template_provision_task']
task_status = task['status']
```

```
result = task.status
```




Then check the result to see how it should proceed:

```
case result
when 'error'
  $evm.root['ae_result'] = 'error'
  .....
when 'retry'
  $evm.root['ae_result'] = 'retry'
  $evm.root['ae_retry_interval'] = '1.minute'
when 'ok'
  $evm.root['ae_result'] = 'ok'
end
```

When the result is "retry", it sets:

```
$evm.root['ae_result'] = 'retry'
$evm.root['ae_retry_interval'] = '1.minute'
```

The following image shows a simple state machine pertaining to approving a provision request. This instance can be found in **Datastore** → **ManageIQ** → **Infrastructure** → **VM** → **Provisioning** → **StateMachines** → **ProvisioningRequestApproval** → **Default**.

Name	Value	On Entry	On Exit	On Error	Collect	Max Retries	Max Time	Message
 max_vms	1							create
 ValidateRequest		validate_request		pending_request		100		create
 ApproveRequest		approve_request				100		create

1. The attribute **max_vms** has a value of 1. State machine processing can use the attributes of the state machine instance to make logic decisions. In this case, the **validate_request** method, which is processed during the **On_Entry** portion of the **ValidateRequest** state, evaluates the **max_vms** attribute. If the number of virtual machines requested is less than the **max_vms** value, the request can be auto-approved.
2. **ValidateRequest** is the first state to be executed.
3. **ApproveRequest** is the next state to be executed.



NOTE

Grayed out items reflect values that are set in the class schema. These values can be overwritten on a per instance basis.

4.2.1.5. Customizing Provisioning States







The steps followed when provisioning a virtual machine or cloud instance are completed based on instances from the **DOMAIN** → **Cloud|Infrastructure** → **VM** → **Provisioning** → **StateMachines** → **VMProvision_VM** class. Depending on your environment you can remove, change, or add steps to the

provisioning process. For example, if you are not integrating with IPAM or a CMDB, then you can remove those execution steps.

Main Info

Fully Qualified Name	ManageIQ-Test / Cloud / VM / Provisioning / StateMachines / VMProvision_VM / template
Name	template
Display Name	Provision VM from Template
Description	

Fields

Name	Value	On Entry	On Exit	On Error	Collect
 (CustomizeRequest)	/Cloud/VM/Provisioning	update_provision_statu	update_provision_statu	update_provision_statu	
 (AcquireIPAddress)		update_provision_statu	update_provision_statu	update_provision_statu	
 (AcquireMACAddress)		update_provision_statu	update_provision_statu	update_provision_statu	
 (RegisterDNS)		update_provision_statu	update_provision_statu	update_provision_statu	
 (RegisterCMDB)		update_provision_statu	update_provision_statu	update_provision_statu	
 (RegisterAD)		update_provision_statu	update_provision_statu	update_provision_statu	

4.2.1.6. Editing the Default State Instance



1. Navigate to **Automate** → **Explorer**.
2. From the accordion menu, click **DOMAIN** → **Cloud** → **VM** → **Provisioning** → **StateMachines** → **VMProvision_VM**.



NOTE

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.







This example uses the **Cloud** Namespace, but can also use the **Infrastructure** namespace.

3. Click  (**Configuration**), then  (**Edit this instance**).

Main Info

Fully Qualified Name	ManageIQ-Test / Cloud / VM / Provisioning / StateMachines / VMProvision_VM / template
Name	template
Display Name	Provision VM from Template
Description	

Fields

Name	Value	On Entry	On Exit	On Error	Collect
 (CustomizeRequest)	/Cloud/VM/Provisioning	update_provision_statu	update_provision_statu	update_provision_statu	
 (AcquireIPAddress)		update_provision_statu	update_provision_statu	update_provision_statu	
 (AcquireMACAddress)		update_provision_statu	update_provision_statu	update_provision_statu	
 (RegisterDNS)		update_provision_statu	update_provision_statu	update_provision_statu	
 (RegisterCMDB)		update_provision_statu	update_provision_statu	update_provision_statu	
 (RegisterAD)		update_provision_statu	update_provision_statu	update_provision_statu	

4. For each step that you want to remove, clear the entries in the **Value**, **On Entry**, **On Exit**, and **On Error** columns.

5. Click **Save**.

4.2.1.7. Viewing the Status of a Provisioning Request




After a request has been approved, the various stages of fulfillment are executed. You can see the progress of the provisioning process by viewing its status.

1. Navigate to **Services** → **Requests**. The list of requests is shown.
2. Click on a specific request for more information. Once the provisioning begins, if the request was supposed to create more than one virtual machine or instance, a field will appear called **Provisioned VMs**. Click on the number that appears next to it for information on each of the individual provisions.

4.2.1.8. Viewing a Provisioned Virtual Machine or Instance

When a virtual machine or instance is created as a result of a provisioning request, its summary screen will show when it was provisioned in the **Lifecycle** area of the respective summary.

1. From **Services** → **Workloads**, click the virtual machine or instance that you want to view.

Discovered	 Mon Mar 01 19:52:21 UTC 2010
Last Analyzed	 Never
Retirement Date	 Never
Provisioned On	Mon Mar 01 19:53:03 UTC 2010
Owner	

4.2.1.9. Viewing a Virtual Machine or Instance Summary

From **Services** → **Workloads**, click the virtual machine or instance that you want to view.

CHAPTER 5. CATALOGS AND SERVICES

Through the use of catalogs, Red Hat CloudForms provides support for multi-tier service provisioning to deploy layered workloads across hybrid environments. You can create customized dialogs that will give consumers of the services the ability to input just a few parameters and provision the entire service. The following table lists the terminology associated with catalogs that you will use within the CloudForms user interface for service provisioning.

Table 5.1. Terminology

Type	Information
Catalog Bundle	A group of templates.
Catalog Item	A single template.
Template	A template is a copy of a preconfigured virtual machine, designed to capture the installed software and software configurations, as well as the hardware configuration of the original virtual machine.
Dialog Tabs	Part of a service dialog.
Element	An item on a tab in a dialog. It can be a button, check box, drop down list, radio button, tag control, text area box, or a text box.
Provisioning Dialogs	Dialogs created for host provisioning, virtual machine migration, or virtual machine provisioning. The dialog name must be added to the appropriate provision instance to be processed.
Service Catalog	A catalog item or catalog bundle that is available for provisioning.
Service Dialogs	Made up of fully customizable tabs, items, and values for use with service provisioning.

5.1. GENERIC OBJECTS

Generic Objects are object-like entities, defined at runtime, that have unique names and user-defined attributes and relationships. Residing in the Automate Engine datastore, generic objects are designed to manage objects other than those related to private infrastructure, and public or private cloud providers.

Using automate requests, services and catalog items, generic objects can be directly accessed or passed, during any step, as a parameter to a service state machine. As a result, generic objects can be used to quickly add the capability to provision and collect data on resources not supported by Red Hat CloudForms.



NOTE

Outside of the Red Hat CloudForms user interface, Red Hat Marketing may refer to **Dynamic Resource Objects** when referencing the features and applications of generic objects.

5.1.1. Viewing Generic Objects Classes






View a list of generic objects and click through to see detailed summary information for each object.

1. Navigate to **Automate** → **Generic Objects**.
2. Click on a generic object class in the table to view its summary information.

5.1.2. Creating Generic Objects Classes



Model a new resource by creating a generic object class and adding it to your Red Hat CloudForms inventory. Each generic object class can have attributes, associations, and methods. Once created, generic object classes are visible to users of the Self Service user interface at the resource level.

Create a generic object class using the following steps:

1. Navigate to **Automation** → **Automate** → **Generic Objects**.
2. Click  (**Configuration**), then click  (**Add a New Generic Object Class**).
3. Provide a **Name** and **Description** for the new object class.
4. In the **Attributes** field, enter a **Name** and choose a **Type** from the drop-down list. Click the  button to add attributes.
5. Enter a **Name** and select a **Class** for the object class's **Associations**. Click the  button to create additional associations.
6. Provide a **Name** for the **Methods**. Click the  button to add methods.
7. Click **Add**.

5.1.3. Editing Generic Object Classes

Edit existing generic object classes using the following steps:

1. Navigate to **Automation** → **Automate** → **Generic Objects**.
2. Click on a generic object class in the list view.
3. Click  (**Configuration**), then click  (**Edit this Generic Object Class**).
4. Make required changes to the generic object class fields.
5. Click **Save**.

5.1.4. Removing Generic Objects Classes

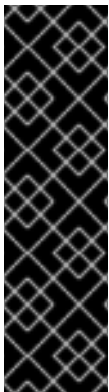
Remove generic object classes from your inventory using the following steps:

1. Navigate to **Automation** → **Automate** → **Generic Objects**.
2. Check the generic objects classes from the table to remove.
3. Click  (**Configuration**), then click  (**Remove selected Generic Object Classes from Inventory**).
4. Click **OK** to confirm.

5.2. SERVICE DIALOGS







When provisioning a service, input will be needed from the requester. Service dialogs are used to take input from the user. This input is connected to a method in the Automate model that defines how the users input is translated into the provision request. Before creating a service dialog, be sure to plan what items you need the user to input.

5.2.1. Adding a Service Dialog



IMPORTANT

- When creating a service dialog for use with Ansible playbook catalog items, variable elements must use the prefix **param_** when assigning the value. For example, a new variable labeled **key1** should have its value set as **param_key1**.
- Using Ansible playbooks to populate dynamic dialog fields is not recommended due to delay times caused by the overhead of interaction between systems.
- If you add the playbook automate method to a service dialog, only users with admin privileges can run the dialog.

1. Navigate to **Automation** → **Automate** → **Customization**.
2. Click the **Service Dialogs** accordion.
3. Click  (**Configuration**), and then  (**Add a new Dialog**).
4. In **Dialog Information**, enter a **Label** and **Description**. Check the boxes for the buttons you want available at the bottom of the dialog form. The description will appear as hover text. As you enter the **Label** of the dialog, it should appear in the **Dialog** pane on the left.
 - a. Click  (**Add**), then  (**Add a New Tab to this Dialog**).
 - b. Enter a **Label** and **Description** for this tab.
As you enter the **Label** of the tab, it should appear in the **Dialog** pane on the left under the dialog you are creating.
 - c. Click  (**Add**), then  (**Add a New Box to this Tab**).

- d. Enter a **Label** and **Description** for this box.

As you enter the **Label** of the box, it should appear in the **Dialog** pane on the left under the tab you are creating.

5. Add an element to this box. Elements are controls that accept input.

- a. Click  (**Add**), then  (**Add a New Element to this Box**).

- b. Enter a **Label**, **Name**, and **Description** for this element.



IMPORTANT

Name must use only alphanumeric characters and underscores without spaces. It is also used to retrieve the value of this element in the method used with the dialog and must start with **dialog_service_type**

- c. Select a **Type** for an element type. All **Type** options have a **Required** and **Default Value** field. Check **Required** or set **Required** to **true** if the element is required to proceed. You can also specify a default value. The rest of the options presented are based on which type of element you select.

Element Types	Additional Info
Check Box	Check Default Value if you want this check box checked by default.
Date Control	Use Date Control to create a field where users can select a date. If you want users to be able to select a date and time, use the Date/Time Control option.
Date/Time Control	Use Date/Time Control to create a field where users can select a date and time. Only one Date Control or Date/Time Control element can be present in a dialog.
Drop Down Dynamic List	Use Drop Down Dynamic List if you want the list options to be created using automate methods. Use Entry Point (NS/Cls/Inst) to select an automate instance. Check Show Refresh Button to allow users to refresh the list options manually.
Radio Button	This element type serves the same purpose as Drop Down List but displays options using radio buttons.
Tag Control	Select a Category of tags you want assigned to the virtual machines associated with this service dialog. Check Single Select if only one tag can be selected.
Text Area Box	Provides text area for users to enter some text. You can also leave a message to users by typing in the Default Value field or leave it as blank.

Element Types	Additional Info
Text Box	This element type serves the same purpose as Text Area Box with the option to check Protected so the text is shown as asterisks (*), instead of plain text.

- Continue adding the dialog items you need. You can switch between dialogs, tabs, boxes, and elements by selecting their respective labels from the **Dialog** pane on the left.
- Click **Add**. Your dialog should appear in the **Service Dialogs** accordion.

5.2.2. Adding a Service Dialog Using Dialog Editor

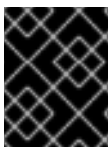
Red Hat CloudForms includes a drag-and-drop service dialog editor to create service dialogs. The editor, with its drag-and-drop feature, provides a visual representation of the components that comprise a service dialog. You can easily design your service dialog utilizing dialog tabs, sections (previously referred to as boxes), and elements.

When users access a service, the majority of options available to them are preset and cannot be altered. The requirements for the service determine the options and fields that need to be present in the dialog for user input. A service dialog exposes some of those options to the user so that even if they are ordering a basic Red Hat Enterprise Linux 7 machine, for example, they can at least choose the amount of memory, virtual CPUs, or other options available to the instance they order. In cases where certain fields must be unique, such as the name of virtual machines in Red Hat Virtualization, users must enter their own unique name for the virtual machine they choose or the operation will fail, so this field must be exposed.

See [Creating a Service in CloudForms for Virtual Machine Provisioning](#) for an example about creating a service that utilizes a service dialog for provisioning a Red Hat Virtualization virtual machine.



A service dialog contains three components:

- One or more **Tabs**.
- Inside the tabs, one or more **Sections**. Note that in the previous method of creating service dialogs using the CloudForms user interface, **Sections** were referred to as **Boxes**.
- Inside the sections, one or more **Elements**. Elements are controls that accept input. Elements contain methods, like check boxes, drop-down lists or text fields, to fill in the options on the provisioning dialog.



IMPORTANT

The names of the elements must correspond to the options used in the provisioning dialog.

- Navigate to **Automation** → **Automate** → **Customization**.
- Click the **Service Dialogs** accordion.
- Click  (**Configuration**), and then  (**Add a new Dialog**).

Automate Customization

General

Dialog's name

Dialog's description

Text Box

Text Area

Check Box

Dropdown

Radio Button

Datepicker

Timepicker

Tag Control

New tab

Create Tab

New section

Text Box



Check Box

Dropdown

Add Section



Cancel

Save

4. Enter basic details under **General**:
- a. Enter the **Dialog's name** and **Dialog's description**.
5. Add a new tab to the dialog:
- a. Click  **Create Tab**. Then, click the  icon on the new tab to edit tab information.


b. Enter a **Label**.

c. Optional: Enter a description for the tab in **Description**.

d. Click **Save**.
6. Add a new section to the tab:
- a. Click  **Add Section**. Then, click the  icon on the upper-right to edit section details.

b. Enter a **Label**.

c. Optional: Enter a description for the section in **Description**.

d. Click **Save**.
7. Add elements to the section:
- a. From the list of elements on the left, click an element you want to add, then drag-and-drop it inside the section. Then, click the  icon next to the element to edit its field details.

Element Types	Additional Info
Text Area	Provides text area for users to enter text. You can enter the default text in Default Value , or leave it as blank.
Text Box	Text box is similar to a text area with the added option to enable Protected so the text is shown as asterisks(*), instead of plain text.

Element Types	Additional Info
Check Box	Enable Default Value if you want the box checked by default.
Drop Down	Use drop down to create list entries either manually or using automate methods. Enable Dynamic to create lists using automate methods; use Entry Point to select an automate instance. Enable Show Refresh Button to allow users to refresh list options manually.
Radio Button	Similar to a drop down but displays options using radio buttons.
Datepicker	Use this to enable users to pick a date by clicking the calendar icon.
Timepicker	use this to enable users to pick a date and time.
Tag Control	Select a Category of tags you want assigned to virtual machines associated with the service dialog. Enable Single Select if only one tag can be selected.

- b. Enter a **Label**, **Name**, and **Description** for the element.



IMPORTANT



Element names must correspond to the options used in the provisioning dialog. **Name** must use only alphanumeric characters and underscores without spaces. It is also used to retrieve the value of this element in the method used with the dialog and must start with **dialog_service_type**.

- c. Optional: Add additional information in **Help** to assist the user to complete the fields in the service dialog. This field is useful for explaining unfamiliar terminology or providing configuration tips.
- d. Set other options as required.
- e. Click **Save**.
8. Optional: Repeat the above step to add more elements to the existing section, or create and add elements to a new section as required.
9. Optional: Repeat the step to add a new tab to the dialog, and subsequent steps to add sections and elements to it as required.
10. Click **Save** to create the dialog.

The service dialog is now created, and added to the **Service Dialogs** accordion.

5.2.3. Creating a Service Dialog from a Container Template

Complete the following procedure to create a Service Dialog from a Container Template.

1. Navigate to **Compute** → **Containers** → **Container Templates** and select the template for provisioning.
2. Click  (**Configuration**), then  (**Create Service Dialog from Container Template**).
3. Enter a name for the dialog in **Service Dialog Name**.
4. Click **Save**.

You can use this service dialog when creating a catalog item for container template provisioning; see [Section 5.4.8, “Creating an OpenShift Template Catalog Item”](#).

5.2.4. Importing Service Dialogs

You can share service dialogs between appliances using the export and import features.

1. Navigate to **Automate** → **Customization**.
2. In the **Import/Export** accordion, click **Service Dialog Import/Export**.
3. In the **Import** area, click **Browse** to select an import file.
4. Click **Upload**.

5.2.5. Exporting Service Dialogs

You can share service dialogs between appliances using the export and import features.

1. Navigate to **Automate** → **Customization**.
2. In the **Import/Export** accordion, click **Service Dialog Import/Export**.
3. In the **Export** area, select the service dialogs that you want to export.
4. Click **Export**.

5.3. METHODS

5.3.1. Creating a Method to Associate with the Dialog

You will need to create a method that connects the values in the dialog with the provisioning request. The method should be created in the **DOMAIN/Service/Provisioning/StateMachines/ServiceProvision_Template** class of the **Automate** model.



NOTE

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.

A method is provided below that was created for the following scenario:

- You want to provision a three-tiered service that contains catalog items of web, app and DB. Each of these virtual machines (or cloud instances) has been tagged under the **Service** category with the appropriate value. Then, added as a catalog item and combined into a catalog bundle.
- The **Service Dialog** captures the selection of small, medium or large application in a dropdown called **service_type**. When referring to a value captured in an element in a dialog, the name of the element should be prefixed with **dialog_**. For example, **service_type** becomes **dialog_service_type** when used in the method.
- The method will set the memory sizes for each of the catalog items based on the **service_type** selection.

```
#           Automate Method
#
$evm.log("info", "Automate Method ConfigureChildDialog Started")
#
#           Method Code Goes here
#
$evm.log("info", "=====")
$evm.log("info", "Listing ROOT Attributes:")
$evm.root.attributes.sort.each { |k, v| $evm.log("info", "\t#{k}: #{v}")}
$evm.log("info", "=====")

stp_task = $evm.root["service_template_provision_task"]
$evm.log("info", "=====")
$evm.log("info", "Listing task Attributes:")
stp_task.attributes.sort.each { |k, v| $evm.log("info", "\t#{k}: #{v}")}
$evm.log("info", "=====")

#####
#### This is how the method would look for dialog variables
#####
dialog_service_type = $evm.root['dialog_service_type']
$evm.log("info", "User selected Dialog option = [#{dialog_service_type}]")

stp_miq_request_task = stp_task.miq_request_task
#$evm.log("info", "(parent) miq_request_task: = [#
{stp_miq_request_task}]")

#####
#### This is how you get the catalog items for the catalog bundle
#####

stp_miq_request_tasks = stp_task.miq_request_tasks
#$evm.log("info", "(children) miq_request_tasks count: = [#
{stp_miq_request_tasks.count}]")

#####
#### By going through the children, you can set the dialog variable for
each of the children (we based our values on the childrens service tags)
#####

stp_miq_request_tasks.each do |t|

  $evm.log("info", " Setting dialog for: #{t.description}")
  service = t.source
```

```

    service_resource = t.service_resource
    # $evm.log("info", "Child service resource name: #
{service_resource.resource_name}")
    # $evm.log("info", "Child service resource description: #
{service_resource.resource_description}")

    service_tag_array = service.tags(:app_tier)
    service_tag = service_tag_array.first.to_s

    memory_size = nil

#####
#### The dialog_service_type is the attribute set on the service dialog
#### We use the service_tag to decide what child gets what dialog
#####

    case dialog_service_type
    when "Small"
      case service_tag
      when "app"
        memory_size = 1024
      when "web"
        memory_size = 1024
      when "db"
        memory_size = 4096
      else
        $evm.log("info", "Unknown Dialog type")
      end
    when "Large"
      case service_tag
      when "app"
        memory_size = 4096
      when "web"
        memory_size = 4096
      when "db"
        memory_size = 8192
      else
        $evm.log("info", "Unknown Dialog type")
      end
    else
      $evm.log("info", "Unknown Dialog type - setting Dialog options here")
    end

#####
#### set_dialog_option sets the dialog for the child
#####

    t.set_dialog_option('memory', memory_size) unless memory_size.nil?
    $evm.log("info", "Set dialog for selection: [{dialog_service_type}]
Service_Tier: [{service_tag}] Memory size: [{memory_size}]")

  end
#
#

```



```
#
$evm.log("info", "Automate Method ConfigureChildDialog Ended")
exit MIQ_OK
```

5.3.2. Creating a Method in the Service Class

Service methods have been split based on purpose.

1. Navigate to **Automate** → **Explorer**.
2. Service Class is located at **DOMAIN** → **Service** → **Provisioning** → **StateMachines** → **Methods** and **Domain** → **Service** → **Retirement** → **StateMachines** → **Methods**.



NOTE

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.

3. Click the **Methods** tab.
4. Click (**Configuration**), then (**Add a New Method**).
5. Enter a **Name** and **Display Name**.
6. In the **Data** field, enter the method contents.
7. Click **Validate** and wait for your data entry to be successfully validated.
8. Click **Add**.

Main Info

Fully Qualified Name	ManageIQ / Service / Provisioning / StateMachines / Methods
Name	<input type="text"/>
Display Name	<input type="text"/>
Location	inline ▼

Data

```
1 #
2 # Description: <Method description here>
3 #
4 #
```



Input Parameters

5.3.3. Creating an Instance in the Service Class

1. Navigate to **Automate** → **Explorer**.
2. Service Class is located at **DOMAIN** → **Service** → **Provisioning** → **StateMachines** → **Methods** and **Domain** → **Service** → **Retirement** → **StateMachines** → **Methods**.

**NOTE**

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.


3. Click the **Instances** tab.
4. Click  (**Configuration**), then  (**Add a new Instance**).
5. Enter a **Name** and **Display Name**.
6. In the **Fields** area, enter the method's name in **Value**.
7. Click **Add**.

The instance is created so that it can be called from the **ServiceProvision** class.

Main Info

Fully Qualified Name	ManageIQ-Test / Service / Provisioning / StateMachines / Methods
Name	<input type="text"/>
Display Name	<input type="text"/>
Description	<input type="text"/>

Fields

Name	Value	On Entry	On Exit	On Error	Collect
 (execute)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Add

Cancel

**NOTE**

After the method has been created, it must be mapped to an instance in the **DOMAIN/Service/Service/Provisioning/StateMachines** class. The name of the instance must be specified as the **Entry Point**. This method must be called before the provision job begins.



5.3.4. Associating a Method with an Automate Instance

Service methods have been split based on purpose.

1. Navigate to **Automate** → **Explorer**.
2. From the accordion menu, click the required service method.
3. Service Class is located at **DOMAIN** → **Service** → **Provisioning** → **StateMachines** → **Methods** and **Domain** → **Service** → **Retirement** → **StateMachines** → **Methods**.

**NOTE**

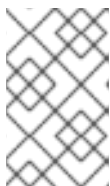
DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.

4. Either create a new instance or select the **clone_to_service** instance.
5. Click  (**Configuration**), then  (**Edit Selected Instance**).
6. In the **configurechilddialog** value, put the path to the method.
7. Click **Save** or **Add** if you are adding this to a new instance.

5.4. CATALOGS

Catalogs are used to create groups of virtual machines or instances for provisioning. For example, a complete package of a database server, desktop with specialized software already on it, and a firewall. You will need to complete the following steps to create and provision a service catalog.

1. Create **Catalog Items** for each virtual machine or instance that will be part of the service.
2. Create a **Service** dialog. For example, create a dropdown with three options small, medium, and large.
3. Create a method for the Service Dialog. This method defines what each of the options means to each of the individual virtual machines or cloud instances for the service. This method is called from a service provisioning instance in the Automate model.
4. Create an instance in the **DOMAIN/Service/Provisioning/StateMachines/ServiceProvision_Template** class that calls the method.



**NOTE**

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.

5. Associate method with Automate instance.
6. Create a **Catalog Bundle**, adding each of the catalog items to it. Select the **Service Dialog** you created. Use the instance created in the **DOMAIN/Service/Provisioning/StateMachines/ServiceProvision_Template** class as the **Entry Point**. Check **Display in Catalog** box.
7. Provision a service.

5.4.1. Creating a Catalog Bundle

1. Navigate to **Services** → **Catalogs**.
2. Click the **Catalog Items** accordion.

- Click  (**Configuration**), and then  (**Add a New Catalog Bundle**).
- In **Basic Info**, enter a name and description:

Basic Info	
Name / Description	<input type="text"/> / <input type="text"/> <input type="checkbox"/> Display in Catalog
Cost	<input type="text"/>

- Click **Display in Catalog**.
- Select the appropriate dialog name.
- Select the path to the appropriate **ServiceProvision** instance.
- Click on the **Resources** tab, then select the catalog items you want added to the bundle from the **Add a Resource** dropdown.
- Click **Add**.

A catalog bundle is created and visible in the **Service Catalog** accordion.





NOTE

You should also create and specify an Entry Point in the **DOMAIN/Service/Provisioning/StateMachines/Methods/CatalogBundle** class for each catalog item that is part of a bundle. If you do not, then the pre and post provision processing will occur for each item in the bundle in addition to processing for the **Catalog Bundle**. To set the entry point, go into each **Catalog Item** and check **Display in Catalog**. Then, you will see the **Entry Point** field.

5.4.2. Creating a Catalog Item

Create a catalog item for each virtual machine or cloud instance that will be part of the service.

- Navigate to **Services** → **Catalogs**.
- Click the **Catalog Items** accordion.
- Click  (**Configuration**), and then  (**Add a New Catalog Item**).
- Select the **Catalog Item Type** you are adding. This list only shows items related to providers available or options activated in the appliance (for example, **Ansible Playbook** is available as a **Catalog Item Type** option if the Embedded Ansible server role is enabled on the appliance).
- In the **Basic Info** subtab:
 - Type a **Name/Description**.
 - Check **Display in Catalog** to edit **Catalog**, **Dialog**, and **Entry Point(NS/CIs/Inst)** options.
 - Provisioning Entry Point (Domain/NS/CIs/Inst)** requires you to select an Automate instance to run upon provisioning.

- ii. **Retirement Entry Point (Domain/NS/Cls/Inst)** requires you to select an Automate instance to run upon retirement.

[Basic Info](#)

[Details](#)

Basic Info

Name / Description	test / test	<input checked="" type="checkbox"/> Display in Catalog
Catalog	<_Unassigned>	
Dialog	<No Dialog>	
Provisioning Entry Point (NS/Cls/Inst)	/Service/Provisioning/StateMachines/ServiceProvisic	X
Reconfigure Entry Point (NS/Cls/Inst)		
Retirement Entry Point (NS/Cls/Inst)	/Service/Retirement/StateMachines/ServiceRetireme	X

Add

Cancel



NOTE

The entry point must be a State Machine since the **Provisioning Entry Point** list is filtered to only show State Machine class instances. No other entry points will be available from the **Provisioning Entry Point** field.





NOTE

You can only choose from the catalogs and dialogs you have already created. If you haven't done so, leave the values blank and edit later.

- In the **Details** subtab, write a **Long Description** for the catalog item.
- In the **Request Info** subtab, select provisioning options that apply to the provider chosen. For more information, refer to [Section 2.8, "Provisioning Virtual Machines"](#) and [Section 2.9, "Provisioning Instances"](#).
- Click **Add**.

5.4.3. Creating a Generic Catalog Item

Create generic catalog items for services non-specific to virtualization or cloud environments. This catalog item type can serve a wide array of needs, from creating a vLAN across a network to accessing virtual machine IP addresses and adding them to a load balancer pool.

- Navigate to **Services** → **Catalogs**.
- Click the **Catalog Items** accordion.
- Click  (**Configuration**), and then  (**Add a New Catalog Item**).
- Select **Generic** from the Catalog Item Type list.
- In the **Basic Info** subtab:

a. Type a **Name/Description**

- a. Type a **Name/Description**.
- b. Check **Display in Catalog** to display the item in the catalog. A **Dialog** will be required if you select **Display in Catalog**.
- c. Choose a **Catalog** to which to add the new item.
- d. Select a **Dialog** from the available options.
- e. Choose a **Subtype** from the list menu.
- f. Add **Entry Point(NS/Cls/Inst)** options.
 - i. **Provisioning Entry Point (Domain/NS/Cls/Inst)** requires you to select an Automate instance to run upon provisioning.
 - ii. **Retirement Entry Point (Domain/NS/Cls/Inst)** requires you to select an Automate instance to run upon retirement.

**NOTE**

The entry point must be a State Machine since the **Provisioning Entry Point** list is filtered to only show State Machine class instances. No other entry points will be available from the **Provisioning Entry Point** field.

6. In the **Details** subtab, write a **Long Description** for the catalog item.
7. Click **Add**.



5.4.4. Creating an Ansible Playbook Service Catalog Item

Create a catalog item that uses an Ansible Playbook to back it.

**NOTE**

- Before creating an Ansible service, at least one repository, one playbook, and one credential must exist in the Red Hat CloudForms inventory. Check your inventory and add the appropriate resources before creating an Ansible service. For more information, see [Automation Management Providers](#) in *Managing Providers*.
- Debugging verbosity is available for Ansible playbook catalog items. Selecting a higher verbosity value provides more detailed output as the playbook executes. **0 (Normal)** is the default value. **1 (Verbose)** will yield return data while a value of **3 (Debug)** provides connection attempt and task invocation details. Higher levels, such as **4 (Connection)** can be useful for debugging SSH connections. Use **5 (WinRM Debug)** when debugging WinRM connections.
- Using Ansible playbooks to populate dynamic dialog fields is not recommended due to delay times caused by the overhead of interaction between systems.
- Only users with administrator privileges can run a service dialog based on a playbook.

1. Navigate to **Services** → **Catalogs**.
2. In the **Catalog Items** accordion, click on the **All Catalog Items**.

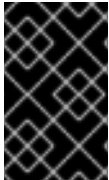
3. Click  (**Configuration**), then  (**Add a New Catalog Item**).
4. Select **Ansible Playbook** from the **Catalog Item Type** drop-down list.
5. Type a **Name** and **Description** for the new service catalog item.
6. Click **Display in Catalog**.
7. Select the appropriate **Catalog** from the drop-down list.
8. In the **Provisioning** tab, set parameters for your catalog item to use by configuring a Playbook to back your service item:
 - a. Choose a **Repository** from the drop-down list.
 - b. Select the **Ansible Playbook** to use.
 - c. Assign the appropriate **Machine Credentials** from the drop-down list.
 - d. Add **Cloud** or **Network Credentials** from the drop-down lists.
 - e. Choose the **Host** against which to run the service item.
 - f. Set the **Max TTL** in minutes. The Time To Live (TTL) field allows you to set the maximum execution time for the playbook to run.
 - g. Use the **Escalate Privilege** toggle switch to enable user privilege escalation if called for in credentials during the playbook run.
 - h. Choose a **Verbosity** value to set the debug level for playbook execution.
 - i. Add key value pairs for **Variables** and their corresponding **Default Values**.
 - j. In the **Dialog** options, choose an existing dialog from the **Use Existing** drop-down list or select **Create New** to add a new dialog.
9. In the **Retirement** tab, set parameters for your catalog item to use by selecting values for the following:
 - a. Choose a **Repository** from the drop-down list.
 - b. Select the **Ansible Playbook** to use.
 - c. Assign the appropriate **Machine Credentials** from the drop-down list.
 - d. Add **Cloud** or **Network Credentials** from the drop-down lists.
 - e. Choose the **Host** against which to run the service item.
 - f. Set the **Max TTL** in minutes. The Time To Live (TTL) field allows you to set the maximum execution time for the playbook to run.
 - g. Use the **Escalate Privilege** toggle switch to enable user privilege escalation if called for in credentials during the playbook run.
 - h. Choose a **Verbosity** value to set the debug level for playbook execution.
 - i. Add key value pairs for **Variables** and their corresponding **Default Values**.

- j. In the **Dialog** options, choose an existing dialog from the **Use Existing** drop-down list or select **Create New** to add a new dialog.

10. Click **Add**.

5.4.5. Creating an Ansible Tower Service Catalog Item



Create a service catalog item from an Ansible Tower template you can use to execute an Ansible Tower playbook in Red Hat CloudForms.







IMPORTANT

You must first create the job template in Ansible Tower. The job templates are automatically discovered by CloudForms when refreshing your Ansible Tower provider's inventory.

First, create a catalog:

1. Navigate to **Services** → **Catalogs**.
2. Click  (**Configuration**), then  (**Add a New Catalog**)
3. Enter a **Name** and **Description** for the catalog.
4. Click **Add**.



Then, create an Ansible Tower service catalog item:

1. Navigate to **Automation** → **Ansible Tower** → **Jobs**.
2. Click **Ansible Tower Job Templates** and select an Ansible Tower job template.
3. Click  (**Configuration**), then  (**Create Service Dialog from this Job Template**).
4. Enter a **Service Dialog Name** (for example, *ansible_tower_job*) and click **Save**.
5. Navigate to **Services** → **Catalogs**. Click **Catalog Items**.
6. Click  (**Configuration**), then  (**Add a New Catalog Item**) to create a new catalog item with the following details, at minimum:
 - For **Catalog Item type**, select **Ansible Tower**.
 - Enter a **Name** for the service catalog item.
 - Select **Display in Catalog**.
 - In **Catalog**, select the catalog you created previously.
 - In **Dialog**, select the service dialog you created previously (in this example, *ansible_tower_job*). **No Dialog** can be selected if the playbook does not require extra variables from the user. To ask the user to enter extra information when running the task, **Service Dialog** must be selected.

- In **Provider**, select your Ansible Tower provider. This brings up the **Ansible Tower Job Template** option and configures the **Provisioning Entry Point State Machine** automatically.
 - Add configuration information for **Reconfigure Entry Point** and **Retirement Entry Point** as applicable.
 - Select your desired **Ansible Tower Job Template** from the list. Generally, this is the Ansible Tower job template previously used to create the service dialog.
7. Click **Add**. The catalog item you created will appear in the **All Service Catalog Items** list.



5.4.6. Creating an Amazon Service Catalog Item

Use the following procedure to create an Amazon catalog item. Once created, the catalog item and service dialog combine with all of the options in the provisioning dialog. Users can then order Red Hat Enterprise Linux instances from the **Service Catalog** in the CloudForms Service user interface.

1. Navigate to **Services** → **Catalogs**, then click on the **Catalog Items** accordion.
2. Click  (**Configuration**), then  (**Add a New Catalog Item**).
3. Select **Amazon** from the **Catalog Item Type** list.
4. Enter the basic details in the **Basic Info** tab:
 - a. Enter a **Name** and **Description** for the new service catalog item.
 - b. Select **Display in Catalog**.
 - c. Select the appropriate catalog from the **Catalog** list.
 - d. Select the appropriate service dialog from the **Dialog** list.
5. Click the **Request Info** tab to enter request details:
 - a. On the **Catalog** tab, select your Amazon AWS image name from the **Name** list, and the number of instances from the **Count** list. The **VM Name** will be overwritten during the provisioning process, but you can enter it as *changeme* for now.
 - b. On the **Properties** tab, select *T2 Micro* from the **Instance Type** list, and *Basic* or *Advanced* for **CloudWatch**. If you plan to access the instance, select a **Guest Access Key Pair**, too.
 - c. On the **Customize** tab, set the **Root Password** under **Credentials**, then select the *Basic root pass template* as a script for cloud-init under **Customize Template**.
6. Click **Add**.

5.4.7. Creating an Azure Service Catalog Item

Use the following procedure to create an Azure catalog item.

1. Navigate to **Services** → **Catalogs**, then click on the **Catalog Items** accordion.
2. Click  (**Configuration**), then  (**Add a New Catalog Item**).

3. Select **Azure** from the **Catalog Item Type** list.
4. Enter the basic details in the **Basic Info** tab:
 - a. Enter a **Name** and **Description** for the new service catalog item.
 - b. Select **Display in Catalog**.
 - c. Select the appropriate catalog from the **Catalog** list.
 - d. Select the appropriate service dialog from the **Dialog** list.
5. Click the **Request Info** tab to enter request details:
 - a. On the **Catalog** tab, select your Azure image name from the **Name** list, and the number of instances from the **Count** list. The **VM Name** will be overwritten during the provisioning process, but you can enter it as *changeme* for now.
 - b. Select appropriate **Environment** settings that are known to work for your Azure environment.
 - c. On the **Customize** tab, set the **Username** and **Password** under **Credentials**, then select the appropriate script under **Customize Template**.
6. Click **Add**.



5.4.8. Creating an OpenShift Template Catalog Item



NOTE



Before adding a new catalog item for container template provisioning, create a service dialog from a container template. See [Section 5.2.3, “Creating a Service Dialog from a Container Template”](#) for details.

Complete the following procedure to create an OpenShift Template catalog item.

1. Navigate to **Services** → **Catalogs**, then click on the **Catalog Items** accordion.
2. Click  (**Configuration**), then  (**Add a New Catalog Item**).
3. Select **OpenShift Template** from the **Catalog Item Type** list.
4. Enter a **Name** and **Description** for the new service catalog item. Select **Display in Catalog**.
5. Select the appropriate catalog from the **Catalog** list.
6. From the **Dialog list**, select the service dialog you have created from a container template.
7. Select your provider from the **Provider** list.
8. Set the **Provisioning Entry Point**.
9. Click **Add**.

5.4.9. Creating an Orchestration Catalog Item

Use the following procedure to create an Orchestration catalog item.

1. Navigate to **Services** → **Catalogs** and select **Catalog Items** in the accordion menu.
2. Click  **Configuration**, then click  **Add a New Catalog Item**. The **Adding a new Service Catalog Item** window is displayed.
3. Select **Orchestration** from the **Catalog Item Type** list.
4. Enter the basic details in the **Basic Info**:
 - a. Enter a **Name** and **Description** for the new service catalog item.
 - b. Select **Display in Catalog** box.
 - c. Select the appropriate catalog from the **Catalog** list.
 - d. Select the appropriate dialog from the **Dialog** list.
 - e. Select the **Orchestration Template** from the list.
5. Click **Add**.

5.4.10. Provisioning a Service

1. Navigate to **Services** → **Catalogs**.
2. Click the **Service Catalogs** accordion, and select the service to provision.
3. Click **Order**.

The parameters are passed to the children based on the method tied to the choices made in the dialog.

5.5. ORCHESTRATION STACKS

Cloud orchestration is a service that allows you to create, update, and manage cloud resources and their software components as a single unit and then deploy them in an automated, repeatable way through a template. Templates use a human-readable syntax and can be defined in text files, thereby allowing users to check them into version control. Templates allow you to easily deploy and reconfigure infrastructure for applications within your cloud. A user can author the stack templates, or can upload them from other sources.

Red Hat CloudForms supports adding Amazon CloudFormation, OpenStack Heat, Microsoft Azure, VNF, and VMware vApp template type, and provides the ability to:

- Inventory stacks and elements of each type into the CloudForms VMDB.
- Model the relationships of instances to their stacks, inclusive of the user interface. For example, selecting an instance within a region that is within a stack, the user interface shows this on the standard instance view.
- Model the stack and its elements in the user interface.



**NOTE**

When importing a template into CloudForms, the selected elements are converted according to their type. For example, lists convert to list boxes, and single items convert to text boxes.

See [Service Provisioning in CloudForms Using an Orchestration Template](#) for an example about creating a service in CloudForms for provisioning virtual machine instances using an AWS CloudFormation orchestration template.



5.5.1. Creating an Orchestration Template

Complete the following procedure to add an orchestration template.

1. Navigate to **Services** → **Catalogs** and select **Orchestration Templates** in the accordion menu.
2. Click  **Configuration**, then click  **Create a new Orchestration Template**.
3. Enter a **Name** and **Description** for your template.
4. Select the template type from the **Template Type** list. The default is Amazon CloudFormation.
5. Select **Draft** to create a draft template.
6. Add your template in the area below for the selected **Template Type**.
7. Click **Add**.

5.5.2. Editing Orchestration Templates

Complete the following procedure to edit orchestration templates.

1. Navigate to **Services** → **Catalogs** and select **Orchestration Templates** in the accordion menu.
2. Select the orchestration template you want to edit from the **All Orchestration Templates** list.
3. Click  **Configuration**, then click  **Edit this Orchestration Template**.
4. Edit the template as needed.

**NOTE**



You can only edit the name and description of a read-only template as there can be stacks associated with the template.

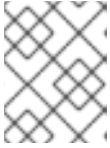
5. Click **Save**.

5.5.3. Copying Orchestration Templates

Complete the following procedure to copy an orchestration template to create a new template.

1. Navigate to **Services** → **Catalogs** and select **Orchestration Templates** in the accordion menu.

2. Click  **Configuration**, then click  **Copy this Orchestration Template**.
3. Change the **Description** and the actual content of the template as required. CloudForms automatically prefixes *Copy of* to the old template **Name**.





NOTE

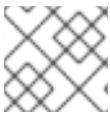
To create a copy of an orchestration template into a new template, the old and new template content must differ.

4. Click **Add**.

5.5.4. Deleting Orchestration Templates

Complete the following procedure to delete orchestration templates.

1. Navigate to **Services** → **Catalogs** and select **Orchestration Templates** in the accordion menu.
2. Select the orchestration template you want to delete from the **All Orchestration Templates** list.
3. Click  **Configuration**, then click  **Remove this Orchestration Template from Inventory**.
4. Click **OK**.



NOTE

Read-only templates cannot be deleted.

CHAPTER 6. RETIREMENT

6.1. RETIRING VIRTUAL MACHINES

6.1.1. Retiring Virtual Machines and Instances

When a virtual machine or instance is no longer required, it can be retired. Once a virtual machine or instance reaches its retirement date, it is immediately shut down and not allowed to restart. If an attempt to restart is made, Red Hat CloudForms will shut down the virtual machine or instance.

There are three built-in policies involved with retirement:

- If the virtual machine or instance reaches its retirement date, it will be stopped even if it is running.
- If a retired virtual machine or instance is requested to start through Red Hat CloudForms, the virtual machine or instance will not be allowed to start.
- If a provider starts a retired virtual machine or instance outside of Red Hat CloudForms, the virtual machine or instance will be stopped.



Red Hat CloudForms provides a number of ways to retire a virtual machine or instance:

- By using the allocated buttons in the Red Hat CloudForms console.
- When creating a provision request, a retirement date can be set up.

6.1.2. Using the Console to Retire a Virtual Machine

Through the Red Hat CloudForms console, you can retire a virtual machine on a specific date or immediately.



6.1.3. Retiring a Virtual Machine Immediately


1. Navigate to **Compute** → **Infrastructure** → **Virtual Machines**.
2. Select the virtual machine or instance that you want to retire.
3. Click  (**Lifecycle**), then  (**Retire this VM/Instance**).

The virtual machine or instance is immediately stopped, and will be shut down if an attempt is made to restart it.

6.1.4. Setting a Retirement Date and Time for a Virtual Machine or Instance




You can schedule virtual machine retirement by specifying a date and time, or by selecting a relative time a number of months, weeks, days or hours ahead of the present time.

1. Navigate to **Compute** → **Infrastructure** → **Virtual Machines**.
2. Select the virtual machine or instance that you want to set a retirement date for.
3. Click  (**Lifecycle**), then  (**Set Retirement Dates**).

4. From **Enter Retirement Date as**, select **Specific Date and Time** or **Time Delay from Now** to schedule retirement.
 - a. To choose a **Specific Date and Time**, click the **Retirement Date and Time** field to open the calendar.
 - i. Select a retirement date using the calendar control.
 - ii. Click  then select a retirement time (in UTC) using the arrows.
 - b. To retire the virtual machine using a relative time, select **Time Delay from Now**.
 - i. From **Time Delay**, specify a retirement time any number of months, weeks, days, or hours in the future using the arrows.
5. Select a **Retirement Warning** if desired.
6. Click **Save**.

The scheduled retirement date and time display in the virtual machine summary screen.

6.1.5. Removing a Retirement Date for a Virtual Machine or Instance

1. Navigate to **Compute** → **Infrastructure** → **Virtual Machines**.
2. Select the virtual machine or instance that you want to remove the retirement date from.
3. Click  (**Lifecycle**), then click  (**Set Retirement Date**).
4. Click  to remove the retirement date.

6.2. SETTING RETIREMENT IN A PROVISION REQUEST

If you are using Red Hat CloudForms to provision, you can set when you want retirement in the provision request. To see how to create a request, see [Chapter 2, Provisioning Requests](#). A warning email will be sent to the owner before the retirement.

6.2.1. Scheduling Retirement in a Provision Request



When provisioning a cloud instance or virtual machine, a multi-tabbed screen appears where you can set up your provision requests.

1. Click the **Schedule** tab to set when to provision your request and the lifespan of the virtual machine or instance.
2. In **Lifespan**, you can choose to power on the virtual machines or instances after creation and set the **Time until Retirement**. If you select the time until retirement, you will select **Retirement Warning** accordingly.
3. Click **Submit**.

6.3. EXTENDING RETIREMENT DATES

Red Hat CloudForms **Automate** includes a method to extend the retirement of a virtual machine or instance by 14 days. This section describes how to create a button that invokes this method and how to edit the method to change the number of days.

6.3.1. Creating a Custom Button to Extend Retirement

1. Navigate to **Automate** → **Customization**.
2. Click the **Buttons** accordion.
3. From the **Object Types** tree, select **VM and Instance**.
4. Navigate to the button group to which you want to add this button. (If you do not have a button group, add one and then create the button.)
5. Click  (**Configuration**), then  (**Add a new Button**).
6. Type in a button text and button hover text, and select the image you want to use.
7. In **Object Details**, select **Request** from the **/System/Process/** dropdown. By default, the message is **create**. Do not change it.
8. In **Request**, type **vm_retire_extend**.
9. Click **Add**.

6.3.2. Changing the Number of Days to Extend Retirement



1. Navigate to **Automate** → **Explorer**.
2. Click **DOMAIN** → **Cloud** → **VM** → **Retirement** → **Email** → **vm_retire_extend**.



NOTE

DOMAIN must be a user-defined Domain and not the locked ManageIQ Domain. If necessary, you can copy the class from the ManageIQ domain into a custom domain.






This example uses the **Cloud** Namespace, but you can also use the **Infrastructure** namespace.

- 3. Click  (**Configuration**), then  (**Edit this Instance**).
- 4. In the Value field, change the **vm_retire_extend_days** attribute to the new value.
- 5. Click **Save**.

Main Info

Fully Qualified Name	ManageIQ-Test / Cloud / VM / Retirement / Email				
Name	<input type="text" value="vm_retire_extend"/>				
Display Name	<input type="text"/>				
Description	<input type="text"/>				

Fields

Name	Value	On Entry	On Exit	On Error	Collect
 ✓ (to_email_address)	<input type="text" value="evmadmin@company.c"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
 ✓ (from_email_address)	<input type="text" value="evmadmin@company.c"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
 ✓ (signature)	<input type="text" value="Virtualization Infrastruc"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
 ✓ (vm_retire_extend_days)	<input type="text" value="14"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
 ✓ (method1)	<input type="text" value="vm_retire_extend"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Save

Reset

Cancel