Advanced Service Design

vRealize Automation 6.2

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see http://www.vmware.com/support/pubs.
You can find the most up-to-date technical documentation on the VMware Web site at:
http://www.vmware.com/support/
The VMware Web site also provides the latest product updates.
If you have comments about this documentation, submit your feedback to:
docfeedback@vmware.com
# Contents

- Advanced Service Design 7
- Updated Information 9

1 Using the Goal Navigator 11

2 Introduction to VMware vRealize Orchestrator 13
   - VMware vRealize Orchestrator Overview 13
   - vRealize Orchestrator Plug-Ins Overview 13
   - vRealize Orchestrator Integration in vRealize Automation 15
   - Log in to the vRealize Orchestrator Configuration Interface 17
   - Log in to the vRealize Orchestrator Client 17

3 Configuring vRealize Orchestrator and Plug-Ins 19
   - Configure an External vRealize Orchestrator Server 19
   - Configure the Default Workflow Folder for a Tenant 20
   - Configuring Advanced Service Endpoints 21
     - Configure the Active Directory Plug-In as an Endpoint 21
     - Configure the HTTP-REST Plug-In as an Endpoint 22
     - Configure the PowerShell Plug-In as an Endpoint 24
     - Configure the SOAP Plug-In as an Endpoint 25
     - Configure the vCenter Server Plug-In as an Endpoint 27
     - Update an Endpoint 28
     - Delete an Endpoint 29
   - Installing Additional Plug-Ins on the Default vRealize Orchestrator Server 29

4 Authoring Services with Advanced Service Designer 31
   - Advanced Service Designer Basics 31
     - Custom Resources 31
     - Service Blueprints 32
     - Resource Actions 32
     - Resource Mappings 32
   - Creating an Advanced Service Overview 32
     - Create a Custom Resource 33
     - Create a Service Blueprint 34
     - Publish a Service Blueprint as a Catalog Item 36
     - Create a Resource Action 36
     - Publish a Resource Action 38
     - Assign an Icon to a Resource Action 38
     - Create a Service 39
     - Associate a Catalog Item with a Service 39
Create an Approval Policy for Advanced Service Blueprints and Actions 40
Entitle a Service, Catalog Items, and Actions to a User or a Group of Users 42
Extending Operations on Resources Provisioned by Other Sources 43
Extending Operations Checklist 44
Resource Mapping Script Actions and Workflows 45
Create a Resource Mapping 45
Create a Resource Action for an Item Provisioned by Another Source 46
Publish a Resource Action 48
Assign an Icon to a Resource Action 48
Importing and Exporting Advanced Service Designer Components 49
Export Advanced Service Designer Components 49
Import Advanced Service Designer Components and Workflows 49

5 Form Designer 51
Fields in the Form Designer 52
Constraints and Values in the Form Designer 53
External Value Definitions in the Form Designer 55
Working With the Form Designer 55
Designing a Custom Resource Form 56
Designing a Service Blueprint Form 59
Designing a Resource Action Form 64

6 Advanced Service Designer Examples and Scenarios 69
Create a Service for Creating and Modifying a User 69
Create a Test User as a Custom Resource 70
Create a Service Blueprint for Creating a User 71
Publish the Create a User Blueprint as a Catalog Item 72
Create a Resource Action to Change a User Password 72
Publish the Change a Password Resource Action 73
Create a Service for Creating a Test User 73
Associate the Catalog Item with the Create a Test User Service 73
Entitle the Service and the Resource Action to a Consumer 74
Create and Publish an Action to Migrate a Virtual Machine 75
Create a Resource Action to Migrate a vSphere Virtual Machine 75
Publish the Action for Migrating a vSphere Virtual Machine 76
Create an Action to Migrate a Virtual Machine With vMotion 76
Create an Action to Migrate a vSphere Virtual Machine With vMotion 77
Edit the Resource Action Form 77
Add a Submitted Action Details Form and Save the Action 78
Publish the Action for Migrating a Virtual Machine with vMotion 79
Create and Publish an Action to Take a Snapshot 79
Create the Action to Take a Snapshot of a vSphere Virtual Machine 80
Publish the Action for Taking a Snapshot 81
Create and Publish an Action to Start an Amazon Virtual Machine 81
Create a Resource Mapping for Amazon Instances 82
Create a Resource Action to Start an Amazon Virtual Machine 82
Publish the Action for Starting Amazon Instances 83
Contents

Creating a Data Refresh Service Offering 83
Preparing a Data Refresh Environment Checklist 84
Provisioning a Golden Database Node Checklist 85
Requesting a Golden Database Clone Checklist 86

Index 89
Advanced Service Design

*Advanced Service Design* provides information about authoring advanced services in VMware vRealize Automation.

*Advanced Service Design* contains instructions about creating the major building blocks of an advanced service, such as resource types, resource mappings, service blueprints, and resource actions. Additionally, information about managing catalog items, organizing them into services and publishing the services to service catalogs is provided.

This documentation also contains information about VMware vRealize Orchestrator and the vRealize Orchestrator plug-ins.

**NOTE** Not all features and capabilities of vRealize Automation are available in all editions. For a comparison of feature sets in each edition, see [https://www.vmware.com/products/vrealize-automation/](https://www.vmware.com/products/vrealize-automation/).

**Intended Audience**

This information is intended for advanced service architects, business group managers, tenant administrators, approval administrators, and system administrators who are familiar with virtual machine technology, data center management operations, and who also have strong experience with vRealize Automation and vRealize Orchestrator.

**VMware Technical Publications Glossary**

VMware Technical Publications provides a glossary of terms that might be unfamiliar to you. For definitions of terms as they are used in VMware technical documentation, go to [http://www.vmware.com/support/pubs](http://www.vmware.com/support/pubs).
Updated Information

This *Advanced Service Design* is updated with each release of the product or when necessary.

This table provides the update history of the *Advanced Service Design*.

<table>
<thead>
<tr>
<th>Revision</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>001646-02</td>
<td>Updated “<em>Constraints and Values in the Form Designer,</em>” on page 53 with additional information regarding constraint values.</td>
</tr>
<tr>
<td>001646-01</td>
<td>Added the new configuration location for Import Content in “<em>Import Advanced Service Designer Components and Workflows,</em>” on page 49.</td>
</tr>
<tr>
<td></td>
<td>Added the new configuration location for Export Content in “<em>Export Advanced Service Designer Components,</em>” on page 49.</td>
</tr>
<tr>
<td></td>
<td>Added the new configuration location for Orchestration Server Configuration in “<em>Configure an External vRealize Orchestrator Server,</em>” on page 19.</td>
</tr>
<tr>
<td></td>
<td>Added the new configuration location for Orchestration plug-in Endpoints in the following topics:</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the Active Directory Plug-In as an Endpoint,</em>” on page 21</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the HTTP-REST Plug-In as an Endpoint,</em>” on page 22</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the PowerShell Plug-In as an Endpoint,</em>” on page 24</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the SOAP Plug-In as an Endpoint,</em>” on page 25</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the vCenter Server Plug-In as an Endpoint,</em>” on page 27</td>
</tr>
<tr>
<td></td>
<td>- “<em>Delete an Endpoint,</em>” on page 29</td>
</tr>
<tr>
<td></td>
<td>Updated user interface labels for Orchestration plug-in Endpoints in the following topics:</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the Active Directory Plug-In as an Endpoint,</em>” on page 21</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the HTTP-REST Plug-In as an Endpoint,</em>” on page 22</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the PowerShell Plug-In as an Endpoint,</em>” on page 24</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the SOAP Plug-In as an Endpoint,</em>” on page 25</td>
</tr>
<tr>
<td></td>
<td>- “<em>Configure the vCenter Server Plug-In as an Endpoint,</em>” on page 27</td>
</tr>
</tbody>
</table>

| 001646-00  | Initial release.                                                                                                                            |
Using the Goal Navigator

The goal navigator guides you through high-level goals that you might want to accomplish in vRealize Automation.

The goals you can achieve depend on your role. To complete each goal, you must complete a sequence of steps that are presented on separate pages in the vRealize Automation console.

The goal navigator can answer the following questions:

- Where do I start?
- What are all the steps I need to complete to achieve a goal?
- What are the prerequisites for completing a particular task?
- Why do I need to do this step and how does this step help me achieve my goal?

The goal navigator is hidden by default. You can expand the goal navigator by clicking the icon on the left side of the screen.

After you select a goal, you navigate between the pages needed to accomplish the goal by clicking each step. The goal navigator does not validate that you completed a step, or force you to complete steps in a particular order. The steps are listed in the recommended sequence. You can return to each goal as many times as needed.

For each step, the goal navigator provides a description of the task you need to perform on the corresponding page. The goal navigator does not provide detailed information such as how to complete the forms on a page. You can hide the page information or move it to a more convenient position on the page. If you hide the page information, you can display it again by clicking the information icon on the goal navigator panel.
VMware vRealize Orchestrator is an automation and management engine that helps you to automate your cloud and integrate the VMware vCloud Suite with the rest of your management systems.

vRealize Orchestrator allows administrators and architects to develop complex automation tasks by using the workflow designer, and then quickly access and run workflows directly from vSphere Web Client or by various triggering mechanisms.

vRealize Orchestrator can access and control external technologies and applications by using vRealize Orchestrator plug-ins.

This chapter includes the following topics:
- “VMware vRealize Orchestrator Overview,” on page 13
- “vRealize Orchestrator Plug-Ins Overview,” on page 13
- “vRealize Orchestrator Integration in vRealize Automation,” on page 15
- “Log in to the vRealize Orchestrator Configuration Interface,” on page 17
- “Log in to the vRealize Orchestrator Client,” on page 17

VMware vRealize Orchestrator Overview

vRealize Orchestrator is a development- and process-automation platform that provides a library of workflows. You can use the vRealize Orchestrator workflows to create and run automated, configurable processes to manage the VMware® vSphere infrastructure as well as other VMware and third-party technologies.

By default, vRealize Orchestrator exposes every operation in the VMware vCenter Server API, and you can integrate all of these operations into your automated processes. By using vRealize Orchestrator, you can also integrate with other management and administration solutions through its open plug-in architecture.

For information about the latest vRealize Orchestrator releases, see the VMware vRealize Orchestrator Documentation landing page.

vRealize Orchestrator Plug-Ins Overview

With plug-ins you can use vRealize Orchestrator to access and control external technologies and applications. By exposing an external technology in a vRealize Orchestrator plug-in, you can incorporate objects and functions in workflows that access the objects and functions of the external technology.

The external technologies that you can access by using plug-ins can include virtualization management tools, email systems, databases, directory services, remote control interfaces, and so on.
You can use the standard set of vRealize Orchestrator plug-ins to incorporate external technologies such as the vCenter Server API and email capabilities into workflows. In addition, you can use the vRealize Orchestrator open plug-in architecture to develop plug-ins to access other applications.

### Table 2-1. Plug-Ins Included by Default in vRealize Orchestrator

<table>
<thead>
<tr>
<th>Plug-In</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>vCenter Server</td>
<td>Provides access to the vCenter Server API so that you can incorporate all of the vCenter Server objects and functions into the management processes that you automate by using vRealize Orchestrator.</td>
</tr>
<tr>
<td>Configuration</td>
<td>Provides workflows for configuring the vRealize Orchestrator authentication, database connection, SSL certificates, and so on.</td>
</tr>
<tr>
<td>vCO Library</td>
<td>Provides workflows that act as basic building blocks for customization and automation of client processes. The workflow library includes templates for lifecycle management, provisioning, disaster recovery, hot backup, and other standard processes. You can copy and edit the templates to modify them according to your needs.</td>
</tr>
<tr>
<td>SQL</td>
<td>Provides the Java Database Connectivity (JDBC) API, which is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases. The databases include SQL databases and other tabular data sources, such as spreadsheets or flat files. The JDBC API provides a call-level API for SQL-based database access from workflows.</td>
</tr>
<tr>
<td>SSH</td>
<td>Provides an implementation of the Secure Shell v2 (SSH-2) protocol. Allows remote command and file transfer sessions with password and public key-based authentication in workflows. Supports keyboard-interactive authentication. Optionally, the SSH plug-in can provide remote file system browsing directly in the vRealize Orchestrator client inventory.</td>
</tr>
<tr>
<td>XML</td>
<td>A complete Document Object Model (DOM) XML parser that you can implement in workflows. Alternatively, you can use the ECMAScript for XML (E4X) implementation in the vRealize Orchestrator JavaScript API.</td>
</tr>
<tr>
<td>Mail</td>
<td>Uses Simple Mail Transfer Protocol (SMTP) to send email from workflows.</td>
</tr>
<tr>
<td>Net</td>
<td>Wraps the Jakarta Apache Commons Net Library. Provides implementations of Telnet, FTP, POP3, and IMAP. The POP3 and IMAP part is used for reading email. In combination with the Mail plug-in, the Net plug-in provides complete email send and receive capabilities in workflows.</td>
</tr>
<tr>
<td>Workflow documentation</td>
<td>Provides workflows that let you generate information in PDF format about a workflow or a workflow category.</td>
</tr>
<tr>
<td>Enumeration</td>
<td>Provides common enumerated types that can be used in workflows by other plug-ins.</td>
</tr>
<tr>
<td>HTTP-REST</td>
<td>Lets you manage REST Web services by providing interaction between vCenter Orchestrator and REST hosts.</td>
</tr>
<tr>
<td>SOAP</td>
<td>Lets you manage SOAP Web services by providing interaction between vCenter Orchestrator and SOAP hosts.</td>
</tr>
<tr>
<td>AMQP</td>
<td>Lets you interact with Advanced Message Queuing Protocol (AMQP) servers also known as brokers.</td>
</tr>
<tr>
<td>SNMP</td>
<td>Enables vCenter Orchestrator to connect and receive information from SNMP-enabled systems and devices.</td>
</tr>
<tr>
<td>Active Directory</td>
<td>Provides interaction between vCenter Orchestrator and Microsoft Active Directory.</td>
</tr>
<tr>
<td>vCO WebOperator</td>
<td>A Web view that lets you access the workflows in the vRealize Orchestrator library and interact with them across a network by using a Web browser.</td>
</tr>
</tbody>
</table>
Table 2-1. Plug-Ins Included by Default in vRealize Orchestrator (Continued)

<table>
<thead>
<tr>
<th>Plug-In</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic Types</td>
<td>Lets you define dynamic types and create and use objects of these dynamic types.</td>
</tr>
<tr>
<td>Multi-Node</td>
<td>Contains workflows for hierarchical orchestration, management of Orchestrator instances, and scale-out of Orchestrator activities.</td>
</tr>
<tr>
<td>PowerShell</td>
<td>Lets you manage PowerShell hosts and run custom PowerShell operations.</td>
</tr>
<tr>
<td>vRealize Automation (only in the instance embedded in vRealize Automation)</td>
<td>Lets you create and run workflows for interaction between vRealize Orchestrator and vRealize Automation.</td>
</tr>
</tbody>
</table>

For more information about the vRealize Orchestrator plug-ins that VMware develops and distributes, see the VMware vRealize Orchestrator Documentation landing page.

**vRealize Orchestrator Integration in vRealize Automation**

vRealize Orchestrator is the workflow engine integrated in vRealize Automation.

The vRealize Orchestrator server distributed with vRealize Automation is preconfigured, and therefore when your system administrator deploys the vRealize Automation Appliance, the vRealize Orchestrator server is up and running.

**Figure 2-1. Create and Request Catalog Items Included in an Advanced Service to Provision a Custom Resource**

Service architects can map vRealize Orchestrator objects to custom resources and vRealize Orchestrator workflows to service blueprints in vRealize Automation. They can then publish the service blueprints as catalog items and combine the items into an advanced service. Tenant administrators or business group managers can display the service to the catalog of the consumers by creating an entitlement. When the consumers request a catalog item, vRealize Automation runs a vRealize Orchestrator workflow and provisions the custom resource.
Service architects can also map vRealize Orchestrator workflows to resource actions to extend the management vRealize Automation capabilities. After the consumers provision a custom resource, they can request to run a post-provisioning action. This way, the consumers run a vRealize Orchestrator workflow and modify the provisioned custom resource.

When a consumer requests a catalog item or resource action, the Advanced Service Designer runs the corresponding vRealize Orchestrator workflow passing the following data as global parameters to the workflow:

**Table 2-2. Advanced Services Global Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>__asd_tenantRef</td>
<td>The tenant of the user requesting the workflow.</td>
</tr>
<tr>
<td>__asd_subtenantRef</td>
<td>The business group of the user requesting the workflow.</td>
</tr>
<tr>
<td>__asd_catalogRequestId</td>
<td>The request id from the catalog for this workflow run.</td>
</tr>
<tr>
<td>__asd_requestedFor</td>
<td>The target user of the request. If the request is on behalf of a user, then this is the user on behalf of whom the workflow is requested, otherwise it is the user requesting the workflow.</td>
</tr>
<tr>
<td>__asd_requestedBy</td>
<td>The user requesting the workflow.</td>
</tr>
</tbody>
</table>

If an advanced service blueprint or resource action uses a vRealize Orchestrator workflow that contains a User Interaction schema element, when a consumer requests the service, the workflow suspends its run and waits for the user to provide the required data. To answer to a waiting user interaction, the user must navigate to **Inbox > Manual User Action**.

The default vRealize Orchestrator server inventory is shared across all tenants and cannot be used per tenant. For example, if a service architect creates a service blueprint for creating a cluster compute resource, the consumers from different tenants have to browse through the inventory items of all vCenter Server instances although they might belong to a different tenant.
System administrators can install vRealize Orchestrator or deploy the VMware vRealize Orchestrator Appliance separately to set up an external vRealize Orchestrator instance and configure vRealize Automation to work with that external vRealize Orchestrator instance.

System administrators can also configure vRealize Orchestrator workflow categories per tenant and define which workflows are available to each tenant.

In addition, tenant administrators can also configure an external vRealize Orchestrator instance but only for their own tenants.

For information about configuring an external vRealize Orchestrator instance and vRealize Orchestrator workflow categories, see Configuring vCenter Orchestrator and Plug-Ins.

Log in to the vRealize Orchestrator Configuration Interface

To edit the configuration of the default vRealize Orchestrator instance embedded in vRealize Automation, you must start the vRealize Orchestrator configuration service and log in to the vRealize Orchestrator configuration interface.

The vRealize Orchestrator configuration service is not started by default in the vRealize Automation appliance. You must start the vRealize Orchestrator configuration service to access the vRealize Orchestrator configuration interface.

Procedure

1. Start the vRealize Orchestrator Configuration service.
   a. Log in to the vRealize Appliance Linux console as root.
   b. Enter `service vco-configurator start` and press Enter.


3. Click vRealize Orchestrator Configurator.
   You are redirected to `https://vra-va-hostname.domain.name:8283/vco-config`.

4. Log in to the vRealize Orchestrator Configurator.
   The user name is `vmware` and the default password is `vmware`.

5. (Optional) If this is the first time you are logging in, change the default password and click Apply changes.
   Your new password must be at least eight characters long, and must contain at least one digit, one special character, and one uppercase letter.

Log in to the vRealize Orchestrator Client

To perform general administration tasks or to edit and create workflows in the default vRealize Orchestrator instance, you must log in to the vRealize Orchestrator client.

The vRealize Orchestrator client interface is designed for developers with administrative rights who want to develop workflows, actions, and other custom elements.

Procedure


2. Click vRealize Orchestrator Client.
   You are redirected to the VMware vRealize Orchestrator Login wizard.
3 Type the IP or the domain name of the vRealize Appliance in the Host name text box, and 8281 as a port number.

For example, type vra-hostname.domain.name:8281 or vrealize_automation_appliance_ip:8281.

4 Log in by using the vRealize Orchestrator Client user name and password.

The credentials are the vCenter Single Sign-On default tenant administrator user name and password.

5 In the Certificate Warning window select an option to handle the certificate warning.

The vRealize Orchestrator client communicates with the vRealize Orchestrator server by using an SSL certificate. A trusted CA does not sign the certificate during installation. You receive a certificate warning each time you connect to the vRealize Orchestrator server.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ignore</td>
<td>Continue using the current SSL certificate. The warning message appears again when you reconnect to the same vRealize Orchestrator server, or when you try to synchronize a workflow with a remote Orchestrator server.</td>
</tr>
<tr>
<td>Cancel</td>
<td>Close the window and stop the login process.</td>
</tr>
<tr>
<td>Install this certificate and do not display any security warnings for it anymore.</td>
<td>Select this check box and click Ignore to install the certificate and stop receiving security warnings.</td>
</tr>
</tbody>
</table>

You can change the default SSL certificate with a certificate signed by a CA. For more information about changing SSL certificates, see Installing and Configuring VMware vRealize Orchestrator.

What to do next

You can import a package, develop workflows, or set root access rights on the system. See Using the VMware vRealize Orchestrator Client and Developing with VMware vRealize Orchestrator.
Configuring vRealize Orchestrator and Plug-Ins

System and tenant administrators can configure vRealize Automation to use an external vRealize Orchestrator server.

In addition, system administrators can also determine the workflow folders that are available to each tenant.

Tenant administrators can configure the vRealize Orchestrator plug-ins as endpoints. They can also import workflow packages.

If you have custom vRealize Orchestrator workflow packages or previously developed Advanced Service Designer components, tenant administrators and service architects can import them into your vRealize Automation. See “Import Advanced Service Designer Components and Workflows,” on page 49.

<table>
<thead>
<tr>
<th>Role</th>
<th>vRealize Orchestrator-Related Configuration Privileges</th>
</tr>
</thead>
<tbody>
<tr>
<td>System administrators</td>
<td>■ Configure the vRealize Orchestrator server for all tenants.</td>
</tr>
<tr>
<td></td>
<td>■ Define the default vRealize Orchestrator workflow folders per tenant.</td>
</tr>
<tr>
<td>Tenant administrators</td>
<td>■ Configure the vRealize Orchestrator server for their own tenant.</td>
</tr>
<tr>
<td></td>
<td>■ Add vRealize Orchestrator plug-ins as endpoints.</td>
</tr>
<tr>
<td></td>
<td>■ Import vRealize Orchestrator workflow packages or Advanced Service Designer components.</td>
</tr>
</tbody>
</table>

This chapter includes the following topics:

- “Configure an External vRealize Orchestrator Server,” on page 19
- “Configure the Default Workflow Folder for a Tenant,” on page 20
- “Configuring Advanced Service Endpoints,” on page 21
- “Installing Additional Plug-Ins on the Default vRealize Orchestrator Server,” on page 29

Configure an External vRealize Orchestrator Server

You can set up vRealize Automation to use an external vRealize Orchestrator server.

System administrators can configure the default vRealize Orchestrator server globally for all tenants. Tenant administrators can configure the vRealize Orchestrator server only for their tenants.

Prerequisites

- Install and configure an external vRealize Orchestrator server. You can also deploy the vRealize Orchestrator Appliance. See Installing and Configuring VMware vCenter Orchestrator.
- Log in to the vRealize Automation console as a system administrator or tenant administrator.
Procedure

1 To specify your Orchestrator server configurations, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Orchestration Configuration &gt; Server Configuration</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Server Configuration</td>
</tr>
</tbody>
</table>

2 Click **Use an external Orchestrator server**.

3 Enter a name and, optionally, a description.

4 Enter the IP or the DNS name of the machine on which the vRealize Orchestrator server runs in the **Host** text box.

5 Enter the port number to communicate with the external vRealize Orchestrator server in the **Port** text box.

   8281 is the default port for vRealize Orchestrator.

6 Select the authentication type.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Sign-On</td>
<td>Connects to the vRealize Orchestrator server by using vCenter Single Sign-On. Note: This option is applicable only if you configured the vRealize Orchestrator and vRealize Automation to use one common vCenter Single Sign-On instance.</td>
</tr>
<tr>
<td>Basic</td>
<td>Connects to the vRealize Orchestrator server with the user name and password that you enter in the <strong>User name</strong> and <strong>Password</strong> text boxes.</td>
</tr>
</tbody>
</table>

7 Click **Test Connection**.

8 Click **Update**.

You configured the connection to the external vRealize Orchestrator server, and the **vCAC workflows folder** and the related utility actions are automatically imported. The **vCAC > ASD workflows folder** contains workflows for configuring endpoints and creating resource mappings.

What to do next


**Configure the Default Workflow Folder for a Tenant**

System administrators can group workflows in different folders and then define workflow categories per tenant. By doing this, a system administrator can grant users from different tenants access to different workflow folders from the same vRealize Orchestrator server.

**Prerequisites**

Log in to the vRealize Automation console as a **system administrator**.

**Procedure**

1 Select **Administration > Advanced Services > Default Orchestrator Folder**.

2 Click the name of the tenant you want to edit.
3. Browse the vRealize Orchestrator workflow library and select a folder.
4. Click Add.

You defined the default vRealize Orchestrator workflow folder for a tenant.

**What to do next**

Repeat the procedure for all of the tenants for which you want to define a default workflow folder.

**Configuring Advanced Service Endpoints**

By configuring advanced service endpoints you can connect the vRealize Automation to your environment. When you configure vRealize Orchestrator plug-ins as endpoints, you use the vRealize Automation user interface to configure the plug-ins instead of using the vRealize Orchestrator configuration interface.

To use vRealize Orchestrator capabilities and the vRealize Orchestrator plug-ins to expose VMware and third-party technologies to vRealize Automation, you can configure the vRealize Orchestrator plug-ins by adding the plug-ins as endpoints. This way, you create connections to different hosts and servers, such as vCenter Server instances, a Microsoft Active Directory host, and so on.

When you add a vRealize Orchestrator plug-in as an endpoint by using the vRealize Automation UI, you run a configuration workflow in the default vRealize Orchestrator server. The configuration workflows are located in the **Orchestrator > vCAC > ASD > Endpoint Configuration** workflows folder.

**IMPORTANT** Configuring a single plug-in in vRealize Orchestrator and in the vRealize Automation console is not supported and results in errors.

**Configure the Active Directory Plug-In as an Endpoint**

You can add an endpoint and configure the Active Directory plug-in to connect to a running Active Directory instance and manage users and user groups, Active Directory computers, organizational units, and so on.

**IMPORTANT** By using the Microsoft Active Directory plug-in, you can orchestrate only one connection to an Active Directory host. You cannot add multiple Microsoft Active Directory instances as endpoints. You cannot delete an Active Directory endpoint. After you add an Active Directory endpoint, you can update it at any time.

**Prerequisites**

- Verify that you have access to a Microsoft Active Directory instance. See the Microsoft Active Directory documentation.
- Log in to the vRealize Automation console as a tenant administrator.

**Procedure**

1. To configure the your Orchestrator plug-in endpoints, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Orchestration Configuration &gt; Endpoints</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Endpoints</td>
</tr>
</tbody>
</table>

2. Click Add (➕).
3. Select **Active Directory** from the **Plug-in** drop-down menu.
4. Click Next.

5. Enter a name and, optionally, a description.

6. Click Next.

7. Configure the Active Directory server details.
   a. Enter the IP address or the DNS name of the host on which Active Directory runs in the **Active Directory host IP/URL** text box.
   b. Enter the lookup port of your Active Directory server in the **Port** text box.
   c. Enter the root element of the Active Directory service in the **Root** text box.
      For example, if your domain name is `mycompany.com`, then your root Active Directory is `dc=mycompany,dc=com`.
      This node is used for browsing your service directory after entering the appropriate credentials. For large service directories, specifying a node in the tree narrows the search and improves performance. For example, rather than searching in the entire directory, you can specify `ou=employees,dc=mycompany,dc=com`. This root element displays all the users in the Employees group.
   d. (Optional) To activate encrypted certification for the connection between vRealize Orchestrator and Active Directory, select **Yes** from the **Use SSL** drop-down menu.
      - **NOTE** The SSL certificate is automatically imported without prompting for confirmation even if the certificate is self-signed.
   e. (Optional) Enter the default domain in the **Default Domain** text box.
      For example, if your domain name is `mycompany.com`, type `@mycompany.com`.

8. Configure the shared session settings.
   a. Enter the user name for the shared session in the **User name for the shared session** text box.
   b. Enter the password for the shared session in the **Password for the shared session** text box.

9. Click Add.

You added an Active Directory instance as an endpoint. Service architects can use the Advanced Service Designer to publish Active Directory plug-in workflows as catalog items and resource actions.

### Configure the HTTP-REST Plug-In as an Endpoint

You can add an endpoint and configure the HTTP-REST plug-in to connect to a REST host.

**Prerequisites**
- Log in to the vRealize Automation console as a **tenant administrator**.
- Verify that you have access to a REST host.
Procedure

1. To configure your Orchestrator plug-in endpoints, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Orchestration Configuration &gt; Endpoints</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Endpoints</td>
</tr>
</tbody>
</table>

2. Click Add (＋).

3. Select HTTP-REST from the Plug-in drop-down menu.

4. Click Next.

5. Enter a name and, optionally, a description.

6. Click Next.

7. Provide information about the REST host.
   a. Enter the name of the host in the Name text box.
   b. Enter the address of the host in the URL text box.

   **Note** If you use Kerberos access authentication, you must provide the host address in FDQN format.

   c. (Optional) Enter the number of seconds before a connection times out in the Connection timeout (seconds) text box.

      The default value is 30 seconds.

   d. (Optional) Enter the number of seconds before an operation times out in the Operation timeout (seconds) text box.

      The default value is 60 seconds.

8. (Optional) Configure proxy settings.
   a. Select Yes to use a proxy from the Use Proxy drop-down menu.
   b. Enter the IP of the proxy server in the Proxy address text box.
   c. Enter the port number to communicate with the proxy server in the Proxy port text box.

9. Click Next.

10. Select the authentication type.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>No authentication is required.</td>
</tr>
<tr>
<td>OAuth 1.0</td>
<td>Uses OAuth 1.0 protocol. You must provide the required authentication parameters under OAuth 1.0.</td>
</tr>
</tbody>
</table>
   a. Enter the key used to identify the consumer as a service provider in the Consumer key text box.
   b. Enter the secret to establish ownership of the consumer key in the Consumer secret text box.
   c. (Optional) Enter the access token that the consumer uses to gain access to the protected resources in the Access token text box.
   d. (Optional) Enter the secret that the consumer uses to establish ownership of a token in the Access token secret text box.
### Option | Action
--- | ---
**OAuth 2.0** | Uses OAuth 2.0 protocol. Enter the authentication token in the **Token** text box.

**Basic** | Provides basic access authentication. The communication with the host is in shared session mode.  
- **a** Enter the user name for the shared session in the **Authentication user name** text box.  
- **b** Enter the password for the shared session in the **Authentication password** text box.

**Digest** | Provides digest access authentication that uses encryption. The communication with the host is in shared session mode.  
- **a** Enter the user name for the shared session in the **Authentication user name** text box.  
- **b** Enter the password for the shared session in the **Authentication password** text box.

**NTLM** | Provides NT LAN Manager (NTLM) access authentication within the Window Security Support Provider (SSP) framework. The communication with the host is in shared session mode.  
- **a** Provide the user credentials for the shared session.  
  - Enter the user name for the shared session in the **Authentication user name** text box.  
  - Enter the password for the shared session in the **Authentication password** text box.  
- **b** Configure the NTLM details  
  - (Optional) Enter the workstation name in the **Workstation for NTLM authentication** text box.  
  - Enter the domain name in the **Domain for NTLM authentication** text box.

**Kerberos** | Provides Kerberos access authentication. The communication with the host is in shared session mode.  
- **a** Enter the user name for the shared session in the **Authentication user name** text box.  
- **b** Enter the password for the shared session in the **Authentication password** text box.

11 Click **Add**.

You configured the endpoint and added a REST host. Service architects can use the Advanced Service Designer to publish HTTP-REST plug-in workflows as catalog items and resource actions.

### Configure the PowerShell Plug-In as an Endpoint

You can add an endpoint and configure the PowerShell plug-in to connect to a running PowerShell host, so that you can call PowerShell scripts and cmdlets from vRealize Orchestrator actions and workflows, and work with the result.

**Prerequisites**
- Verify that you have access to a Windows PowerShell host. For more information about Microsoft Windows PowerShell, see the Windows PowerShell documentation.
- Log in to the vRealize Automation console as a **tenant administrator**.
Procedure

1. To configure the your Orchestrator plug-in endpoints, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Orchestration Configuration &gt; Endpoints</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Endpoints</td>
</tr>
</tbody>
</table>

2. Click **Add** (✚).

3. Select **PowerShell** from the Plug-in drop-down menu.

4. Click **Next**.

5. Enter a name and, optionally, a description.

6. Click **Next**.

7. Specify the PowerShell host details.
   a. Enter the name of the host in the **Name** text box.
   b. Enter the IP address or the FDQN of the host in the **Host/IP** text box.

8. Select the PowerShell host type to which the plug-in connects.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
</table>
| WinRM  | a. Enter the port number to use for communication with the host in the **Port** text box under the PowerShell host details.  
   b. Select a transport protocol from the **Transport protocol** drop-down menu.  
   Note: If you use the HTTPS transport protocol, the certificate of the remote PowerShell host is imported to the vRealize Orchestrator keystore.  
   c. Select the authentication type from the **Authentication** drop-down menu.  
   Note: To use Kerberos authentication, enable it on the WinRM service.  
For information about configuring Kerberos authentication, see Using the PowerShell Plug-In. |
| SSH    | None.                                                                  |

9. Enter the credentials for a shared session communication with the PowerShell host in the **User name** and **Password** text boxes.

10. Click **Add**.

You added an Windows PowerShell host as an endpoint. Service architects can use the Advanced Service Designer to publish PowerShell plug-in workflows as catalog items and resource actions.

Configure the SOAP Plug-In as an Endpoint

You can add an endpoint and configure the SOAP plug-in to define a SOAP service as an inventory object, and perform SOAP operations on the defined objects.

Prerequisites

- Verify that you have access to a SOAP host. The plug-in supports SOAP Version 1.1 and 1.2, and WSDL 1.1 and 2.0.
- Log in to the vRealize Automation console as a tenant administrator.
Procedure

1 To configure the your Orchestrator plug-in endpoints, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Orchestration Configuration &gt; Endpoints</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Endpoints</td>
</tr>
</tbody>
</table>

2 Click Add (+).

3 From the Plug-in drop-down menu, select SOAP.

4 Click Next.

5 Enter a name and, optionally, a description.

6 Click Next.

7 Provide the details about the SOAP host.
   a Enter the name of the host in the Name text box.
   b Select whether to provide the WSDL content as text from the Provide WSDL content drop-down menu.

<table>
<thead>
<tr>
<th>Option Action</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Enter the WSDL text in the WSDL content text box.</td>
</tr>
<tr>
<td>No</td>
<td>Enter the correct path in the WSDL URL text box.</td>
</tr>
</tbody>
</table>

   c (Optional) Enter the number of seconds before a connection times out in the Connection timeout (in seconds) text box.
   The default value is 30 seconds.

   d (Optional) Enter the number of seconds before an operation times out in the Request timeout (in seconds) text box.
   The default value is 60 seconds.

8 (Optional) Specify the proxy settings.
   a To use a proxy, select Yes from the Proxy drop-down menu.
   b Enter the IP of the proxy server in the Address text box.
   c Enter the port number to communicate with the proxy server in the Port text box.

9 Click Next.

10 Select the authentication type.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>No authentication is required.</td>
</tr>
<tr>
<td>Basic</td>
<td>Provides basic access authentication. The communication with the host is in shared session mode.</td>
</tr>
<tr>
<td></td>
<td>a Enter the user name for the shared session in the User name text box.</td>
</tr>
<tr>
<td></td>
<td>b Enter the password for the shared session in the Password text box.</td>
</tr>
<tr>
<td>Digest</td>
<td>Provides digest access authentication that uses encryption. The communication with the host is in shared session mode.</td>
</tr>
<tr>
<td></td>
<td>a Enter the user name for the shared session in the User name text box.</td>
</tr>
<tr>
<td></td>
<td>b Enter the password for the shared session in the Password text box.</td>
</tr>
<tr>
<td>Option</td>
<td>Action</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NTLM</td>
<td>Provides NT LAN Manager (NTLM) access authentication in the Window Security Support Provider (SSP) framework. The communication with the host is in shared session mode.</td>
</tr>
<tr>
<td></td>
<td>a. Provide the user credentials.</td>
</tr>
<tr>
<td></td>
<td>■ Enter the user name for the shared session in the User name text box.</td>
</tr>
<tr>
<td></td>
<td>■ Enter the password for the shared session in the Password text box.</td>
</tr>
<tr>
<td></td>
<td>b. Provide the NTLM settings.</td>
</tr>
<tr>
<td></td>
<td>■ Enter the domain name in the NTLM domain text box.</td>
</tr>
<tr>
<td></td>
<td>(Optional) Enter the workstation name in the NTLM workstation text box.</td>
</tr>
<tr>
<td>Negotiate</td>
<td>Provides Kerberos access authentication. The communication with the host is in shared session mode.</td>
</tr>
<tr>
<td></td>
<td>a. Provide the user credentials.</td>
</tr>
<tr>
<td></td>
<td>1. Enter the user name for the shared session in the User name text box.</td>
</tr>
<tr>
<td></td>
<td>2. Enter the password for the shared session in the Password text box.</td>
</tr>
<tr>
<td></td>
<td>b. Enter the Kerberos service SPN in the Kerberos service SPN text box.</td>
</tr>
</tbody>
</table>

11. Click Add.

You added a SOAP service. Service architects can use the Advanced Service Designer to publish SOAP plug-in workflows as catalog items and resource actions.

**Configure the vCenter Server Plug-In as an Endpoint**

You can add an endpoint and configure the vCenter Server plug-in to connect to a running vCenter Server instance to create service blueprints to manage vSphere inventory objects.

**Prerequisites**

- Install and configure vCenter Server. See *vSphere Installation and Setup*.
- Log in to the vRealize Automation console as a **tenant administrator**.

**Procedure**

1. To configure the your Orchestrator plug-in endpoints, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Orchestration Configuration &gt; Endpoints</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Endpoints</td>
</tr>
</tbody>
</table>

2. Click Add (➕).

3. Select **vCenter Server** from the Plug-in drop-down menu.

4. Click Next.

5. Enter a name and, optionally, a description.

6. Click Next.
7 Provide information about the vCenter Server instance.
   a Enter the IP address or the DNS name of the machine in the IP or host name of the vCenter Server instance to add text box.
      This is the IP address or DNS name of the machine on which the vCenter Server instance you want to add is installed.
   b Enter the port to communicate with the vCenter Server instance in the Port of the vCenter Server instance text box.
      The default port is 443.
   c Enter the location of the SDK to use for connecting to your vCenter Server instance in the Location of the SDK that you use to connect to the vCenter Server instance text box.
      For example, /sdk.

8 Click Next.

9 Define the connection parameters.
   a Enter the HTTP port of the vCenter Server instance in the HTTP port of the vCenter Server instance - applicable for VC plugin version 5.5.2 or earlier text box.
   b Enter the credentials for vRealize Orchestrator to use to establish the connection to the vCenter Server instance in the User name of the user that Orchestrator will use to connect to the vCenter Server instance and Password of the user that Orchestrator will use to connect to the vCenter Server instance text boxes.
      The user that you select must be a valid user with privileges to manage vCenter Server extensions and a set of custom defined privileges.

10 Click Add.

You added a vCenter Server instance as an endpoint. Service architects can use the Advanced Service Designer to publish vCenter Server plug-in workflows as catalog items and resource actions.

Update an Endpoint

After you create an endpoint, and configure a vRealize Orchestrator plug-in, you can edit the endpoint. For example, if the IP of the external system changes, you can update the IP so that vRealize Orchestrator does not lose connection to the host.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator.

Procedure

1 To configure the your Orchestrator plug-in endpoints, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Orchestration Configuration &gt; Endpoints</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Endpoints</td>
</tr>
</tbody>
</table>

2 Select the row of the endpoint to update.

3 Click Edit (-pencil).
Follow the prompts and make your edits. All of the required fields are the same as the ones when you create an endpoint.

Click Update.

Delete an Endpoint

You can delete an advanced service endpoint. The only kind of endpoint that you cannot delete is an Active Directory endpoint.

Prerequisites

Log in to the vRealize Automation console as a tenant administrator.

Procedure

1. To configure your Orchestrator plug-in endpoints, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Orchestration Configuration &gt; Endpoints</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Endpoints</td>
</tr>
</tbody>
</table>

2. Select the row of the endpoint to delete.

3. Click Delete (X).

4. Click OK.

Installing Additional Plug-Ins on the Default vRealize Orchestrator Server

You can install additional plug-ins on the default vRealize Orchestrator server by using the vRealize Orchestrator configuration interface.

Additional plug-ins are not supported for configuration as vRealize Automation endpoints, but you can install additional plug-ins on the default vRealize Orchestrator server and use the workflows with the Advanced Service Designer.

Plug-in installation files are available as .vmoapp or .dar files from either the VMware Solution Exchange website or the vCenter Orchestrator Plug-Ins Documentation.

For more information about installing new plug-ins, see Installing and Configuring VMware vCenter Orchestrator
Creating an advanced service involves a number of tasks.

This chapter includes the following topics:

- “Advanced Service Designer Basics,” on page 31
- “Creating an Advanced Service Overview,” on page 32
- “Extending Operations on Resources Provisioned by Other Sources,” on page 43
- “Importing and Exporting Advanced Service Designer Components,” on page 49

**Advanced Service Designer Basics**

By using Advanced Service Designer you can publish vRealize Orchestrator workflows as catalog items, include them in services, and publish the services in the catalog. You can also extend the vRealize Automation capabilities to manage provisioned items and resources by using custom resource actions.

For example, you can create custom resource actions to extend the post-provisioning actions on vSphere virtual machines provisioned with Infrastructure as a Service (IaaS).

Advanced Service Designer exposes the following objects in vRealize Automation:

- Custom resource
- Resource mapping
- Service blueprint
- Resource action

You can use the objects to create an advanced service in which to include your catalog items, and entitle the service, the catalog items and the resource actions to a user or a group of users within a business group.

**Custom Resources**

You must create a custom resource so that you can create an advanced service for provisioning with the option to access and manage the provisioned items. Custom resources define the items for provisioning, and you can use them to define post-provisioning operations that the consumers can perform.

You create a custom resource to define a new type of provisioned item and map it to an existing vRealize Orchestrator object type. vRealize Orchestrator object types are the objects exposed through the APIs of the vRealize Orchestrator plug-ins. The custom resource is the output type of a blueprint workflow for provisioning and can be the input type for a resource action workflow.
For example, if you have a running vCenter Server instance, and you also have the vCenter Server plug-in that is configured to work with vRealize Orchestrator, all of the object types from the vCenter Server API are exposed in vRealize Orchestrator. The vCenter Server plug-in exposes the vSphere inventory objects in the vRealize Orchestrator inventory. The vSphere inventory objects include data centers, folders, ESXi hosts, virtual machines and appliances, resource pools, and so on. You can perform operations on these objects. For example, you can create, clone, or destroy virtual machines.

For more information about the vRealize Orchestrator object types exposed through the vCenter Server API, see the vCenter Server 5.5 Plug-In API Reference for vCenter Orchestrator.

Service Blueprints

A blueprint is a complete specification of a service.

With service blueprints, you can publish predefined and custom vRealize Orchestrator workflows as catalog items for either requesting or provisioning. Blueprints for requesting run workflows with no provisioning and provide no options for managing a provisioned item. Before you create a blueprint for provisioning, you must map the workflow output parameter as a custom resource. Then you can assign resource actions that define post-provisioning operations.

Resource Actions

You can create custom resource actions to configure the post-provisioning operations that the consumers can perform.

To create post-provisioning operations, you must publish vRealize Orchestrator workflows as resource actions. To create a resource action for an item provisioned by using the Advanced Service Designer, you use a custom resource as an input parameter for the workflow. To create a resource action for an item that is provisioned by a source different from the Advanced Service Designer, you use a resource mapping as an input parameter for the workflow. When you entitle the resource actions, they appear in the Actions dropdown menu of the provisioned items on the Items tab.

Resource Mappings

You create resource mappings between the vRealize Automation catalog resource type and the vRealize Orchestrator inventory type to manage resources provisioned outside of Advanced Service Designer.

For example, you might want to create an action so that users can take a snapshot of their Amazon machines. For this action to work on an Amazon machine provisioned by using IaaS, the three components involved, Advanced Service Designer, vRealize Orchestrator, and IaaS, need a common language for the Amazon machine. You create that common language by adding a resource mapping in Advanced Service Designer that runs a vRealize Orchestrator scripting action or workflow to map the IaaS Cloud Machine resource type to the vRealize Orchestrator AWS:EC2Instance inventory type.

vRealize Automation provides resource mappings, and the underlying vRealize Orchestrator script actions and workflows, for vSphere virtual machines, vCloud Director virtual machines, and vCloud Director vApps.

Creating an Advanced Service Overview

Creating an advanced service and exposing the service to the catalog includes a number of tasks that are performed by different users.

The following is a high-level overview of the sequence of steps required to create and entitle a service for provisioning to a user or a group of users.

1. A service architect creates a custom resource to define the item for provisioning. See “Create a Custom Resource,” on page 33.
A service architect creates a service blueprint to provision the custom resource and publishes the blueprint as a catalog item. See “Create a Service Blueprint,” on page 34 and “Publish a Service Blueprint as a Catalog Item,” on page 36.

A service architect creates and publishes resource actions to define the post-provisioning operations that the consumers of the catalog items can perform on the provisioned items. See “Create a Resource Action,” on page 36 and “Publish a Resource Action,” on page 38.

After creating the custom resource, the blueprint, and the resource actions, a service architect, tenant administrator, or a business group manager creates a service and includes the catalog item in the service. See “Create a Service,” on page 39 and “Associate a Catalog Item with a Service,” on page 39.

A tenant administrator or an approval administrator creates an approval policy for advanced service blueprints and actions. See “Create an Approval Policy for Advanced Service Blueprints and Actions,” on page 40.

A business group manager or a tenant administrator entitles the service, catalog items, and resource actions to a user or a group of users. The users specified are the consumers of the service who can request the catalog items. See “Entitle a Service, Catalog Items, and Actions to a User or a Group of Users,” on page 42.

This is the most common and straightforward scenario for creating an advanced service. It suggests provisioning resource items on the Items tab, defining the post-provisioning operations, adding catalog items in the service, and entitling the service and the resource actions to a consumer.

You can also create and publish advanced services for requesting that do not result in provisioned items on the Items tab, for example, sending notifications. For creating such services, you skip steps 1 and 3.

Create a Custom Resource

To create an advanced service for provisioning, you must create a custom resource to define the item for provisioning.

By creating a custom resource, you map as a resource an object type exposed through the API of a vRealize Orchestrator plug-in. You create a custom resource to define the output parameter of a service blueprint for provisioning and to define an input parameter of a resource action. If you do not create a custom resource, the service blueprint does not provision items on the Items tab and you cannot define post-provisioning operations for the provisioned items.

During the process of creating a custom resource, on the Details Form page, you can specify the fields of the read-only form for the resource that displays information on the details view of a provisioned item. See “Designing a Custom Resource Form,” on page 56.

Prerequisites

Log in to the vRealize Automation console as a service architect.

Procedure

1. Select Advanced Services > Custom Resources.
2. Click Add (+).
3. Enter the vRealize Orchestrator object type in the Orchestrator Type text box and press Enter.
4. Enter a name and, optionally, a description.
5. Click Next.
6  Edit the form of the custom resource.

You can edit the custom resource form by deleting, editing, and rearranging elements. You can also add a new form and form pages and drag elements to the new form and form page.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a form</td>
<td>Click the New Form icon (new) next to the form name, provide the required information, and click Submit.</td>
</tr>
<tr>
<td>Add a form page</td>
<td>Click the New Page icon (page) next to the form page name, provide the required information, and click Submit.</td>
</tr>
<tr>
<td>Add an element to the form page</td>
<td>Drag an element from the New Fields pane on the left to the pane on the right. You can then provide the required information and click Submit. The available elements are specific for the vRealize Orchestrator object type.</td>
</tr>
<tr>
<td>Edit an element</td>
<td>Click the Edit icon (edit) next to the element to edit, make the necessary changes, and click Submit.</td>
</tr>
<tr>
<td>Delete an element</td>
<td>Click the Delete icon (delete) next to the element to delete, and in the confirmation dialog box click OK.</td>
</tr>
<tr>
<td>Delete a form</td>
<td>Click the Delete icon (delete) next to the form name, and in the confirmation dialog box click OK.</td>
</tr>
</tbody>
</table>

7  Click Add.

You created a custom resource and you can see it on the Custom Resources page.

What to do next
Create a service blueprint. See “Create a Service Blueprint,” on page 34.

Create a Service Blueprint

A blueprint is a complete specification for a service. A blueprint determines the components of a service, which are the input parameters, submission and read-only forms, sequence of actions, and provisioning.

You can create service blueprints to provision custom resources that you previously created. When consumers request these catalog items, the provisioned items are stored on the Items tab and you can define post-provisioning operations for this type of provisioned resources.

If you create a service blueprint for provisioning without specifying the output parameter, when the consumers request this catalog item, the blueprint does the provisioning but the provisioned items are not added on the Items tab. You cannot perform post-provisioning operations on this type of provisioned resource.

You can also create service blueprints for requesting that do not have output parameters and do not result in provisioning. For example, you can create a service blueprint for sending notifications.

By creating a service blueprint, you publish a vRealize Orchestrator workflow as a catalog item. During this process you can edit the default generated forms. See “Designing a Service Blueprint Form,” on page 59.

Prerequisites

- Log in to the vRealize Automation console as a service architect.
- For items provisioning, create a custom resource corresponding to the output parameter of the service blueprint.
Procedure

1. Select Advanced Services > Service Blueprints.

2. Click Add (+).

3. Navigate through the vRealize Orchestrator workflow library and select a workflow.
   
   You can see the name and description of the selected workflow, and the input and output parameters as they are defined in vRealize Orchestrator.

4. Click Next.

5. Enter a name and, optionally, a description.
   
   The Name and Description text boxes are prepopulated with the name and description of the workflow as they are defined in vRealize Orchestrator.

6. (Optional) If you do not want to prompt consumers to enter a description and reason for requesting this resource action, select the Hide catalog request information page check box.

7. Click Next.

8. (Optional) Edit the form of the service blueprint on the Blueprint Form page.

   By default, the service blueprint form is mapped to the vRealize Orchestrator workflow presentation. You can edit the blueprint form by deleting, editing, and rearranging the elements in the form. You can also add a new form and form pages and drag elements to the new form and form page.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a form</td>
<td>Click the New Form icon (⁺) next to the form name, provide the required information, and click Submit.</td>
</tr>
<tr>
<td>Edit a form</td>
<td>Click the Edit icon (✏️) next to the form name, make the necessary changes, and click Submit.</td>
</tr>
<tr>
<td>Regenerate the workflow</td>
<td>Click the Rebuild icon (🔨) next to the form name and click OK.</td>
</tr>
<tr>
<td>presentation</td>
<td></td>
</tr>
<tr>
<td>Delete a form</td>
<td>Click the Delete icon (❌) next to the form name, and in the confirmation dialog box click OK.</td>
</tr>
<tr>
<td>Add a form page</td>
<td>Click the New Page icon (⁺) next to the form page name, provide the required information, and click Submit.</td>
</tr>
<tr>
<td>Edit a form page</td>
<td>Click the Edit icon (✏️) next to the form page name, make the necessary changes, and click Submit.</td>
</tr>
<tr>
<td>Delete a form page</td>
<td>Click the Delete icon (❌) next to the form name, and in the confirmation dialog box click OK.</td>
</tr>
<tr>
<td>Add an element to the form page</td>
<td>Drag an element from the New Fields pane on the left to the pane on the right. You can then provide the required information and click Submit.</td>
</tr>
<tr>
<td>Edit an element</td>
<td>Click the Edit icon (✏️) next to the element to edit, make the necessary changes, and click Submit.</td>
</tr>
<tr>
<td>Delete an element</td>
<td>Click the Delete icon (❌) next to the element to delete, and in the confirmation dialog box click OK.</td>
</tr>
</tbody>
</table>

9. Click Next.
10 Select an output parameter from the drop-down menu.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A custom resource that you</td>
<td>When users request this catalog item, the provisioned items are stored on</td>
</tr>
<tr>
<td>previously created</td>
<td>the Items tab.</td>
</tr>
<tr>
<td>No provisioning</td>
<td>The service blueprint does not add new items on the Items tab.</td>
</tr>
</tbody>
</table>

11 Click Add.

You created a service blueprint and you can see it on the Service blueprints page.

**What to do next**

Publish the blueprint as a catalog item. See “Publish a Service Blueprint as a Catalog Item,” on page 36.

**Publish a Service Blueprint as a Catalog Item**

After you create a service blueprint, it is in a draft state and you can publish it as a catalog item.

**Prerequisites**

Log in to the vRealize Automation console as a **service architect**.

**Procedure**

1. Select **Advanced Services > Service Blueprints**.
2. Select the row of the service blueprint to publish, and click **Publish**.

The status of the service blueprint changes to Published. If you select **Administration > Catalog Management > Catalog Items**, you can see that the blueprint is published as a catalog item.

**What to do next**

You can create a resource action to define the operations that the consumers can perform against the item they provision. See “Create a Resource Action,” on page 36. If you do not want to create a resource action, you can create a service and include the catalog item in it. For more information about creating a service, see “Create a Service,” on page 39.

**Create a Resource Action**

As a service architect, you can create resource actions to define the operations that consumers can perform on the provisioned items.

By creating a resource action, you associate a vRealize Orchestrator workflow as a post-provisioning operation. During this process, you can edit the default submission and read-only forms. See “Designing a Resource Action Form,” on page 64.

**Prerequisites**

- Log in to the vRealize Automation console as a **service architect**.
- Create a custom resource corresponding to the input parameter of the resource action.

**Procedure**

1. Select **Advanced Services > Resource Actions**.
2. Click **Add (✚)**.
3 Navigate through the vRealize Orchestrator workflow library and select a workflow.
You can see the name and description of the selected workflow, and the input and output parameters as they are defined in vRealize Orchestrator.

4 Click Next.

5 Select the custom resource that you previously created from the Resource type drop-down menu.

6 Select the input parameter for the resource action from the Input parameter drop-down menu.

7 Click Next.

8 Enter a name and, optionally, a description.

The Name and Description text boxes are prepopulated with the name and description of the workflow as they are defined in vRealize Orchestrator.

9 (Optional) If you do not want to prompt consumers to enter a description and reason for requesting this resource action, select the Hide catalog request information page check box.

10 (Optional) Select the type of the action.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disposing</td>
<td>The input parameter of the resource action workflow is disposed and the item is removed from the Items tab. For example, the resource action is for deleting a provisioned machine.</td>
</tr>
<tr>
<td>Provisioning</td>
<td>The resource action is for provisioning. For example, the resource action is for copying a catalog item. From the drop-down menu, select an output parameter. You can select a custom resource that you previously created so that when the consumers request this resource action, the provisioned items are added on the Items tab. If you have only the No provisioning option, either the resource action is not for provisioning, or you did not create a proper custom resource for the output parameter, and you cannot proceed.</td>
</tr>
</tbody>
</table>

Depending on the action workflow, you can select one, both, or none of the options.

11 Select the conditions under which the resource action is available to users, and click Next.

12 (Optional) Edit the form of the resource action on the Form tab.

The form of the resource action maps the vRealize Orchestrator workflow presentation. You can change the form by deleting, editing, and rearranging the elements. You can also add a new form and form pages and drag the necessary elements to the new form and form page.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a form</td>
<td>Click the New Form icon (.addButton) next to the form name, provide the required information, and click Submit.</td>
</tr>
<tr>
<td>Edit a form</td>
<td>Click the Edit icon (editButton) next to the form name, make the necessary changes, and click Submit.</td>
</tr>
<tr>
<td>Regenerate the workflow presentation</td>
<td>Click the Rebuild icon (rebuildButton) next to the form name and click OK.</td>
</tr>
<tr>
<td>Delete a form</td>
<td>Click the Delete icon (deleteButton) next to the form name, and in the confirmation dialog box click OK.</td>
</tr>
<tr>
<td>Add a form page</td>
<td>Click the New Page icon (addPageButton) next to the form page name, provide the required information, and click Submit.</td>
</tr>
<tr>
<td>Edit a form page</td>
<td>Click the Edit icon (editPageButton) next to the form page name, make the necessary changes, and click Submit.</td>
</tr>
</tbody>
</table>
### Option | Action
--- | ---
Delete a form page | Click the Delete icon (❌) next to the form name, and in the confirmation dialog box click OK.
Add an element to the form page | Drag an element from the New Fields pane on the left to the pane on the right. You can then provide the required information and click Submit.
Edit an element | Click the Edit icon (🛠️) next to the element to edit, make the necessary changes, and click Submit.
Delete an element | Click the Delete icon (❌) next to the element to delete, and in the confirmation dialog box click OK.

13 Click Add.

You created a resource action and you can see it listed on the Resource Actions page.

**What to do next**
Publish the resource action. See “Publish a Resource Action,” on page 38.

### Publish a Resource Action

The newly created resource action is in draft state, and you must publish the resource action.

**Prerequisites**
Log in to the vRealize Automation console as a service architect.

**Procedure**
1 Select Advanced Services > Resource Actions.
2 Select the row of the resource action to publish, and click Publish.

The status of the resource action changes to Published.

**What to do next**
Assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38. Business group managers and tenant administrators can then use the action when they create an entitlement.

### Assign an Icon to a Resource Action

After you create and publish a resource action, you can edit it and assign an icon to the action.

**Prerequisites**
Log in to the vRealize Automation console as a service architect.

**Procedure**
1 Select Administration > Catalog Management > Actions.
2 Select the resource action that you created.
3 Click Configure.
4 Click Browse and select the icon to add.
5 Click Open.
6 Click Update.
You assigned an icon to the resource action. Business group managers and tenant administrators can use the resource action in an entitlement.

Create a Service

You can create a service to organize catalog items into related offerings.

**Prerequisites**

Log in to the vRealize Automation console as a tenant administrator or service architect.

**Procedure**

1. Select Administration > Catalog Management > Services.
2. Click Add (➕).
3. Enter a name and, optionally, a description.
4. (Optional) Click Browse to select an icon for the service and click Open.
5. (Optional) Select a status for the service from the Status drop-down menu.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inactive</td>
<td>Service is saved in a draft state and cannot be published to the catalog.</td>
</tr>
<tr>
<td>Active</td>
<td>Service can be published in the catalog.</td>
</tr>
</tbody>
</table>

6. (Optional) Configure information to display to users of the service when they browse the catalog.
   a. Select the availability of internal support services teams from the Hours drop-down menus.
   b. Enter the name of the service owner in the Owner text box and press Enter.
   a. Enter the name of the person or group of persons in charge of the service in the Support Team text box and press Enter.
   b. Select a time for scheduled maintenance from the Change Window drop-down menus.
7. Click Add.

**What to do next**

Associate your catalog item with a service.

Associate a Catalog Item with a Service

Include your catalog items in the service. You can include a catalog item in only one service, but a service can contain many catalog items.

**Prerequisites**

Log in to the vRealize Automation console as a tenant administrator, business group manager, or service architect.

**Procedure**

1. Select Administration > Catalog Management > Catalog Items.
2. Select the catalog item to associate with a service.
3. Click Configure.
4. (Optional) Click Browse to change the icon of the catalog item.
5 Select the status of the catalog item from the **Status** drop-down menu.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Can be used in the service.</td>
</tr>
<tr>
<td>Inactive</td>
<td>Cannot be used in the service until you activate it.</td>
</tr>
</tbody>
</table>

6 Select a service with which to associate the catalog item from the **Service** drop-down menu.

7 Click **Update**.

You associated the catalog item with a service.

**What to do next**

Repeat the steps for all of the catalog items you want to include in the service. Create an entitlement to entitle the service and the catalog items included in it to users or groups in a business group. These users are the consumers of your service. See “Entitle a Service, Catalog Items, and Actions to a User or a Group of Users,” on page 42.

**Create an Approval Policy for Advanced Service Blueprints and Actions**

Tenant administrators and approval administrators can define approval policies and use them in entitlements. You can set up the approval policies with multiple levels for pre- and post-approval events.

For detailed information about approvals and approval levels, see *Tenant Administration*.

**Prerequisites**

- Log in to the vRealize Automation console as a **tenant administrator** or **approval administrator**.
- Verify that a service architect created and published an advanced service blueprint. See “Create a Service Blueprint,” on page 34 and “Publish a Service Blueprint as a Catalog Item,” on page 36.
- Verify that a service architect created and published a custom resource action. See “Create a Resource Action,” on page 36 and “Publish a Resource Action,” on page 38.

**Procedure**

1 **Specify Approval Policy Information** on page 40
   
   When you create an approval policy, first define the approval policy information, such as policy type, name, description, and status.

2 **Create an Approval Level** on page 41
   
   When you create an approval policy, you can add levels for both pre- and post-provisioning phases within the provisioning workflow.

You created an approval policy. You can use it when you create an entitlement.

**Specify Approval Policy Information**

When you create an approval policy, first define the approval policy information, such as policy type, name, description, and status.

**Procedure**

1 Select **Administration > Approval Policies**.

2 Click **Add (✚)**.
3 Select a policy type from the Policy Type drop-down menu.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Catalog - Catalog Item</td>
<td>Applies the policy to a catalog item that you created by publishing an advanced service blueprint.</td>
</tr>
<tr>
<td>Request - Advanced Service Blueprint</td>
<td></td>
</tr>
<tr>
<td>Service Catalog - Resource Action Request - your_custom_resource_action</td>
<td>Applies the policy to a resource action that a service architect created by using Advanced Service Designer. The resource action can be any custom resource action, including resource actions for managing IaaS-provisioned vSphere virtual machines.</td>
</tr>
</tbody>
</table>

4 Click OK.

5 Enter a name and, optionally, a description.

6 Select the state of the policy from the Status drop-down menu.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft</td>
<td>Saves the approval policy in an editable state.</td>
</tr>
<tr>
<td>Active</td>
<td>Saves the approval policy in a read-only state that you can use in an entitlement.</td>
</tr>
<tr>
<td>Inactive</td>
<td>Saves the approval policy in a read-only state that you cannot use in an entitlement until you activate the policy.</td>
</tr>
</tbody>
</table>

What to do next

Create the pre- and post-approval levels.

Create an Approval Level

When you create an approval policy, you can add levels for both pre- and post-provisioning phases within the provisioning workflow.

Procedure

1 On the Pre Approval or Post Approval tab, click the Add Levels icon ( ).

2 Enter a name and, optionally, a description.

3 Select a manual approval requirement.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Always Required</td>
<td>A manual approval is always required to complete the request.</td>
</tr>
<tr>
<td>Required based on conditions</td>
<td>A manual approval is required based on the conditions that you select from the Clause drop-down menu.</td>
</tr>
</tbody>
</table>

4 Select the approvers.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific Users and Groups</td>
<td>In the Search text box, enter the name of a user or group, and press Enter.</td>
</tr>
<tr>
<td>Determine approvers from the request</td>
<td>From the Value drop-down menu, select the approvers.</td>
</tr>
</tbody>
</table>
5 Indicate who must approve the request or action.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anyone can approve</td>
<td>Any one of the approvers can approve the request.</td>
</tr>
<tr>
<td>All must approve</td>
<td>Each of the approvers must approve the request.</td>
</tr>
</tbody>
</table>

6 Click the Approval Form tab.

7 Double-click the fields you want to make editable.

An approver can modify the editable fields when completing an approval for this level and policy.

Editing fields is only in the pre-provisioning phase. In the post-provisioning phase, the fields are displayed as read-only and cannot be edited.

8 Click Add.

9 Click Add.

You can apply your approval policy to services, catalog items, and actions when you create an entitlement.

**Entitle a Service, Catalog Items, and Actions to a User or a Group of Users**

Business group managers and tenant administrators can entitle services, catalog items, and resource actions to groups of users, or to a specific user in a selected business group. They can also assign an approval policy for each service, catalog item, and resource action. The users to whom an object is entitled are the users who can access and request the entitled objects.

**Prerequisites**

Log in to the vRealize Automation console as a tenant administrator or business group manager.

**Procedure**

1 Select Administration > Catalog Management > Entitlements.

2 Click Add (+).

3 Enter a name and, optionally, a description.

4 (Optional) Select the date and the time when the entitlement expires in the Expiration Date text boxes.

If you do not specify expiry date and time, the entitlement remains active indefinitely.

5 Select the status of the entitlement from the Status drop-down menu.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft</td>
<td>Saves the entitlement in draft state.</td>
</tr>
<tr>
<td>Active</td>
<td>The entitlement is active and can be considered when determining access to entities in the service catalog.</td>
</tr>
</tbody>
</table>

If you select to activate the entitlement, you must select users or groups.

6 Select a business group from the Business Group drop-down menu.

7 Enter the name of a user or group in the Users & Groups text box, and press Enter.

You can add more than one user or group from the selected business group to the list. The groups and users must be part of the business group you selected previously.

8 Click Next.
9  Enter the name of a service in the Entitled Services text box, and press Enter.
   You can select more than one service to add to an entitlement or you can add specific individual catalog
   items instead.

10 (Optional) Enter the name of a catalog item in the Entitled Catalog Items text box and press Enter.
   By default, all of the catalog items included in the service are entitled to the users or user groups you
   selected on the Details tab.
   If you entitle a service, by adding catalog items individually you can apply different approval policies
   to the catalog items.

11 (Optional) Enter the name of a resource action in the Entitled Actions text box, and press Enter.

12 (Optional) Modify the approval policies for a service, catalog item, or action.
   a  Point to the down arrow in the Approval Policy column and select Modify Policy.
   b  Select the policy to apply from the Apply this Policy drop-down menu.
   c  Click OK.

13 Click Add.

You created an entitlement and you can see it on the Entitlements page. If you set the entitlement in Active
status, the users and user groups that you specified can access the catalog items in the service.

When consumers of the service log in to their vRealize Automation consoles, they see the service on the
Catalog tab, and they can request the catalog items included in the service.

Extending Operations on Resources Provisioned by Other Sources

By using resource mappings, you can publish vRealize Orchestrator workflows as resource actions for
managing items provisioned by sources other than Advanced Service Designer, for example, by IaaS or
Application Services.

- Extending Operations Checklist on page 44
  Resource actions for items provisioned outside of Advanced Service Designer require a resource
  mapping as the input parameter. Because some resource mappings are included with your
  vRealize Automation installation, the steps required to extend operations on resources provisioned
  outside of Advanced Service Designer depend on the type of resource you want to extend.

- Resource Mapping Script Actions and Workflows on page 45
  You can use the provided resource mappings for vSphere virtual machines, vCloud Director virtual
  machines, and vCloud Director vApps, or you can create custom vRealize Orchestrator script actions
  or workflows to map additional vRealize Automation catalog resource types to vRealize Orchestrator
  inventory types.

- Create a Resource Mapping on page 45
  vRealize Automation provides resource mappings for vSphere virtual machines, vCloud Director
  virtual machines, and vCloud Director vApps. You can create additional resource mappings for other
  types of catalog resources.

- Create a Resource Action for an Item Provisioned by Another Source on page 46
  You can create resource actions to extend the operations that the consumers of the catalog items can
  perform on items provisioned by sources other than the Advanced Service Designer.

- Publish a Resource Action on page 48
  The newly created resource action is in draft state, and you must publish the resource action.
Assign an Icon to a Resource Action on page 48

After you create and publish a resource action, you can edit it and assign an icon to the action.

Extending Operations Checklist

Resource actions for items provisioned outside of Advanced Service Designer require a resource mapping as the input parameter. Because some resource mappings are included with your vRealize Automation installation, the steps required to extend operations on resources provisioned outside of Advanced Service Designer depend on the type of resource you want to extend.

Extending Operations for vSphere and vCloud Director Machines

vRealize Automation provides the resource mappings for vSphere virtual machines, vCloud Director virtual machines, and vCloud Director vApps.

Table 4-1. Extending Operations for vSphere and vCloud Director Machines Checklist

<table>
<thead>
<tr>
<th>Task</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a resource action with a resource mapping as the input parameter. For the input parameter, select one of the provided resource mappings for vSphere virtual machines, vCloud Director virtual machines, or vCloud Director vApps.</td>
<td>“Create a Resource Action for an Item Provisioned by Another Source,” on page 46.</td>
</tr>
<tr>
<td>Optionally, you can assign an icon to your resource action.</td>
<td>“Assign an Icon to a Resource Action,” on page 38.</td>
</tr>
</tbody>
</table>

Tenant administrators and business group managers can entitle users to your resource action. Entitled users see the resource action in the Actions drop-down menu on the Items tab.

Extending Operations for Other Resource Types

You can extend operations on any vRealize Automation catalog resource by creating your own custom resource mappings.

Table 4-2. Extending Operations for Other Resource Types Checklist

<table>
<thead>
<tr>
<th>Task</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Develop or import a vRealize Orchestrator resource mapping script action or workflow.</td>
<td>For information about creating workflows and script actions to map resources, see “Resource Mapping Script Actions and Workflows,” on page 45. For information about developing with vRealize Orchestrator, see Developing with VMware vCenter Orchestrator.</td>
</tr>
<tr>
<td>Create an Advanced Service Designer resource mapping that runs your custom vRealize Orchestrator resource mapping script action or workflow.</td>
<td>“Create a Resource Mapping,” on page 45.</td>
</tr>
<tr>
<td>Create a resource action with a resource mapping as the input parameter. For the input parameter, select your custom resource mapping.</td>
<td>“Create a Resource Action for an Item Provisioned by Another Source,” on page 46.</td>
</tr>
<tr>
<td>Optionally, you can assign an icon to your resource action.</td>
<td>“Assign an Icon to a Resource Action,” on page 38.</td>
</tr>
</tbody>
</table>

Tenant administrators and business group managers can entitle users to your resource action. Entitled users see the resource action in the Actions drop-down menu on the Items tab.
Resource Mapping Script Actions and Workflows

You can use the provided resource mappings for vSphere virtual machines, vCloud Director virtual machines, and vCloud Director vApps, or you can create custom vRealize Orchestrator script actions or workflows to map additional vRealize Automation catalog resource types to vRealize Orchestrator inventory types.

Depending on your version of vRealize Orchestrator, you can create either a vRealize Orchestrator workflow or a script action to map resources between vRealize Orchestrator and vRealize Automation.

To develop the resource mapping, you use an input parameter of type Properties, which contains a key-value pair defining the provisioned resource, and an output parameter of a vRealize Orchestrator inventory type expected by the corresponding vRealize Orchestrator plug-in. The properties available for the mapping depend on the type of resource. For example, the EXTERNALREFERENCE_ID property is a common key parameter that defines individual machines, and you can use this property to query a catalog resource. If you are creating a mapping for a resource that does not use an EXTERNALREFERENCE_ID, you can use one of the other properties that are passed for the individual machines such as name, description, and so on.

vRealize Automation includes vRealize Orchestrator resource mapping script actions and resource mapping workflows for each of the provided Advanced Service Designer resource mappings. Script actions for the provided resource mappings are located in the com.vmware.vcac.asd.mappings package of the embedded vRealize Orchestrator server. Workflows for the provided resource mappings are located in the Orchestrator > vCAC > ASD > Resource Mappings workflows folder of the embedded vRealize Orchestrator server.

For more information about developing workflows and script actions, see Developing with VMware vCenter Orchestrator.

Create a Resource Mapping

vRealize Automation provides resource mappings for vSphere virtual machines, vCloud Director virtual machines, and vCloud Director vApps. You can create additional resource mappings for other types of catalog resources.

Prerequisites

- Log in to the vRealize Automation console as a service architect.
- Develop or import a vRealize Orchestrator resource mapping script action or workflow.

Procedure

2. Click Add (+).
3. Enter a name and, optionally, a description.
4. Enter the type of the catalog resource in the Catalog Resource Type text box and press enter. The type of catalog resource appears on the details view of the provisioned item.
5. Enter the vRealize Orchestrator object type in the Orchestrator Type text box and press enter. This is the output parameter of the resource mapping workflow.
6 (Optional) Add target criteria to restrict the availability of resource actions created by using this resource mapping.

Resource actions are also subject to restrictions based on approvals and entitlements.

a Select **Available based on conditions**.

b Select the type of condition.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>All of the following</td>
<td>If all of the clauses you define are satisfied, resource actions created by using this resource mapping are available to the user.</td>
</tr>
<tr>
<td>Any of the following</td>
<td>If any one of the clauses you define are satisfied, resource actions created by using this resource mapping are available to the user.</td>
</tr>
<tr>
<td>Not the following</td>
<td>If the clause you define exists, resource actions created by using this resource mapping are not available.</td>
</tr>
</tbody>
</table>

c Follow the prompts to build your clauses and complete the condition.

7 Select the type of resource mapping to use.

8 Select your resource mapping script action or workflow from the vRealize Orchestrator library.

9 Click **Add**.

### Create a Resource Action for an Item Provisioned by Another Source

You can create resource actions to extend the operations that the consumers of the catalog items can perform on items provisioned by sources other than the Advanced Service Designer.

By creating a resource action, you publish a vRealize Orchestrator workflow as a post-provisioning operation. During this process, you can edit the default submission and read-only forms. See “Designing a Resource Action Form,” on page 64.

**Prerequisites**

- Log in to the vRealize Automation console as a **service architect**.
- Create a resource mapping corresponding to the input parameter of the resource action. See “Create a Resource Mapping,” on page 45

**Procedure**

1 Select **Advanced Services > Resource Actions**.

2 Click **Add** (⊕).

3 Navigate through the vRealize Orchestrator workflow library and select a workflow.

   You can see the name and description of the selected workflow, and the input and output parameters as they are defined in vRealize Orchestrator.

4 Click **Next**.

5 Select the resource mapping that you previously created from the **Resource type** drop-down menu.

6 Select the input parameter for the resource action from the **Input parameter** drop-down menu.

7 Click **Next**.

8 Enter a name and, optionally, a description.

   The **Name** and **Description** text boxes are prepopulated with the name and description of the workflow as they are defined in vRealize Orchestrator.
9 (Optional) If you do not want to prompt consumers to enter a description and reason for requesting this resource action, select the Hide catalog request information page check box.

10 (Optional) Select the type of the action.

Depending on the action workflow, you can select one, both, or none of the options.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disposing</td>
<td>The input parameter of the resource action workflow is disposed and the item is removed from the Items tab. For example, the resource action is for deleting a provisioned machine.</td>
</tr>
<tr>
<td>Provisioning</td>
<td>The resource action is for provisioning. For example, the resource action is for copying a catalog item. Select an output parameter from the drop-down menu. You can select a custom resource or resource mapping that you previously created so that when the consumers request this resource action, the provisioned items are added on the Items tab. If you have only the No provisioning option, either the resource action is not for provisioning, or you did not create a proper custom resource or resource mapping for the output parameter, and you cannot proceed.</td>
</tr>
</tbody>
</table>

11 Select the conditions under which the resource action is available to users, and click Next.

12 (Optional) Edit the form of the resource action on the Form tab.

The form of the resource action maps the vRealize Orchestrator workflow presentation. You can change the form by deleting, editing, and rearranging the elements. You can also add a new form and form pages and drag the necessary elements to the new form and form page.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a form</td>
<td>Click the New Form icon ( ) next to the form name, provide the required information, and click Submit.</td>
</tr>
<tr>
<td>Edit a form</td>
<td>Click the Edit icon ( ) next to the form name, make the necessary changes, and click Submit.</td>
</tr>
<tr>
<td>Regenerate the workflow presentation</td>
<td>Click the Rebuild icon ( ) next to the form name and click OK.</td>
</tr>
<tr>
<td>Delete a form</td>
<td>Click the Delete icon ( ) next to the form name, and in the confirmation dialog box click OK.</td>
</tr>
<tr>
<td>Add a form page</td>
<td>Click the New Page icon ( ) next to the form page name, provide the required information, and click Submit.</td>
</tr>
<tr>
<td>Edit a form page</td>
<td>Click the Edit icon ( ) next to the form page name, make the necessary changes, and click Submit.</td>
</tr>
<tr>
<td>Delete a form page</td>
<td>Click the Delete icon ( ) next to the form page name, and in the confirmation dialog box click OK.</td>
</tr>
<tr>
<td>Add an element to the form page</td>
<td>Drag an element from the New Fields pane on the left to the pane on the right. You can then provide the required information and click Submit.</td>
</tr>
<tr>
<td>Edit an element</td>
<td>Click the Edit icon ( ) next to the element to edit, make the necessary changes, and click Submit.</td>
</tr>
<tr>
<td>Delete an element</td>
<td>Click the Delete icon ( ) next to the element to delete, and in the confirmation dialog box click OK.</td>
</tr>
</tbody>
</table>

13 Click Add.

You created a resource action and you can see it listed on the Resource Actions page.
What to do next
Publish the resource action.

Publish a Resource Action
The newly created resource action is in draft state, and you must publish the resource action.

Prerequisites
Log in to the vRealize Automation console as a service architect.

Procedure
1 Select Advanced Services > Resource Actions.
2 Select the row of the resource action to publish, and click Publish.

The status of the resource action changes to Published.

What to do next
Assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38. Business group managers and tenant administrators can then use the action when they create an entitlement.

Assign an Icon to a Resource Action
After you create and publish a resource action, you can edit it and assign an icon to the action.

Prerequisites
Log in to the vRealize Automation console as a service architect.

Procedure
1 Select Administration > Catalog Management > Actions.
2 Select the resource action that you created.
3 Click Configure.
4 Click Browse and select the icon to add.
5 Click Open.
6 Click Update.

You assigned an icon to the resource action. Business group managers and tenant administrators can use the resource action in an entitlement.
Importing and Exporting Advanced Service Designer Components

Instead of recreating Advanced Service Designer components for every environment, you can export your custom resources, service blueprints, resource mappings, and resource actions from one environment and import them into another.

Export Advanced Service Designer Components

You can export a zip file of your custom resources, service blueprints, resource mappings, and resource actions so that you can reuse Advanced Service Designer components in separate tenants or different vRealize Automation environments.

vRealize Orchestrator workflows are not exported with Advanced Service Designer components. You can use vRealize Orchestrator to create a .package for import into the target vRealize Automation environment.

Prerequisites

Log in to the vRealize Automation console as a service architect.

Procedure

1. To configure the your content export options, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Content Management &gt; Export Content</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Export Content</td>
</tr>
</tbody>
</table>

2. Select the components to export.

3. Click Next.
   
   You are directed to the next tab to complete your selections.

4. From the Export Content tab, click Export Content to begin downloading your selections.

All Advanced Service Designer components you selected are downloaded in a zip file.

What to do next

You can import your content into another vRealize Automation environment.

Import Advanced Service Designer Components and Workflows

You can import custom vRealize Orchestrator workflows, custom resources, service blueprints, resource mappings, and resource actions so that you can use previously created Advanced Service Designer components in another tenant or in different vRealize Automation environments.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- Create a .package file to import vRealize Orchestrator workflows.
- Create a .zip bundle to import vRealize Automation components.
Procedure

1. To configure the your content import options, browse to the location based on your current version.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize Automation 6.2.1</td>
<td>Select Administration &gt; Content Management &gt; Import Content</td>
</tr>
<tr>
<td>vRealize Automation 6.2</td>
<td>Select Administration &gt; Advanced Services &gt; Import Content</td>
</tr>
</tbody>
</table>

2. Configure a prefix to prevent conflicts in cases where an Advanced Service Designer component with the same name already exists.

   a. (Optional) Select **Prefix only conflicting** to add a prefix only in cases where a naming conflict exists.

   b. Enter a prefix to add to imported components.

3. Click **Browse** and select the files to import.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.package</td>
<td>Import workflows into your vRealize Orchestrator.</td>
</tr>
<tr>
<td>.zip</td>
<td>Import custom resources, service blueprints, resource mappings, and resource actions into your Advanced Service Designer.</td>
</tr>
</tbody>
</table>

4. Click **Open**.

5. (Optional) Click **Validate** to ensure you are not missing any vRealize Orchestrator workflows required by the Advanced Service Designer components.

   vRealize Orchestrator workflow packages are not validated and might return an error.

6. Click **Import Content**.

After you import Advanced Service Designer components, you can see or edit them in Advanced Service Designer. After you import vRealize Orchestrator workflows you can see or edit them in vRealize Orchestrator, and you can use them to create components in Advanced Service Designer.
The Advanced Service Designer includes a form designer that you can use to design submission and details forms for blueprints, resources, and actions. Based on the presentation of the workflows, the form designer dynamically generates default forms and fields you can use to modify the default forms.

You can create interactive forms that the users can complete for submission of catalog items and resource actions. You can also create read-only forms that define what information the users can see on the details view for a catalog item or a provisioned resource.

As you create objects in the Advanced Service Designer, forms are generated for common use cases.

Table 5-1. Advanced Service Designer Object Types and Associated Forms

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Default Form</th>
<th>Additional Forms</th>
</tr>
</thead>
</table>
| Service blueprint | Request submission form based on the presentation of the selected workflow. | ■ Catalog item details (read-only)  
■ Submitted request details (read-only) |
| Custom resource   | Resource details form based on the attributes of the vRealize Orchestrator plug-in inventory type (read-only). | ■ None |
| Resource action   | Action submission form based on the presentation of the selected workflow. | ■ Submitted action details (read-only) |

You can modify the default forms and design new forms. You can drag fields to add and reorder them on the form. You can place constraints on the values of certain fields, specify default values, or provide instructional text for the end user who is completing the form.

Because of their different purposes, the operations you can perform to design read-only forms are limited compared to the operations for designing submission forms.

This chapter includes the following topics:

■ “Fields in the Form Designer,” on page 52
■ “Constraints and Values in the Form Designer,” on page 53
■ “External Value Definitions in the Form Designer,” on page 55
■ “Working With the Form Designer,” on page 55
Fields in the Form Designer

You can extend the workflow presentation and functionality by adding new predefined fields to the default generated forms of resource actions and service blueprints.

If an input parameter is defined in the vRealize Orchestrator workflow, in vRealize Automation it appears on the default generated form. If you do not want to use the default generated fields in the form, you can delete them and drag and drop new fields from the palette. You can replace default generated fields without breaking the workflow mappings if you use the same ID as the field you are replacing.

You can also add new fields, other than the ones that were generated based on the vRealize Orchestrator workflow inputs, so that you can extend the workflow presentation and functionality in the following cases:

- Add constraints to the existing fields
  
  For example, you can create a new drop-down menu and name it `dd`. You can also create predefined options of Gold, Silver, Bronze, and Custom. If there is a predefined field, such as CPU, you can add the following constraints to this field:

  - If `dd` equals Gold, then CPU is 2000 MHz
  - If `dd` equals Silver, then CPU is 1000 MHz
  - If `dd` equals Bronze then CPU is 500 MHz
  - If `dd` equals Custom, the CPU field is editable, and the consumer can specify a custom value

- Add external value definitions to fields
  
  You can add an external value definition to a field so that you can run vRealize Orchestrator script actions and supply additional information to consumers on the forms you design. For instance, you might want to create a workflow to change the firewall settings of a virtual machine. On the resource action request page, you want to provide the user with the ability to change the open port settings, but you also want to restrict the options to ports that are open. You can add an external value definition to a dual list field and select a custom vRealize Orchestrator script action that queries for open ports. When the request form loads, the script actions runs, and the open ports are presented as options to the user.

- Add new fields that are handled in the vRealize Orchestrator workflow as global parameters
  
  For instance, the workflow provides an integration with a third-party system and the workflow developer defined input parameters to be handled in the general case, but has also provided a way for passing custom fields. For example, in a scripting box, all global parameters that start with `my3rdparty` are handled. Then, if the service architect wants to pass specific values for consumers to provide, the service architect can add a new field named `my3rdparty_CPU`.

### Table 5.2. New Fields in the Resource Action or Service Blueprint Form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text field</td>
<td>Single-line text box</td>
</tr>
<tr>
<td>Text area</td>
<td>Multi-line text box</td>
</tr>
<tr>
<td>Link</td>
<td>Field in which consumers enter a URL</td>
</tr>
<tr>
<td>Email</td>
<td>Field in which consumers enter an email address</td>
</tr>
<tr>
<td>Password field</td>
<td>Field in which consumers enter a password</td>
</tr>
<tr>
<td>Integer field</td>
<td>Text box in which consumers enter an integer</td>
</tr>
<tr>
<td></td>
<td>You can make this field a slider with a minimum and maximum value, as well as an increment.</td>
</tr>
<tr>
<td>Decimal field</td>
<td>Text box in which consumers enter a decimal</td>
</tr>
<tr>
<td></td>
<td>You can make this field a slider with a minimum and maximum value, as well as an increment.</td>
</tr>
</tbody>
</table>
### Table 5-2. New Fields in the Resource Action or Service Blueprint Form (Continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date &amp; time</td>
<td>Text boxes in which consumers specify a date (by selecting a date from a calendar menu) and can also select the time (by using up and down arrows)</td>
</tr>
<tr>
<td>Dual List</td>
<td>A list builder in which consumers move a predefined set of values between two lists, the first list contains all unselected options and the second list contains the user's choices.</td>
</tr>
<tr>
<td>Check box</td>
<td>Check box</td>
</tr>
<tr>
<td>Yes/No</td>
<td>Drop-down menu for selecting Yes or No</td>
</tr>
<tr>
<td>Drop-down</td>
<td>Drop-down menu</td>
</tr>
<tr>
<td>List</td>
<td>List</td>
</tr>
<tr>
<td>Check box list</td>
<td>Check box list</td>
</tr>
<tr>
<td>Radio button group</td>
<td>Group of radio buttons</td>
</tr>
<tr>
<td>Search</td>
<td>Search text box that auto completes the query and where consumers select an object</td>
</tr>
<tr>
<td>Tree</td>
<td>Tree that consumers use to browse and select available objects</td>
</tr>
<tr>
<td>Map</td>
<td>Map table that consumers use to define key-value pairs for properties</td>
</tr>
</tbody>
</table>

You can also use the **Section header** form field to split form pages in sections with separate headings and the **Text** form field to add read-only informational texts.

### Constraints and Values in the Form Designer

When you edit an element of the blueprint or resource action form, you can apply various constraints and values to the element.

#### Constraints

The constraints that you can apply to an element vary depending on the type of element you are editing or adding to the form. Some constraint values might be configured in the vRealize Orchestrator workflow. Those values do not appear on the Constraints tab because they are often dependent on conditions that are evaluated when the workflow runs. Any constraint values that you configure for the blueprint form overrides any constraints specified in the vRealize Orchestrator workflow.

For each constraint you apply to an element, you can select one of the following options to define the constraint:

- **Not set**: Gets the property from the vRealize Orchestrator workflow presentation.
- **Constant**: Sets the element you are editing to required or optional.
- **Field**: Binds the element to another element from the form. For example, you can set the element to be required only when another element, such as a check box, is selected.
- **Conditional**: Applies a condition. By using conditions you can create various clauses and expressions and apply them to the state or constraints of the element.
- **External**: Select a vRealize Orchestrator script action to define the value.
### Table 5-3. Constraints in the forms designer

<table>
<thead>
<tr>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required</td>
<td>Indicates whether the element is required.</td>
</tr>
<tr>
<td>Read only</td>
<td>Indicates whether the field is read-only.</td>
</tr>
<tr>
<td>Value</td>
<td>Allows you to set a value for the element.</td>
</tr>
<tr>
<td>Visible</td>
<td>Indicates whether the consumer can see the element.</td>
</tr>
<tr>
<td>Minimum length</td>
<td>Allows you to set a minimum number of characters of the string input element.</td>
</tr>
<tr>
<td>Maximum length</td>
<td>Allows you to set a maximum allowed number of characters of the string input element.</td>
</tr>
<tr>
<td>Minimum value</td>
<td>Allows you to set a minimum value of the number input element.</td>
</tr>
<tr>
<td>Maximum value</td>
<td>Allows you to set a maximum value of the number input element.</td>
</tr>
<tr>
<td>Increment</td>
<td>Allows you to set an increment for an element such as a <strong>Decimal</strong> or <strong>Integer</strong> field. For example, when you want an <strong>Integer</strong> field to be rendered as a <strong>Slider</strong>, you can use the value of the step.</td>
</tr>
<tr>
<td>Minimum count</td>
<td>Allows you to set a minimum count of items of the element that can be selected. For example, when you add or edit a <strong>Check box list</strong> you can set the minimum number of check boxes that the consumer must select to proceed.</td>
</tr>
<tr>
<td>Maximum count</td>
<td>Allows you to set a maximum count of items of the element that can be selected. For example, when you add or edit a <strong>Check box list</strong> you can set the maximum number of check boxes that the consumer must select to proceed.</td>
</tr>
</tbody>
</table>

### Values

You can apply values to some of the elements and define what the consumers see for some of the fields. The options available depend on the type of element you are editing or adding to the form.

### Table 5-4. Values in the Form Designer

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not set</td>
<td>Get the value of the element you are editing from the vRealize Orchestrator workflow presentation.</td>
</tr>
<tr>
<td>Predefined values</td>
<td>Select values from a list of related objects from the vRealize Orchestrator inventory.</td>
</tr>
<tr>
<td>Value</td>
<td>Define a static custom values with labels.</td>
</tr>
<tr>
<td>External Values</td>
<td>Select a vRealize Orchestrator script action to define your value with information not directly exposed by the workflow.</td>
</tr>
</tbody>
</table>
External Value Definitions in the Form Designer

When you edit some elements in the forms designer, you can assign external value definitions that use custom vRealize Orchestrator script actions to supply information not directly exposed by the workflow.

For instance, you might want to publish a resource action to install software on a provisioned machine. Instead of providing the consumer with a static list of all software available for download, you can dynamically populate that list with software that is relevant for the machine's operating system, software that the user has not previously installed on the machine, or software that is out of date on the machine and requires an update.

To provide custom dynamic content for your consumer, you create a vRealize Orchestrator script action that retrieves the information you want to display to your consumers. You assign your script action to a field in the form designer as an external value definition. When the resource or service blueprint form is presented to your consumers, the script action retrieves your custom information and displays it to your consumer.

You can use external value definitions to supply default or read-only values, to build boolean expressions, to define constraints, or to provide options for consumers to select from lists, check boxes, and so on.

Working With the Form Designer

When you create advanced service blueprints, custom resource actions, and custom resources, you can edit the forms of the blueprints, actions, and resources by using the form designer. You can edit the representation and define what the consumers of the item or action see when they request the catalog item or run the post-provisioning operation.

By default, any service blueprint, resource action, or custom resource form is generated based on the workflow presentation in vRealize Orchestrator.
The steps in the vRealize Orchestrator presentation are represented as form pages and the vRealize Orchestrator presentation groups are represented as separate sections. The input types of the selected workflow are displayed as various fields in the form. For example, the vRealize Orchestrator type string is represented by a text box. A complex type such as VC:VirtualMachine is represented by a search box or a tree, so that the consumers can type an alphanumeric value to search for a virtual machine or browse to select a virtual machine.

Add Blueprint

You can edit how an object is represented in the form designer. For example, you can edit the default VC:VirtualMachine representation and make it a tree instead of a search box. You can also add new fields such as check boxes, drop-down menus, and so on, and apply various constraints. If the new fields you add are not valid or are not correctly mapped to the vRealize Orchestrator workflow inputs, when the consumer runs the workflow, vRealize Orchestrator skips the invalid or unmapped fields.

Designing a Custom Resource Form

All fields on the resource details form are displayed as read-only to the consumer on the item details page when they provision your custom resource. You can perform basic edit operations to the form, such as deleting, modifying, or rearranging fields, or you can add new externally defined fields that use vRealize Orchestrator script actions to supply additional read-only information to consumers.

- **Edit a Custom Resource Element** on page 57
  You can edit some of the characteristics of an element on the custom resource Details Form page. Each default field on the page represents a property of the custom resource. You cannot change the type of a property or the default values, but you can edit the name, size, description.

- **Add a New Custom Resource Form Page** on page 57
  You can add a new page to rearrange the form into multiple tabs.

- **Insert a Section Header in a Custom Resource Form** on page 58
  You can insert a section header to split the form into sections.

- **Insert a Text Element in a Custom Resource Form** on page 58
  You can insert a text box to add some descriptive text to the form.

- **Insert an Externally Defined Field in a Custom Resource Form** on page 58
  You can insert a new field and assign it an external value definition to dynamically provide read-only information that consumers can see on the item details page when they provision a custom resource.
Edit a Custom Resource Element

You can edit some of the characteristics of an element on the custom resource Details Form page. Each default field on the page represents a property of the custom resource. You cannot change the type of a property or the default values, but you can edit the name, size, description.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Custom Resource,” on page 33.

Procedure

1. Select Advanced Services > Custom Resources.
2. Click the custom resource to edit.
3. Click the Details Form tab.
4. Point to the element you want to edit and click the Edit icon.
5. Enter a new name for the field in the Label text box to change the label.
6. Edit the description in the Description text box.
7. Select an option from the Size drop-down menu to change the size of the element.
8. Select an option from the Label size drop-down menu to change the size of the label.
9. Click Submit.
10. Click Update.

Add a New Custom Resource Form Page

You can add a new page to rearrange the form into multiple tabs.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Custom Resource,” on page 33.

Procedure

1. Select Advanced Services > Custom Resources.
2. Click the custom resource to edit.
3. Click the Details Form tab.
4. Click the New Page icon ( ) next to the form page name.
5. Enter a name for the page in the Heading text box.
6. Click Submit.
7. Click Update.

You can delete some of the elements from the original form page and insert them in the new form page, or you can add new fields that use external value definitions to provide information to consumers that is not directly exposed by the vRealize Orchestrator workflow.
Insert a Section Header in a Custom Resource Form

You can insert a section header to split the form into sections.

Prerequisites
- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Custom Resource,” on page 33.

Procedure
1. Select Advanced Services > Custom Resources.
2. Click the custom resource to edit.
3. Click the Details Form tab.
4. Drag the Section header element from the Form pane to the Form page pane.
5. Type a name for the section.
6. Click outside of the element to save the changes.
7. Click Update.

Insert a Text Element in a Custom Resource Form

You can insert a text box to add some descriptive text to the form.

Prerequisites
- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Custom Resource,” on page 33.

Procedure
1. Select Advanced Services > Custom Resources.
2. Click the custom resource to edit.
3. Click the Details Form tab.
4. Drag the Text element from the Form pane to the Form page pane.
5. Enter the text you want to add.
6. Click outside of the element to save the changes.
7. Click Update.

Insert an Externally Defined Field in a Custom Resource Form

You can insert a new field and assign it an external value definition to dynamically provide read-only information that consumers can see on the item details page when they provision a custom resource.

Prerequisites
- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Custom Resource,” on page 33.
- Develop or import a vRealize Orchestrator script action to retrieve the information you want to provide to consumers.
Procedure

1. Select **Advanced Services > Custom Resources**.
2. Click the custom resource to edit.
3. Click the **Details Form** tab.
4. Drag an element from the New Fields pane and drop it to the Form page pane.
5. Enter an ID for the element in the **ID** text box.
6. Enter a label in the **Label** text box.
   - Labels appear to consumers on the forms.
7. (Optional) Select a type for the field from the **Type** drop-down menu.
8. Enter the result type of your vRealize Orchestrator script action in the **Entity Type** search box and press Enter.
   - For example, if you want to use a script action to display the current user, and the script returns a vRealize Orchestrator result type of `LdapUser`, enter `LdapUser` in the **Entity Type** search box and press Enter.
9. Click **Add External Value**.
10. Select your custom vRealize Orchestrator script action.
11. Click **Submit**.
12. Click **Submit** again.
13. Click **Update**.

   When the form is presented to your consumers, the script action retrieves your custom information and displays it to your consumer.

**Designing a Service Blueprint Form**

When you create a service blueprint, you can edit the form of the blueprint by adding new fields to the form, modifying the existing fields, deleting, or rearranging fields. You can also create new forms and form pages, and drag and drop new fields to them.

- **Add a New Service Blueprint Form** on page 60
  When you edit the default generated form of a workflow that you want to publish as a service blueprint, you can add a new service blueprint form.
- **Edit a Service Blueprint Element** on page 60
  You can edit some of the characteristics of an element on the Blueprint Form page of a service blueprint. You can change the type of an element, its default values, and apply various constraints and values.
- **Add a New Element** on page 62
  When you edit the default generated form of a service blueprint, you can add a predefined new element to the form. For example, if you do not want to use a default generated field, you can delete it and replace it with a new one.
- **Insert a Section Header in a Service Blueprint Form** on page 63
  You can insert a section header to split the form into sections.
- **Add a Text Element to a Service Blueprint Form** on page 63
  You can insert a text box to add some descriptive text to the form.
Add a New Service Blueprint Form

When you edit the default generated form of a workflow that you want to publish as a service blueprint, you can add a new service blueprint form.

By adding a new service blueprint form, you define the look and feel of the catalog item details and submitted request details pages. If you do not add a catalog item details and submitted request details forms, the consumer sees what is defined in the request form.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Service Blueprint,” on page 34.

Procedure

1. Select Advanced Services > Service Blueprints.
2. Click the service blueprint you want to edit.
3. Click the Blueprint Form tab.
4. Click the New Form icon (Φ).
5. Enter a name and, optionally, a description.
6. Select the screen type from the Screen type menu.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog item details</td>
<td>A catalog item details page that consumers see when they click a catalog item.</td>
</tr>
<tr>
<td>Request form</td>
<td>The default service blueprint form. The consumers see the request form when they request the catalog item.</td>
</tr>
<tr>
<td>Submitted request details</td>
<td>A request details page that consumers see after they request the item and want to view the request details on the Request tab.</td>
</tr>
</tbody>
</table>

7. Click Submit.

What to do next

Add the fields you want by dragging them from the New fields pane to the Form page pane.

Edit a Service Blueprint Element

You can edit some of the characteristics of an element on the Blueprint Form page of a service blueprint. You can change the type of an element, its default values, and apply various constraints and values.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Service Blueprint,” on page 34.

Procedure

1. Select Advanced Services > Service Blueprints.
2. Click the service blueprint you want to edit.
3. Click the Blueprint Form tab.
4. Locate the element you want to edit.
5 Click the **Edit** icon (📝).

6 Enter a new name for the field in the **Label** text box to change the label that consumers see.

7 Edit the description in the **Description** text box.

8 Select an option from the **Type** drop-down menu to change the display type of the element.
   The options vary depending on the type of element you edit.

9 Select an option from the **Size** drop-down menu to change the size of the element.

10 Select an option from the **Label size** drop-down menu to change the size of the label.

11 Edit the default value of the element.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not set</td>
<td>Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.</td>
</tr>
<tr>
<td>Constant</td>
<td>Sets the default value of the element you are editing to a constant value that you specify.</td>
</tr>
<tr>
<td>Field</td>
<td>Binds the default value of the element to a parameter of another element from the representation.</td>
</tr>
<tr>
<td>Conditional</td>
<td>Applies a condition. By using conditions you can create various clauses and expressions and apply them to an element.</td>
</tr>
<tr>
<td>External</td>
<td>Select a vRealize Orchestrator script action to define the value.</td>
</tr>
</tbody>
</table>

12 Apply constraints to the element on the **Constraints** tab.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not set</td>
<td>Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.</td>
</tr>
<tr>
<td>Constant</td>
<td>Sets the default value of the element you are editing to a constant value that you specify.</td>
</tr>
<tr>
<td>Field</td>
<td>Binds the default value of the element to a parameter of another element from the representation.</td>
</tr>
<tr>
<td>Conditional</td>
<td>Applies a condition. By using conditions you can create various clauses and expressions and apply them to an element.</td>
</tr>
<tr>
<td>External</td>
<td>Select a vRealize Orchestrator script action to define the value.</td>
</tr>
</tbody>
</table>

13 Add one or more values for the element on the **Values** tab.

The options available depend on the type of element you are editing.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not set</td>
<td>Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.</td>
</tr>
</tbody>
</table>
| Predefined values | Select values from a list of related objects from the vRealize Orchestrator inventory.  
  a Enter a value in the **Predefined values** search box to search the vRealize Orchestrator inventory.  
  b Select a value from the search results and press Enter. |
14 Click Submit.
15 Click Update.

**Add a New Element**

When you edit the default generated form of a service blueprint, you can add a predefined new element to the form. For example, if you do not want to use a default generated field, you can delete it and replace it with a new one.

**Prerequisites**

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Service Blueprint,” on page 34.

**Procedure**

1. Select Advanced Services > Service Blueprints.
2. Click the service blueprint you want to edit.
3. Click the Blueprint Form tab.
4. Drag an element from the New Fields pane and drop it to the Form page pane.
5. Enter the ID of a workflow input parameter in the ID text box.
6. Enter a label in the Label text box.
   - Labels appear to consumers on the forms.
7. (Optional) Select a type for the field from the Type drop-down menu.
8. Enter a vRealize Orchestrator object in the Entity type text box and press Enter.
   - This step is not required for all field types.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Result Type</td>
<td>If you are using a script action to define an external value for the field, enter the result type of your vRealize Orchestrator script action.</td>
</tr>
<tr>
<td>Input Parameter</td>
<td>If you are using the field to accept consumer input and pass parameters back to vRealize Orchestrator, enter the type for the input parameter accepted by the vRealize Orchestrator workflow.</td>
</tr>
<tr>
<td>Output Parameter</td>
<td>If you are using the field to display information to consumers, enter the type for the output parameter of the vRealize Orchestrator workflow.</td>
</tr>
</tbody>
</table>

9. (Optional) Select the Multiple values check box to allow consumers to select more than one object. This option is not available for all field types.
10 Click Submit.
11 Click Update.

What to do next
You can edit the element to change the default settings and apply various constraints or values.

Insert a Section Header in a Service Blueprint Form
You can insert a section header to split the form into sections.

Prerequisites
- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Service Blueprint,” on page 34.

Procedure
1 Select Advanced Services > Service Blueprints.
2 Click the service blueprint you want to edit.
3 Click the Blueprint Form tab.
4 Drag the Section header element from the Form pane to the Form page pane.
5 Type a name for the section.
6 Click outside of the element to save the changes.
7 Click Update.

Add a Text Element to a Service Blueprint Form
You can insert a text box to add some descriptive text to the form.

Prerequisites
- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Service Blueprint,” on page 34.

Procedure
1 Select Advanced Services > Service Blueprints.
2 Click the service blueprint you want to edit.
3 Click the Blueprint Form tab.
4 Drag the Text element from the New Fields pane to the Form page pane.
5 Enter the text you want to add.
6 Click outside of the element to save the changes.
7 Click Update.
Designing a Resource Action Form

When you create a resource action, you can edit the form of the action by adding new fields to the form, modifying the existing fields, deleting, or rearranging fields. You can also create new forms and form pages, and drag and drop new fields to them.

Add a New Resource Action Form

When you edit the default generated form of a workflow you want to publish as a resource action, you can add a new resource action form.

By adding a new resource action form, you define how the submitted action details page looks. If you do not add a submitted action details form, the consumer sees what is defined in the action form.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Resource Action,” on page 36.

Procedure

2. Click the resource action you want to edit.
3. Click the Form tab.
4. Click the New Form icon (➕).
5. Enter a name and, optionally, a description.
6. Select the screen type from the Screen type menu.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action form</td>
<td>The default resource action form that consumers see when they decide to run the post-provisioning action.</td>
</tr>
<tr>
<td>Submitted action details</td>
<td>A request details page that consumers see when they request the action and decide to view the request details on the Request tab.</td>
</tr>
</tbody>
</table>

7. Click Submit.

What to do next

Add the fields you want by dragging them from the New fields pane to the Form page pane.

Add a New Element to a Resource Action Form

When you edit the default generated form of a resource action, you can add a predefined new element to the form. For example, if you do not want to use a default generated field, you can delete it and replace it with a new one.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Resource Action,” on page 36.

Procedure

2. Click the resource action you want to edit.
3 Click the **Form** tab.

4 Drag an element from the New Fields pane and drop it to the Form page pane.

5 Enter the ID of a workflow input parameter in the **ID** text box.

6 Enter a label in the **Label** text box.

   Labels appear to consumers on the forms.

7 (Optional) Select a type for the field from the **Type** drop-down menu.

8 Enter a vRealize Orchestrator object in the **Entity type** text box and press Enter.

   This step is not required for all field types.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Result Type</strong></td>
<td>If you are using a script action to define an external value for the field,</td>
</tr>
<tr>
<td></td>
<td>enter the result type of your vRealize Orchestrator script action.</td>
</tr>
<tr>
<td><strong>Input Parameter</strong></td>
<td>If you are using the field to accept consumer input and pass parameters</td>
</tr>
<tr>
<td></td>
<td>back to vRealize Orchestrator, enter the type for the input parameter</td>
</tr>
<tr>
<td></td>
<td>accepted by the vRealize Orchestrator workflow.</td>
</tr>
<tr>
<td><strong>Output Parameter</strong></td>
<td>If you are using the field to display information to consumers, enter the</td>
</tr>
<tr>
<td></td>
<td>type for the output parameter of the vRealize Orchestrator workflow.</td>
</tr>
</tbody>
</table>

9 (Optional) Select the **Multiple values** check box to allow consumers to select more than one object.

   This option is not available for all field types.

10 Click **Submit**.

11 Click **Update**.

**What to do next**

You can edit the element to change the default settings and apply various constraints or values.

**Edit a Resource Action Element**

You can edit some of the characteristics of an element on the resource action Form page. You can change the type of an element, its default values, and apply various constraints and values.

**Prerequisites**

- Log in to the vRealize Automation console as a **tenant administrator** or **service architect**.
- “Create a Resource Action,” on page 36.

**Procedure**

1 Select **Advanced Services > Resource Actions**.

2 Click the resource action you want to edit.

3 Click the **Form** tab.

4 Locate the element you want to edit.

5 Click the **Edit** icon (📝).

6 Enter a new name for the field in the **Label** text box to change the label that consumers see.

7 Edit the description in the **Description** text box.

8 Select an option from the **Type** drop-down menu to change the display type of the element.

   The options vary depending on the type of element you edit.
9 Select an option from the **Size** drop-down menu to change the size of the element.

10 Select an option from the **Label size** drop-down menu to change the size of the label.

11 Edit the default value of the element.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not set</td>
<td>Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.</td>
</tr>
<tr>
<td>Constant</td>
<td>Sets the default value of the element you are editing to a constant value that you specify.</td>
</tr>
<tr>
<td>Field</td>
<td>Binds the default value of the element to a parameter of another element from the representation.</td>
</tr>
<tr>
<td>Conditional</td>
<td>Applies a condition. By using conditions you can create various clauses and expressions and apply them to an element.</td>
</tr>
<tr>
<td>External</td>
<td>Select a vRealize Orchestrator script action to define the value.</td>
</tr>
</tbody>
</table>

12 Apply constraints to the element on the **Constraints** tab.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not set</td>
<td>Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.</td>
</tr>
<tr>
<td>Constant</td>
<td>Sets the default value of the element you are editing to a constant value that you specify.</td>
</tr>
<tr>
<td>Field</td>
<td>Binds the default value of the element to a parameter of another element from the representation.</td>
</tr>
<tr>
<td>Conditional</td>
<td>Applies a condition. By using conditions you can create various clauses and expressions and apply them to an element.</td>
</tr>
<tr>
<td>External</td>
<td>Select a vRealize Orchestrator script action to define the value.</td>
</tr>
</tbody>
</table>

13 Add one or more values for the element on the **Values** tab.

The options available depend on the type of element you are editing.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not set</td>
<td>Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.</td>
</tr>
<tr>
<td>Predefined values</td>
<td>Select values from a list of related objects from the vRealize Orchestrator inventory.</td>
</tr>
<tr>
<td></td>
<td>a Enter a value in the <strong>Predefined values</strong> search box to search the vRealize Orchestrator inventory.</td>
</tr>
<tr>
<td></td>
<td>b Select a value from the search results and press Enter.</td>
</tr>
<tr>
<td>Value</td>
<td>Define custom values with labels.</td>
</tr>
<tr>
<td></td>
<td>a Enter a value in the <strong>Value</strong> text box.</td>
</tr>
<tr>
<td></td>
<td>b Enter a label for the value in the <strong>Label</strong> text box.</td>
</tr>
<tr>
<td></td>
<td>c Click the <strong>Add</strong> icon (➕).</td>
</tr>
<tr>
<td>External Values</td>
<td>Select a vRealize Orchestrator script action to define your value with information not directly exposed by the workflow.</td>
</tr>
<tr>
<td></td>
<td>■ Select <strong>Add External Value</strong>.</td>
</tr>
<tr>
<td></td>
<td>■ Select your vRealize Orchestrator script action.</td>
</tr>
<tr>
<td></td>
<td>■ Click <strong>Submit</strong>.</td>
</tr>
</tbody>
</table>

14 Click **Submit**.

15 Click **Update**.
Insert a Section Header in a Resource Action Form

You can insert a section header to split the form into sections.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Resource Action,” on page 36.

Procedure

2. Click the resource action you want to edit.
3. Click the Form tab.
4. Drag the Section header element from the Form pane to the Form page pane.
5. Type a name for the section.
6. Click outside of the element to save the changes.
7. Click Update.

Add a Text Element to a Resource Action Form

You can insert a text box to add some descriptive text to the form.

Prerequisites

- Log in to the vRealize Automation console as a tenant administrator or service architect.
- “Create a Resource Action,” on page 36.

Procedure

2. Click the resource action you want to edit.
3. Click the Form tab.
4. Drag the Text element from the New Fields pane to the Form page pane.
5. Enter the text you want to add.
6. Click outside of the element to save the changes.
7. Click Update.
The examples and scenarios identify common tasks in vRealize Automation and suggest ways to accomplish them.

This chapter includes the following topics:

- “Create a Service for Creating and Modifying a User,” on page 69
- “Create and Publish an Action to Migrate a Virtual Machine,” on page 75
- “Create an Action to Migrate a Virtual Machine With vMotion,” on page 76
- “Create and Publish an Action to Take a Snapshot,” on page 79
- “Create and Publish an Action to Start an Amazon Virtual Machine,” on page 81
- “Creating a Data Refresh Service Offering,” on page 83

Create a Service for Creating and Modifying a User

By using Advanced Service Designer, you can create and publish a catalog item for provisioning a user in a group. You can also associate a new post-provisioning operation to the provisioned user, for example, an operation allowing the consumers to change the user password.

In this scenario, as a service architect, you create a new custom resource, a service blueprint, and publish a catalog item for creating a user. You create service and include the catalog item in the service. You also create a resource action for changing the password of the user. In addition, you edit the workflow presentation of the catalog item by using the form designer and change the way the consumers see the request form.

As a business group manager or a tenant administrator, you entitle the newly created service, catalog item, and resource action to a consumer.

Prerequisites

Verify that the Active Directory plug-in is properly configured and you have the rights to create users in Active Directory.

Procedure

1. Create a Test User as a Custom Resource on page 70
   
   You can create a custom resource and map it to the vRealize Orchestrator object type AD:User.

2. Create a Service Blueprint for Creating a User on page 71

   After you created the custom resource, you can create the service blueprint to publish the Create a user in a group workflow as a catalog item.
3 Publish the Create a User Blueprint as a Catalog Item on page 72
   After you create the Create a test user service blueprint, you can publish it as a catalog item.

4 Create a Resource Action to Change a User Password on page 72
   You can create a resource action to allow the consumers of the service to change the password of the
   user after they provision the user.

5 Publish the Change a Password Resource Action on page 73
   To use the Change the password of the Test User resource action as a post-provisioning operation, you
   must publish it.

6 Create a Service for Creating a Test User on page 73
   You can create a service to display in the service catalog and allow consumers to easily locate the
   catalog items related to creating the test user.

7 Associate the Catalog Item with the Create a Test User Service on page 73
   To include the Create a test user catalog item in the Create a Test User service, you must associate it
   with this service.

8 Entitle the Service and the Resource Action to a Consumer on page 74
   Business group managers and tenant administrators can entitle the service and the resource action to a
   user or a group of users so that they can see the service in their catalog and request the Create a test
   user catalog item included in the service. After the consumers provision the item, they can request to
   change the user password.

When consumers of the service log in to their vRealize Automation consoles, they see the service you
created, Create a test user, on the Catalog tab. They can request the catalog item you created and included in
the service, Create a user in a group. After they create the user, they can change the user password.

Create a Test User as a Custom Resource

You can create a custom resource and map it to the vRealize Orchestrator object type AD:User.

Prerequisites

Log in to the vRealize Automation console as a service architect.

Procedure

1 Select Advanced Services > Custom Resources.

2 Click Add (+).

3 Type AD:User in the Inventory Type text box and press Enter.

4 Type a name for the resource.
   For example, Test User.

5 Type a description for the resource.
   For example, This is a test custom resource that I will use for my catalog item to create a user in a
   group.

6 Click Next.

7 Leave the form as is.

8 Click Add.

You created a Test User custom resource and you can see it on the Custom Resources page.
What to do next

Create a service blueprint.

Create a Service Blueprint for Creating a User

After you created the custom resource, you can create the service blueprint to publish the Create a user in a group workflow as a catalog item.

Prerequisites

Log in to the vRealize Automation console as a service architect.

Procedure

1. Select Advanced Services > Service Blueprints.
2. Click Add (+).
3. Navigate to Orchestrator > Library > Microsoft > Active Directory > User in the vRealize Orchestrator workflow library, and select the Create a user in a group workflow.
4. Click Next.
5. Change the name of the blueprint to Create a test user, and leave the description as is.
6. Click Next.
7. Edit the blueprint form.
   a. Click the Edit icon next to the groupContainer text box, and change the name of the label to Group container.
   b. Click Submit to save the changes.
   c. Click the Edit icon next to the accountName text box, and change the label to Account name.
   d. Click Submit to save the changes.
   e. Click the Edit icon next to the domainName text box, and change the name of the label to Domain name.
   f. Click the Constraints tab.
   g. Select Constant from the Value drop-down menu and type test.domain.
      You set the domain name to a constant value.
   h. Click Constant in the Visible drop-down menu and select No.
      You made the domain name invisible to the consumer of the catalog item.
   i. Click Submit to save the changes.
   j. Click the Edit icon next to the displayName text box, and change the label to Display name.
   k. Click Submit to save the changes.
8. Click Next.
9. Select newUser [Test User] as an output parameter to be provisioned.
10. Click Add.

You created a blueprint for creating a test user and you can see it on the Service blueprints page.
What to do next

Publish the Create a test user blueprint to make it an active catalog item.

Publish the Create a User Blueprint as a Catalog Item

After you create the Create a test user service blueprint, you can publish it as a catalog item.

Prerequisites

Log in to the vRealize Automation console as a service architect.

Procedure

1. Select Advanced Services > Service Blueprints.
2. Select the row of the Create a test user blueprint, and click the Publish button.

The status of the Create a test user blueprint changes to Published. You can navigate to Administration > Catalog Management > Catalog Items and see that the Create a test user blueprint is published as a catalog item.

Create a Resource Action to Change a User Password

You can create a resource action to allow the consumers of the service to change the password of the user after they provision the user.

Prerequisites

Log in to the vRealize Automation console as a service architect.

Procedure

2. Click Add (➕).
3. Navigate to Orchestrator > Library > Microsoft > Active Directory > User in the vRealize Orchestrator workflow library, and select the Change a user password workflow.
4. Click Next.
5. Select Test User from the Resource type drop-down menu.
   This is the custom resource you created previously.
6. Select user from the Input parameter drop-down menu.
7. Click Next.
8. Change the name of the resource action to Change the password of the Test User, and leave the description as it appears on the Details tab.
9. Click Next.
10. (Optional) Leave the form as is.
11. Click Add.

You created a resource action for changing the password of a user and you can see it listed on the Resource Actions page.

What to do next

Publish the Change the password of the Test User resource action.
Publish the Change a Password Resource Action

To use the Change the password of the Test User resource action as a post-provisioning operation, you must publish it.

Prerequisites
Log in to the vRealize Automation console as a service architect.

Procedure
2. Select the row of the Change the password of the Test User action, and click the Publish button.

The status of the Change the password of the Test User resource action changes to Published.

What to do next
Assign an icon to the resource action. You can then use the action when you create an entitlement. For more information about assigning an icon to a resource action, see “Assign an Icon to a Resource Action,” on page 38.

Create a Service for Creating a Test User

You can create a service to display in the service catalog and allow consumers to easily locate the catalog items related to creating the test user.

Prerequisites
Log in to the vRealize Automation console as a service architect.

Procedure
1. Select Administration > Catalog Management > Services.
2. Click the Add icon (+).
3. Enter Create a Test User as the name of the service.
4. Select Active from the Status drop-down menu.
5. Leave the other text boxes blank.
6. Click Add.

You created the service called Create a Test User, and you can see it on the Services page.

What to do next
Edit the Create a test user catalog item to include it in the service.

Associate the Catalog Item with the Create a Test User Service

To include the Create a test user catalog item in the Create a Test User service, you must associate it with this service.

Prerequisites
Log in to the vRealize Automation console as a service architect.
Procedure

1. Select Administration > Catalog Management > Catalog Items.
2. Locate the Create a test user catalog item, and click the catalog item name.
3. (Optional) Click Choose File to change the icon of the catalog item.
4. Select the Create a Test User service from the Service drop-down menu.
5. Click Update.

You associated the Create a test user catalog item with the Create a Test User service.

What to do next

Business group managers and tenant administrators can entitle the service and the resource action to a user or a group of users.

Entitle the Service and the Resource Action to a Consumer

Business group managers and tenant administrators can entitle the service and the resource action to a user or a group of users so that they can see the service in their catalog and request the Create a test user catalog item included in the service. After the consumers provision the item, they can request to change the user password.

Prerequisites

Log in to the vRealize Automation console as a tenant administrator or business group manager.

Procedure

1. Select Administration > Catalog Management > Entitlements.
2. Click the Add icon (+).
3. Enter Create a user in the Name text box.
4. Leave the Description and Expiration Date text boxes empty.
5. Select Active from the Status drop-down menu.
6. Select your business group from the Business Group drop-down menu.
7. Enter a user name in the Users & Groups text box and press Enter.
   The person you select can see the service and the catalog items included in the service in the catalog.
8. Click Next.
9. Enter Create a Test User in the Entitled Services text box and press Enter.
10. Enter Change the password of the Test User in the Entitled Resource Actions text box and press Enter.
11. Click Add.

You created an active entitlement and exposed the service to the catalog of the consumers.

When consumers of the service log in to their vRealize Automation consoles, they see the service you created, Create a test user, on the Catalog tab. They can request the catalog item you created and included in the service, Create a user in a group. After they create the user, they can change the user password.
Create and Publish an Action to Migrate a Virtual Machine

By using Advanced Service Designer, you can create and publish a resource action to extend the operations that consumers can perform on IaaS-provisioned vSphere virtual machines.

In this scenario, you create a resource action for quick migration of a vSphere virtual machine.

Prerequisites

Log in to the vRealize Automation console as a service architect.

Procedure

1. Create a Resource Action to Migrate a vSphere Virtual Machine on page 75
   You can create a custom resource action to allow the consumers to migrate vSphere virtual machines after they provision the vSphere virtual machines with IaaS.

2. Publish the Action for Migrating a vSphere Virtual Machine on page 76
   To use the Quick migration of virtual machine resource action as a post-provisioning operation, you must publish it.

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to Administration > Catalog Management > Actions and see the Quick migration of virtual machine resource action in the list of actions. You can assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38.

What to do next

Business group managers and tenant administrators can include the Quick migration of virtual machine resource action in an entitlement. For more information about how to create and publish IaaS blueprints for virtual platforms, see IaaS Configuration for Virtual Platforms.

Create a Resource Action to Migrate a vSphere Virtual Machine

You can create a custom resource action to allow the consumers to migrate vSphere virtual machines after they provision the vSphere virtual machines with IaaS.

Procedure


2. Click Add (➕).

3. Navigate to Orchestrator > Library > vCenter > Virtual Machine management > Move and migrate in the vRealize Orchestrator workflow library and select the Quick migration of virtual machine workflow.

4. Click Next.

5. Select IaaS VC VirtualMachine from the Resource type drop-down menu.

6. Select vm from the Input parameter drop-down menu.

7. Click Next.

8. Leave the name of the resource action and the description as they appear on the Details tab.

9. Click Next.

10. Leave the form as is.

11. Click Add.
You created a resource action for migrating a virtual machine and you can see it listed on the Resource Actions page.

**What to do next**
Publish the resource action.

**Publish the Action for Migrating a vSphere Virtual Machine**

To use the Quick migration of virtual machine resource action as a post-provisioning operation, you must publish it.

**Procedure**

2. Select the row of the Quick migration of virtual machine resource action, and click the Publish button.

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to Administration > Catalog Management > Actions and see the Quick migration of virtual machine resource action in the list of actions. You can assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38.

**What to do next**
Business group managers and tenant administrators can include the Quick migration of virtual machine resource action in an entitlement. For more information about how to create and publish IaaS blueprints for virtual platforms, see IaaS Configuration for Virtual Platforms.

**Create an Action to Migrate a Virtual Machine With vMotion**

By using Advanced Service Designer, you can create and publish a resource action to migrate an IaaS-provisioned virtual machine with vMotion.

In this scenario, you create a resource action to migrate a vSphere virtual machine with vMotion. In addition, you edit the workflow presentation by using the form designer and change the way the consumers see the action when they request it.

**Prerequisites**
Log in to the vRealize Automation console as a service architect.

**Procedure**

1. **Create an Action to Migrate a vSphere Virtual Machine With vMotion** on page 77
   You can create a custom resource action to allow the consumers of the service to migrate a vSphere virtual machine with vMotion after they provision the machine with IaaS.

2. **Edit the Resource Action Form** on page 77
   The form of the resource action maps the vRealize Orchestrator workflow presentation. You can edit the form and define what the consumers of the resource action see when they decide to run the post-provisioning operation.

3. **Add a Submitted Action Details Form and Save the Action** on page 78
   You can add a new form to the Migrate a virtual machine with vMotion resource action to define what the consumers see after they request to run the post-provisioning operation.

4. **Publish the Action for Migrating a Virtual Machine with vMotion** on page 79
   To use the Migrate a virtual machine with vMotion resource action as a post-provisioning operation, you must publish it.
You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to Administration > Catalog Management > Actions and see the Migrate virtual machine with vMotion resource action in the list of actions. You can assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38.

You also edited the presentation of the workflow and defined the look and feel of the action.

**What to do next**

Business group managers and tenant administrators can include the Migrate a virtual machine with vMotion resource action in an entitlement. For more information about how to create and publish IaaS blueprints for virtual platforms, see *IaaS Configuration for Virtual Platforms*.

**Create an Action to Migrate a vSphere Virtual Machine With vMotion**

You can create a custom resource action to allow the consumers of the service to migrate a vSphere virtual machine with vMotion after they provision the machine with IaaS.

**Procedure**

2. Click Add (✚).
3. Navigate to Orchestrator > Library > vCenter > Virtual Machine management > Move and migrate in the vRealize Orchestrator workflow library and select the Migrate virtual machine with vMotion workflow.
4. Click Next.
5. Select IaaS VC VirtualMachine from the Resource type drop-down menu.
6. Select vm from the Input parameter drop-down menu.
7. Click Next.
8. Leave the name of the resource action and the description as they appear on the Details tab.
9. Click Next.

**What to do next**

Edit the resource action form.

**Edit the Resource Action Form**

The form of the resource action maps the vRealize Orchestrator workflow presentation. You can edit the form and define what the consumers of the resource action see when they decide to run the post-provisioning operation.

**Procedure**

1. Click the Delete icon (❌) to delete the pool element.
2. Edit the host element.
   a. Click the Edit icon (✎) next to the host field.
   b. Type Target host in the Label text box.
   c. Select Search from the Type drop-down menu.
   d. Click the Constraints tab.
e Select **Constant** from the **Required** drop-down menu and select **Yes**. You made the host field always required.

f Click **Submit**.

3 Edit the **priority** element.

a Click the **Edit** icon ( ) next to the **priority** field.

b Type **Priority of the task** in the **Label** text box.

c Select **Radio button group** from the **Type** drop-down menu.

d Click the **Values** tab, and deselect the **Not set** check box.

e Enter **lowPriority** in the **Predefined values** search text box, and press Enter.

f Enter **defaultPriority** in the **Predefined values** search text box, and press Enter.

g Enter **highPriority** in the **Predefined values** search text box, and press Enter.

h Click **Submit**.

When the consumers request the resource action, they see a radio button group with three radio buttons: **lowPriority**, **defaultPriority**, and **highPriority**.

4 Edit the **state** element.

a Click the **Edit** icon ( ) next to the **state** field.

b Type **Virtual machine state** in the **Label** text box.

c Select **Drop-down** from the **Type** drop-down menu.

d Click the **Values** tab, and deselect the **Not set** check box.

e Enter **poweredOff** in the **Predefined values** search text box, and press Enter.

f Enter **poweredOn** in the **Predefined values** search text box, and press Enter.

g Enter **suspended** in the **Predefined values** search text box, and press Enter.

h Click **Submit**.

When the consumers request the resource action, they see a drop-down menu with three options: **poweredOff**, **poweredOn**, and **suspended**.

You edited workflow presentation of the Migrate a virtual machine with vMotion workflow.

**What to do next**

Add a new form and save the resource action.

### Add a Submitted Action Details Form and Save the Action

You can add a new form to the Migrate a virtual machine with vMotion resource action to define what the consumers see after they request to run the post-provisioning operation.

**Procedure**

1 Click the **New Form** icon ( ) next to the **Form** drop-down menu.

2 Type **Submitted action** in the **Name** text box.

3 Leave the **Description** field blank.

4 Select **Submitted action details** from the **Screen type** menu.
5 Click Submit.

6 Click the Edit icon (-pencil) next to the Form page drop-down menu.

7 Type Details in the Heading text box.

8 Click Submit.

9 Drag the Text element from the Form pane and drop it to the Form page.

10 Type

   You submitted a request to migrate your machine with vMotion. Wait until the process completes successfully.

11 Click outside of the text box to save the changes.

12 Click Submit.

13 Click Add.

You created a resource action to migrate a virtual machine with vMotion and you can see it listed on the Resource Actions page.

What to do next
Publish the resource action to make it active.

Publish the Action for Migrating a Virtual Machine with vMotion

To use the Migrate a virtual machine with vMotion resource action as a post-provisioning operation, you must publish it.

Procedure

1 Select Advanced Services > Resource Actions.

2 Select the row of the Migrate a virtual machine with vMotion action, and lick the Publish button.

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to Administration > Catalog Management > Actions and see the Migrate virtual machine with vMotion resource action in the list of actions. You can assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38.

You also edited the presentation of the workflow and defined the look and feel of the action.

What to do next
Business group managers and tenant administrators can include the Migrate a virtual machine with vMotion resource action in an entitlement. For more information about how to create and publish IaaS blueprints for virtual platforms, see IaaS Configuration for Virtual Platforms.

Create and Publish an Action to Take a Snapshot

By using Advanced Service Designer, you can create and publish a resource action to take a snapshot of a vSphere virtual machine provisioned with IaaS.

In this scenario, you create a resource action to take a snapshot of a vSphere virtual machine provisioned with IaaS. In addition, you edit the workflow presentation by using the form designer and change the way the consumers see the action when they request it.

Prerequisites
Log in to the vRealize Automation console as a service architect.
Procedure

1. Create the Action to Take a Snapshot of a vSphere Virtual Machine on page 80
   You can create a custom resource action to allow the consumers to take a snapshot of a vSphere virtual machine after they provision the machine with IaaS.

2. Publish the Action for Taking a Snapshot on page 81
   To use the Create a snapshot resource action as a post-provisioning operation, you must publish it.

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to Administration > Catalog Management > Actions and see the Create a snapshot resource action in the list of actions. You can assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38.

What to do next

Business group managers and tenant administrators can include the Create a snapshot resource action in an entitlement. For more information about how to create and publish IaaS blueprints for virtual platforms, see IaaS Configuration for Virtual Platforms.

Create the Action to Take a Snapshot of a vSphere Virtual Machine

You can create a custom resource action to allow the consumers to take a snapshot of a vSphere virtual machine after they provision the machine with IaaS.

Procedure


2. Click Add (➕).

3. Navigate to Orchestrator > Library > vCenter > Virtual Machine management > Snapshot in the vRealize Orchestrator workflow library and select the Create a snapshot workflow.

4. Click Next.

5. Select IaaS VC VirtualMachine from the Resource type drop-down menu.

6. Select vm from the Input parameter drop-down menu.

7. Click Next.

8. Leave the name of the resource action and the description as they appear on the Details tab.

9. Click Next.

10. Leave the form as is.

11. Click Add.

You created a resource action for taking a snapshot of a virtual machine and you can see it listed on the Resource Actions page.

What to do next

Publish the resource action.
Publish the Action for Taking a Snapshot

To use the Create a snapshot resource action as a post-provisioning operation, you must publish it.

Procedure
2. Select the row of the Create a snapshot action, and click the Publish button.

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to Administration > Catalog Management > Actions and see the Create a snapshot resource action in the list of actions. You can assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38.

What to do next
Business group managers and tenant administrators can include the Create a snapshot resource action in an entitlement. For more information about how to create and publish IaaS blueprints for virtual platforms, see IaaS Configuration for Virtual Platforms.

Create and Publish an Action to Start an Amazon Virtual Machine

By using the Advanced Service Designer, you can create and publish actions to extend the operations that the consumers can perform on third-party provisioned resources.

In this scenario, you create and publish a resource action for quick starting of Amazon virtual machines.

Prerequisites
- Install the vRealize Orchestrator plug-in for Amazon Web Services on your default vRealize Orchestrator server.
- Create or import a vRealize Orchestrator workflow for resource mapping of Amazon instances.

Procedure
1. Create a Resource Mapping for Amazon Instances on page 82
   You can create a resource mapping to associate Amazon instances provisioned by using IaaS with the vRealize Orchestrator type AWS:EC2Instance exposed by the Amazon Web Services plug-in.
2. Create a Resource Action to Start an Amazon Virtual Machine on page 82
   You can create a resource action so that the consumers can start provisioned Amazon virtual machines.
3. Publish the Action for Starting Amazon Instances on page 83
   To use the newly created Start Instances resource action for post-provisioning operations on Amazon virtual machines, you must publish it.

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to Administration > Catalog Management > Actions and see the Start Instances resource action for EC2 Instance resource types. You can assign an icon to the resource action. See “Assign an Icon to a Resource Action,” on page 38.

What to do next
Business group managers and tenant administrators can assign the Start Instances resource action to an entitlement.
Create a Resource Mapping for Amazon Instances

You can create a resource mapping to associate Amazon instances provisioned by using IaaS with the vRealize Orchestrator type AWS:EC2Instance exposed by the Amazon Web Services plug-in.

Prerequisites

- Log in to the vRealize Automation console as a service architect.
- Create or import a vRealize Orchestrator resource mapping workflow or script action.

Procedure

2. Click Add (+).
3. Enter EC2 Instance in the Name text box.
4. Enter Cloud Machine in the Catalog Resource Type text box.
5. Enter AWS:EC2Instance in the Orchestrator Type text box.
6. Select Always available.
7. Select the type of resource mapping to use.
8. Select your custom resource mapping script action or workflow from the vRealize Orchestrator library.
9. Click Add.

You can use your Amazon resource mapping to create resource actions for Amazon machines provisioned by using IaaS.

What to do next

Create a resource action for Amazon instances.

Create a Resource Action to Start an Amazon Virtual Machine

You can create a resource action so that the consumers can start provisioned Amazon virtual machines.

Prerequisites

Log in to the vRealize Automation console as a service architect.

Procedure

2. Click Add (+).
3. Select Orchestrator > Library > Amazon Web Services > Elastic Cloud > Instances and select the Start Instances workflow in the workflows folder.
4. Click Next.
5. Select EC2 Instance from the Resource type drop-down menu.
   This is the name of the resource mapping you previously created.
6. Select instance from the Input parameter drop-down menu.
   This is the input parameter of the resource action workflow to match the resource mapping.
7. Click Next.
8 Leave the name and the description as they are.
   The default name of the resource action is Start Instances.
9 Click Next.
10 Leave the fields as they are on the Form tab.
11 Click Add.

You created a resource action for starting Amazon virtual machines and you can see it on the Resource
Actions page.

What to do next
Publish the Start Instances resource action.

**Publish the Action for Starting Amazon Instances**

To use the newly created Start Instances resource action for post-provisioning operations on Amazon virtual
machines, you must publish it.

**Prerequisites**

Log in to the vRealize Automation console as a service architect.

**Procedure**

1 Select Advanced Services > Resource Actions.
2 Select the row of the Start Instances resource action, and click Publish.

The status of the Start Instances resource action changes to Published.

What to do next
Assign an icon to the Start Instances resource action.

**Creating a Data Refresh Service Offering**

By using vRealize Orchestrator, Application Services, Advanced Service Designer, and IaaS, you can create
an advanced service offering to refresh external database backups in a golden database node, mask the data,
and clone the database nodes without increasing storage costs.

Data Refresh is a database-oriented feature that uses components to provide a complete feature in
vRealize Automation. Data Refresh helps you refresh the external database backup. The components
include vRealize Orchestrator, Application Services, IaaS, Advanced Service Designer and Service Catalog.
Data Refresh duplicates the copies of databases from one environment to another. For example, a developer
or a test engineer can refresh the database from production to development or testing.

Data Refresh helps you refresh the external database backups in a golden database node, mask the data in
the golden database, and clone any number of database nodes from the golden database node without
increasing storage costs. A golden database is a database instance that contains preset data obtained through
import and mask operations. The golden database is in a template node. You can duplicate multiple copies
of a golden database and clone from the template node.

The Data Refresh process consists of the following phases.

- Provisioning a golden database node
- Requesting a golden database clone
You must prepare an environment before users can provision and request to clone a database.

- Complete the vRealize Automation environment preparation. For more information, see “Preparing a Data Refresh Environment Checklist,” on page 84.

- The database administrator (DBA) configures Application Services, including application and deployment profile.

- DBA configures the data refresh service catalog through Advanced Service Designer and vRealize Automation.

- DBA requests the golden database node with the prepared mask data SQL script.

- Trigger cloning by DBA or by the vRealize Automation user.

The DBA uses data refresh repeatedly. vRealize Automation allows DBAs to define their own data refresh workflow to generate a golden database and clone the golden database.

Oracle Database 12c R1 and SQL Server 2012 Enterprise Edition for single node are supported in vRealize Automation 6.1. DBAs follow the sample to create the Data Refresh vRealize Orchestrator workflow and are not limited to Oracle 12c and SQL Server 2012. You can change either the database type or refresh backup source type through the customized Data Refresh vRealize Orchestrator workflow.

### Preparing a Data Refresh Environment Checklist

Data Refresh helps you refresh the external database backups in a golden database node, mask the data in the golden database, and clone any number of database nodes from the golden database node. You must prepare an environment to provision and clone the database nodes.

The preparing a data refresh environment checklist provides a high-level overview of the sequence of steps required to prepare an environment to generate a golden database.

<table>
<thead>
<tr>
<th>Table 6-1. Preparing a Data Refresh Environment Checklist</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Task</strong></td>
</tr>
<tr>
<td>Complete all required IaaS configurations, and set up reservation policies and network profiles.</td>
</tr>
<tr>
<td>Register vRealize Automation server from the Application Services client in Application Services.</td>
</tr>
<tr>
<td>Create the cloud provider and deployment environment in Application Services.</td>
</tr>
<tr>
<td>Log in to Application Services as a system administrator.</td>
</tr>
<tr>
<td>Download the database installation files from the database website to any location on the server.</td>
</tr>
<tr>
<td>Prepare a set of backups (plus control file backup) in a server or directory appropriately.</td>
</tr>
<tr>
<td>Prepare a virtual machine on vCenter Server with vRealize Automation and the Application Services agent installed.</td>
</tr>
<tr>
<td>Create a vRealize Automation machine blueprint of type vSphere vCenter based on the virtual machine in vCenter Server.</td>
</tr>
<tr>
<td>Add the vSphere vCenter machine blueprint in the cloud provider and logical template in Application Services.</td>
</tr>
</tbody>
</table>
### Table 6-1. Preparing a Data Refresh Environment Checklist (Continued)

<table>
<thead>
<tr>
<th>Task</th>
<th>Comments</th>
</tr>
</thead>
</table>
| ☐ In Application Services, create an application and application version. | 1 Create a blueprint.  
2 Add a database node.  
3 Add an appropriate database service in the database node.  
4 Add additional disks with appropriate disk tags in the database node.  
5 Set the properties in the appropriate database service.  
6 Bind the disk_layout_info to self:disk_layout_info. See Using Application Services. |
| ☐ Create a deployment profile for the application version.          | See Using Application Services.                                           |

### Provisioning a Golden Database Node Checklist

A golden database is a database instance with preset data. It is contained in a template node and can be duplicated to any number of copies along with cloning from the template node.

Ensure that you prepared the environment for data refresh for which you want to generate the golden database. See “Preparing a Data Refresh Environment Checklist,” on page 84.

Log in to the vRealize Automation console as a service architect.

### Table 6-2. Provisioning a Golden Database Node Checklist

<table>
<thead>
<tr>
<th>Task</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐ In vRealize Orchestrator, create a data refresh workflow to publish as a catalog item.</td>
<td>You can use the sample data refresh workflows provided or create your own.</td>
</tr>
<tr>
<td>☐ Create a data refresh service blueprint the data refresh workflow you created in vRealize Orchestrator.</td>
<td>For information about creating service blueprints, see “Create a Service Blueprint,” on page 34.</td>
</tr>
<tr>
<td>☐ Publish your data refresh service blueprint.</td>
<td>See “Publish a Service Blueprint as a Catalog Item,” on page 36.</td>
</tr>
<tr>
<td>☐ Create a Data Management service for catalog items such as data refresh.</td>
<td>See “Create a Service,” on page 39.</td>
</tr>
<tr>
<td>☐ Associate your data refresh catalog item with the data management service you created.</td>
<td>See “Associate a Catalog Item with a Service,” on page 39.</td>
</tr>
<tr>
<td>☐ Entitle users to your data management service and data refresh catalog item.</td>
<td>Log in to the vRealize Automation console as a tenant administrator or business group manager. See “Entitle a Service, Catalog Items, and Actions to a User or a Group of Users,” on page 42.</td>
</tr>
<tr>
<td>☐ Select Request for the appropriate catalog item on the Service Catalog page.</td>
<td></td>
</tr>
</tbody>
</table>
Table 6-2. Provisioning a Golden Database Node Checklist (Continued)

<table>
<thead>
<tr>
<th>Task</th>
<th>Comments</th>
</tr>
</thead>
</table>
| ☐ Provision a golden database node for the appropriate database on the New Request page. | 1 On the Request Information tab, enter the Description.  
2 On the Register App Services Instance tab, enter the values for the Application Services address, tenant, username, and password.  
3 On the Create Database tab, enter the values for the Application Services name, owning group, version, qualifier, and deployment profile name.  
4 On the Find Import Data Task tab, retain the default values for the Application Services task name, task version, and owning group. Set the parameters on the Import tab to import the data.  
5 On the Find Run SQL Script Task tab, retain the default values for the Application Services task name, task version, and owning group.  
6 On the Mask Data tab, enter the mask data SQL script and the required database access parameters.  
7 On the Execution Settings tab, set the timeout in minutes for each task.  
Click Submit to complete the request. |

☐ On the Requests page, select the completed request and check the request details for Application Services Deployment ID, name, node name, node hostname, and node IP address.

Requesting a Golden Database Clone Checklist

You can use several methods to clone a virtual machine in vRealize Automation. You can use an IaaS machine blueprint to link clone a database node from the generated database node to save the storage cost.

Ensure that you complete the following tasks before you request a database clone.

- Shut down the database instance by using an appropriate database tool.
- Ensure that you configured the database node before you clone the database.
- Log in to the vRealize Automation console as a tenant administrator or business group manager.

Table 6-3. Requesting a Golden Database Clone Checklist

<table>
<thead>
<tr>
<th>Task</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐ Edit the existing Data Management entitlement.</td>
<td>Add the Power Off (Machine) and Power On (Machine) to Entitled Actions.</td>
</tr>
<tr>
<td>☐ Select the machine, click Item Details, and click Power Off under Actions.</td>
<td>Wait for the machine status to turn off.</td>
</tr>
<tr>
<td>☐ Create a new snapshot.</td>
<td>For more information, see Managing Virtual Machines in Tenant Administration.</td>
</tr>
<tr>
<td>☐ Create and publish a new machine blueprint.</td>
<td>For more information, see IaaS Configuration for Virtual Platforms.</td>
</tr>
<tr>
<td>☐ Create a new service, golden database in Catalog Management.</td>
<td>For more information, see Managing the Service Catalog in Tenant Administration.</td>
</tr>
<tr>
<td>☐ Update the published machine blueprint in the Catalog Items list and set the service to golden database.</td>
<td></td>
</tr>
<tr>
<td>Task</td>
<td>Comments</td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
</tr>
</tbody>
</table>
| ☐ Add a new Entitlement and set the name to golden database. | This action adds the following items.  
- Golden database in Entitled Services  
- Published machine blueprint in Entitled Catalog Items  
- The granted users or groups to Users and Groups |
| ☐ Request a clone from the published machine blueprint in the golden database and grant the appropriate user and groups to access the published machine blueprint. | On the New Request page, set the CPU and memory details. Click **Submit**. |
| ☐ In Request Details, find the clone machine name from Service Catalog. | |
| ☐ Use an appropriate database tool to start the clone database instance. | |
# Index

## A
- Active Directory plug-in, adding endpoint 21
- Active Directory plug-in overview 13
- adding a new form
  - resource actions 64
  - service blueprints 60
- adding a new form page 57
- advanced service, creating 32
- Advanced Service Design 31
- advanced services 69
- advanced service design, updated information, advanced service design 9
- Advanced Service Designer, form designer 51
- advanced service endpoint, creating 21
- advanced service endpoints
  - deleting 29
  - editing 28
- approval policies
  - creating 40
  - overview 40
  - post-approval level 41
  - pre-approval level 41
- approval policy, general settings 40

## C
- catalog items
  - associating with a service 39
  - including in a service 39
  - including in an example service 73
  - publishing 36
- Clone 86
- concepts 31
- Configuring database instance 85
- constraints, form designer 53
- create a resource action for starting an Amazon virtual machine 82
- create a resource mapping for Amazon instances 82
- creating script actions and workflows 45
- custom resources
  - creating 33
  - editing the form 56
  - exporting 49
  - importing 49
  - overview 31
- custom resource elements, editing 57

## D
- Data Refresh environment 84
- Data Refresh Overview 83
- Database clone 86
- Database Instance 84
- dynamic forms, Advanced Service Designer 51

## E
- editing a form, examples 77
- embedded Orchestrator instance 15
- endpoints
  - deleting 29
  - editing 28
  - updating 28
- entitling
  - example resource action 74
  - example service to a user 74
  - resource action 42
  - service to a user 42
- example
  - associating catalog items with a service 73
  - changing user password 72
  - creating services 73
  - creating user in a group 71
  - creating a custom resource 70
  - entitling service to a customer 74
  - migrating virtual machine 75, 77
  - publishing a resource action 73, 76, 79, 81
  - resource action for creating and modifying a user 71
  - resource action for migrating virtual machine 75
  - resource action for migrating virtual machine with vMotion 76
  - resource action for taking a snapshot 79
  - service for creating and modifying a user 69
  - take a snapshot of a virtual machine 80
  - example resource action, editing the form 77
  - examples 69
- export, Advanced Service Designer components 49
- external Orchestrator server, configuring 19
- external value definitions
  - adding 58
  - assigning in forms designer 55
F
form designer
constraints to apply  53
new fields  52
overview  55
forms designer, editing custom resources  56

G
Generate Database  85
goal navigator, using  11
Golden Database  85

H
HTTP-REST plug-in, adding endpoints  22
HTTP-REST plug-in overview  13

I
import, Advanced Service Designer
components  49

L
log in to
  Orchestrator client  17
  Orchestrator configuration interface  17
  starting  17

M
mapping a resource to Orchestrator  31, 32

O
Orchestrator
  default root folder  20
  integration  15
  overview  13
  Orchestrator plug-in
    Active Directory  21
    HTTP-REST  22
    PowerShell  24
    SOAP  25
    vCenter Server  27
  Orchestrator configuration interface, logging in  17
  Orchestrator object types  31
  Orchestrator plug-ins, adding as an endpoint  21
  Orchestrator plug-ins, overview  13
  Overview  83
  overview of
    approval policies  40
    custom resources  31
    form designer  55
    Orchestrator  13
    Orchestrator plug-ins  13

P
plug-ins, installing  29
post-approval levels, creating  41
PowerShell plug-in, adding endpoint  24
PowerShell plug-in overview  13
pre-approval levels, creating  41
publish a resource action  83
publishing
  example resource actions  73, 76, 79, 81
  example service blueprint  72
  resource actions  38, 48
  service blueprints  36

R
resource actions
  adding a new field  62, 64
  adding a new form  64
  adding an example form  78
  assigning an icon  38, 48
  changing user password  72
  creating  36, 46
  editing fields  65
  editing the form  64
  example  72
  exporting  49
  extending operations on IaaS machines  46
  for items provisioned outside of Advanced Service Designer  43, 44
  importing  49
  migrating virtual machine  75, 77
  new fields  52
  overview  32
  publishing  38, 48
  saving an example action  78
  take a snapshot  80
  resource mapping  32, 82
  resource mappings
    create  45
    exporting  49
    high-level overview of procedure  44
    importing  49
    vCenter  45
    vCloud Director  45
    vSphere  45
    resource action for starting an Amazon virtual machine  81

S
script actions, assigning as external values  55
section headers, adding  58, 63, 67
service, including catalog items  39
service blueprints
  adding a new form  60
adding a new field 62, 64
creating 34
creating user in a group 71
editing fields 60
editing the form 59
exporting 49
importing 49
new fields 52
overview 32
publishing 36
services
creating 39
including an example catalog item 73
SOAP plug-in, adding endpoints 25
SOAP plug-in overview 13
starting the Orchestrator configuration
interface 17
submitted action details form, adding 78

T
tenants, editing workflow folder access 20
text boxes, adding 58, 63, 67

U
updated information 9

V
values, form designer 53
vApps, extending operations 46
vCenter, resource mappings 45
vCenter Orchestrator, configuring external 19
vCenter Server plug-in, adding endpoints 27
vCloud Director, resource mappings 45
virtual machines, extending operations 46
vSphere, resource mappings 45

W
workflows, importing 49
workflows per tenant, configuring default
folder 20