

Using the vCenter Orchestrator Plug-In for vCloud Automation Center 6.1

vRealize Orchestrator 5.5.2
vRealize Automation 6.1.0



vmware®

You can find the most up-to-date technical documentation on the VMware website at:

<https://docs.vmware.com/>

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2017 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

Contents

Using the vCenter Orchestrator Plug-In for vCloud Automation Center 6.1 4

- 1 Introduction to the VMware vCenter Orchestrator Plug-In for vCloud Automation Center 5**
 - Role of vCenter Orchestrator with the vCloud Automation Center Plug-In 5
 - Installing the vCloud Automation Center Plug-In 6
 - vCloud Automation Center Plug-In Functional Prerequisites 6
 - Install the vCloud Automation Center Plug-In On an External vCenter Orchestrator Server 6

- 2 Configuring the vCloud Automation Center Plug-In 8**
 - Configuration Workflows 8
 - Add a vCloud Automation Center Host 9
 - Add a vCloud Automation Center Infrastructure Administration Host 10

- 3 Using the vCloud Automation Center Plug-In 11**
 - Using the vCloud Automation Center Plug-In Inventory 11
 - Using the vCloud Automation Center Plug-In Administration Workflows 12
 - Using the vCloud Automation Center Plug-In Infrastructure Administration Workflows 15
 - Create a vCloud Automation Center Model Entity 17
 - Read a vCloud Automation Center Model Entity 18
 - Using the vCloud Automation Center Plug-In Requests Workflows 18
 - Using the vCloud Automation Center Plug-In Sample Workflows 19
 - Access the vCloud Automation Center Plug-In API 19
 - Common Tasks Example Scripts 20
 - CRUD Infrastructure Administration Tasks Example Scripts 22
 - Finding vCloud Automation Center Entities Example Scripts 26
 - Get a Resource Provisioned by vCloud Automation Center Example Script 28
 - Transfer Content Between vCloud Automation Center Hosts Example Script 29

Using the vCenter Orchestrator Plug-In for vCloud Automation Center 6.1

Using the vCenter Orchestrator Plug-In for vCloud Automation Center 6.1 provides information and instructions about configuring and using the VMware® vCenter Orchestrator plug-in for VMware vCloud Automation Center 6.1.

Intended Audience

The information in *Using the vCenter Orchestrator Plug-In for vCloud Automation Center 6.1* is written for experienced users who are familiar with virtual machine technology, with Orchestrator workflow development, and with VMware vCloud Automation Center.

For more information about Orchestrator, see http://www.vmware.com/support/pubs/orchestrator_pubs.html.

For more information about vCloud Automation Center, see <http://www.vmware.com/support/pubs/vcac-pubs.html>.

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>.

Introduction to the VMware vCenter Orchestrator Plug-In for vCloud Automation Center

1

The VMware vCenter Orchestrator plug-in for vCloud Automation Center allows interaction between vCenter Orchestrator and vCloud Automation Center.

You can use the vCloud Automation Center plug-in to create and run workflows for the following vCloud Automation Center functions:

- Service management
- Catalog item and resource management and requesting
- Advanced Services custom resource and blueprint management
- Entitlement configuration
- Approval policy configuration
- Work item interactions
- vSphere and vCloud Director virtual machine provisioning and post-provisioning actions
- Create, read, update, and delete (CRUD) operations on the vCloud Automation Center IaaS model

This section includes the following topics:

- [Role of vCenter Orchestrator with the vCloud Automation Center Plug-In](#)
- [Installing the vCloud Automation Center Plug-In](#)

Role of vCenter Orchestrator with the vCloud Automation Center Plug-In

You use the Orchestrator client to run and create workflows and access the plug-in API. You can use either the embedded vCenter Orchestrator instance in your vCloud Automation Center installation, or an external vCenter Orchestrator server.

vCenter Orchestrator powers the vCloud Automation Center plug-in. Orchestrator is a development and process-automation platform that provides a library of extensible workflows to manage the vCenter infrastructure and other technologies.

Orchestrator allows integration with management and administration solutions through its open plug-in architecture.

Installing the vCloud Automation Center Plug-In

The vCloud Automation Center plug-in is installed on the embedded vCenter Orchestrator instance in your vCloud Automation Center installation. You can use the configuration interface of an external vCenter Orchestrator server to install the vCloud Automation Center plug-in.

vCloud Automation Center Plug-In Functional Prerequisites

To install and use the vCloud Automation Center plug-in, your system must meet certain functional prerequisites.

vCloud Automation Center

You must have access to a vCloud Automation Center server. Version 6.1 of the plug-in works with vCloud Automation Center 6.1.

For information about setting up vCloud Automation Center, see vCloud Automation Center *Installation and Configuration*.

External vCenter Orchestrator Server

To use an external vCenter Orchestrator server, you must have a running instance of Orchestrator. Version 6.1 of the plug-in works with vCenter Orchestrator 5.5.2.

For information about setting up Orchestrator, see *Installing and Configuring VMware vCenter Orchestrator*.

Install the vCloud Automation Center Plug-In On an External vCenter Orchestrator Server

To use the plug-in on an external vCenter Orchestrator server, you must install the plug-in by using the Orchestrator configuration interface.

Prerequisites

- Download the .vmoapp file that contains the plug-in.
- Log in to the Orchestrator configuration interface at http://orchestrator_server:8283.

Procedure

- 1 On the **General** tab, click **Install Application**.
- 2 Upload the vCloud Automation Center plug-in.
 - a Click the magnifying glass icon.
 - b Select the .vmoapp file to install.
 - c Click **Open**.
 - d Click **Install**.

A message appears after successful installation. The vCloud Automation Center plug-in is installed without a tab in the Orchestrator configuration interface.

- 3 On the **Startup Options** tab, click **Restart service** to complete the plug-in installation.

What to do next

Configure the vCloud Automation Center plug-in.

Configuring the vCloud Automation Center Plug-In

2

You add vCloud Automation Center hosts and IaaS hosts to configure the plug-in.

Configuration Workflows

You can use the workflows in the **Configuration** workflow categories to manage vCloud Automation Center hosts.

vCloud Automation Center Hosts

You can access these workflows from the **Workflows** view of the Orchestrator client, in the **Configuration** subdirectory of the plug-in library.

Workflow Name	Description
Add a vCAC host	Adds a vCloud Automation Center host to the plug-in inventory. For tenant management and administration tasks, you can use the Inventory view to run workflows on each tenant. To use the full function of the plug-in for a tenant, create a dedicated vCloud Automation Center host for each tenant.
Add a vCAC host using component registry	Adds a vCloud Automation Center host to the plug-in inventory with a Per User Session connection. You must be logged in to the Orchestrator client with the credentials of the vCloud Automation Center system administrator. To use this function with an external vCenter Orchestrator server, you must register the Orchestrator server in the vCloud Automation Center component registry. To do this registration, you must run the Register vCO in vCAC Component Registry workflow and restart the Orchestrator server.
Add the IaaS host of a vCAC host	Adds the IaaS host of the selected vCloud Automation Center host to the plug-in inventory.
Remove a vCAC host	Removes a vCloud Automation Center host from the plug-in inventory.
Update a vCAC host	Updates a vCloud Automation Center host in the plug-in inventory.
Validate a vCAC host	Validates the vCloud Automation Center host and the connection to it.

Note If your vCenter Orchestrator server is registered in the vCloud Automation Center component registry, a vCloud Automation Center host with the name Default is automatically added. The Default host is using Per User Session connection to the default tenant. The embedded Orchestrator server in the vCloud Automation Center installation is registered in the vCloud Automation Center component registry by default.

vCloud Automation Center IaaS Hosts

You can access these workflows from the **Workflows** view of the Orchestrator client, in the **Infrastructure Administration > Configuration** subdirectory of the plug-in library.

The embedded Orchestrator server in the vCloud Automation Center installation is registered in the vCloud Automation Center component registry by default.

Workflow Name	Description
Add an IaaS host	Adds a vCloud Automation Center IaaS host to the plug-in inventory. This workflow is functionally the same as Add the IaaS host of a vCAC host, but does not require a vCloud Automation Center host.
Remove an IaaS host	Removes a vCloud Automation Center IaaS host from the plug-in inventory.
Update an IaaS host	Updates a vCloud Automation Center IaaS host in the plug-in inventory.
Validate an IaaS host	Validates the vCloud Automation Center IaaS host and the connection to it.

Add a vCloud Automation Center Host

You can run a workflow to a vCloud Automation Center host and configure the host connection parameters.

Procedure

- 1 From the drop-down menu in the Orchestrator client, select **Run** or **Design**.
- 2 Click the **Workflows** view.
- 3 Select **Library > vCloud Automation Center > Configuration**.
- 4 Right-click the **Add a vCAC host** workflow and select **Start workflow**.
- 5 Provide the required information and click **Next**.
- 6 From the **Session mode** drop-down menu, select the type of connection to the host.

Option	Actions
Shared Session	<p>The connection to the host uses the credentials you provide for a vCloud Automation Center user in a tenant.</p> <ol style="list-style-type: none"> a In the Tenant text box, enter the name of the tenant. b In the Authentication username and Authentication password text boxes, enter the credentials for a user in the tenant.
Per User Session	<p>The connection to the host uses the credentials of the user that is currently logged in. You must be logged in to the Orchestrator client with the credentials of the vCloud Automation Center system administrator.</p> <p>To use this option with an external vCenter Orchestrator server, you must register the Orchestrator server in the vCloud Automation Center component registry. To do the registration, you must run the Register vCO in vCAC Component Registry workflow and restart the Orchestrator server.</p>

- 7 Click **Submit**.

What to do next

Add a vCloud Automation Center Infrastructure Administration host.

Add a vCloud Automation Center Infrastructure Administration Host

You can run a workflow to add the IaaS host of a vCloud Automation Center host and configure the connection parameters.

Procedure

- 1 From the drop-down menu in the Orchestrator client, select **Run** or **Design**.
- 2 Click the **Workflows** view.
- 3 Select **Library > vCloud Automation Center > Configuration**.
- 4 Right-click **Add the IaaS host of a vCAC host** and select **Start workflow**.
- 5 From the **vCAC host** drop-down menu, select the vCloud Automation Center host for which to configure an IaaS host and click **Next**.
- 6 Provide the required information and click **Next**.
- 7 From the **Session mode** drop-down menu, select the type of connection to the host.

Option	Actions
Shared Session	The connection to the host uses the credentials you provide for a vCloud Automation Center user who has access rights to the machine where the IaaS software is installed. In the Authentication user name and Authentication password text boxes, enter the user credentials.
Per User Session	The connection to the host uses the credentials of the user that is currently logged in. You must be logged in to the Orchestrator client with the credentials of a vCloud Automation Center user who has access rights to the machine where the IaaS software is installed.

- 8 Click **Next**.
- 9 Enter the name of the Workstation machine and the NetBIOS domain name.
- 10 Click **Submit**.

Using the vCloud Automation Center Plug-In

3

The vCloud Automation Center plug-in workflow library contains workflows that you can use for common tasks such as interacting with the catalog, managing infrastructure, and creating tenants and services.

You can use custom HTTP headers, such as the vCloud Automation Center specific headers Tasks and Identity, and apply them in the CRUD, provisioning, and post-provisioning workflows.

This section includes the following topics:

- [Using the vCloud Automation Center Plug-In Inventory](#)
- [Using the vCloud Automation Center Plug-In Administration Workflows](#)
- [Using the vCloud Automation Center Plug-In Infrastructure Administration Workflows](#)
- [Using the vCloud Automation Center Plug-In Requests Workflows](#)
- [Using the vCloud Automation Center Plug-In Sample Workflows](#)
- [Access the vCloud Automation Center Plug-In API](#)
- [Common Tasks Example Scripts](#)
- [CRUD Infrastructure Administration Tasks Example Scripts](#)
- [Finding vCloud Automation Center Entities Example Scripts](#)
- [Get a Resource Provisioned by vCloud Automation Center Example Script](#)
- [Transfer Content Between vCloud Automation Center Hosts Example Script](#)

Using the vCloud Automation Center Plug-In Inventory

You can use the **Inventory** view to run workflows on vCloud Automation Center objects.

To display the workflows that are available for an inventory object, navigate to **Tools > User preferences > Inventory** and select the **Use contextual menu in inventory** check box. After the option is enabled, when you right-click an object in the Orchestrator inventory, all available workflows for the object are displayed.

Using the vCloud Automation Center Plug-In Administration Workflows

You can use the administration workflows to manage vCloud Automation Center services, tenants, approval policies, entitlements, business groups, catalog items, and advanced services custom resources, service blueprints, and resource actions.

You can find these workflows on the **Workflows** view of the Orchestrator client, in the **Administration** subdirectory of the plug-in library.

You can use the workflows in the **Approval Policies** subdirectory to copy, activate, deactivate, and delete approval policies.

Table 3-1. Approval Policies

Workflow	Description
Activate an approval policy	Activates an approval policy. After you activate an approval policy, it becomes read-only.
Copy an approval policy	Copies an approval policy.
Deactivate an approval policy	Deactivates an approval policy. You can also delete all existing entitlements associated with the approval policy.
Delete an approval policy	Deletes an approval policy that is in draft state. Active approval policies are read-only.

You can use the workflows in the **ASD Custom Resource** subdirectory to copy and delete advanced services custom resources.

Table 3-2. ASD Custom Resource

Workflow	Description
Create custom resource	Creates an Advanced Service custom resource.
Delete custom resource	Removes an Advanced Service custom resource.

You can use the workflows in the **ASD Resource Action** subdirectory to create and manage advanced services resource actions.

Table 3-3. ASD Resource Action

Workflow	Description
Clone resource action	Creates a copy of an existing Advanced Service resource action.
Create resource action	Creates an Advanced Services resource action.
Delete resource action	Deletes an Advanced Services resource action.
Publish resource action	Publishes an Advanced Services resource action.
Unpublish resource action	Unpublishes an Advanced Services resource action.

You can use the workflows in the **ASD Service Blueprints** subdirectory to create and manage advanced services service blueprints.

Table 3-4. ASD Service Blueprints

Workflow	Description
Clone a service blueprint	Creates a copy of an Advanced Services service blueprint.
Create a service blueprint	Creates an Advanced Services service blueprint.
Delete a service blueprint	Deletes an Advanced Services service blueprint.
Publish a service blueprint	Publishes an Advanced Services service blueprint.
Unpublish a service blueprint	Unpublishes an Advanced Services service blueprint.

You can use the workflows in the **Business Groups** subdirectory to create and manage business groups.

Table 3-5. Business Groups

Workflow	Description
Add a custom property	Adds a custom property to a business group.
Create a business group	Creates a business group.
Delete a business group	Deletes a business group.
Delete a custom property	Removes a custom property from a business group.
Update a business group	Updates details for a business group, such as default machine prefix, active directory containers, and user roles.
Update a custom property	Updates a custom property for a business group.

You can use the workflows in the **Catalog Items** subdirectory to manage catalog items.

Table 3-6. Catalog Item

Workflow	Description
Activate a catalog item	Activates a catalog item. You must activate and assign a catalog item to a service before users can request it.
Assign a catalog item to a service	Assigns a catalog item to a service. You must activate and assign a catalog item to a service before users can request it.
Deactivate a catalog item	Deactivates a catalog item and removes it from the service catalog so that users cannot request it.

You can use the workflows in the **Entitlements** subdirectory to create and manage entitlements.

Table 3-7. Entitlements

Workflow	Description
Activate an entitlement	Activates an entitlement.
Assign catalog items to an entitlement	Assigns one or more catalog items to an entitlement. You can also use this workflow to assign an approval policy.
Assign resource actions to an entitlement	Assigns one or more resource actions to an entitlement. You can also use this workflow to assign an approval policy.

Table 3-7. Entitlements (Continued)

Workflow	Description
Assign immediate actions to an entitlement	Assigns one or more immediate actions to an entitlement. The immediate actions do not create requests.
Assign services to an entitlement	Assigns one or more services to an entitlement. You can also use this workflow to assign an approval policy.
Assign users and groups to an entitlement	Assigns one or more users or groups to an entitlement.
Create an entitlement	Creates an entitlement.
Deactivate an entitlement	Deactivates an entitlement.

You can use the workflows in the **Services** subdirectory to manage services.

Table 3-8. Service

Workflow	Description
Activate a service	Activates a service.
Assign catalog items to a service	Assigns one or more catalog items to a service.
Copy a service	Copies a service.
Create a service	Creates a service.
Deactivate a service	Deactivates a service.
Delete a service	Deletes a service.

You can use the workflows in the **Tenants** subdirectory to create and manage tenants.

Table 3-9. Tenants

Workflow	Description
Add administrators	Adds one or more tenant administrators and infrastructure administrators to a tenant.
Add an identity store to a tenant	Adds an identity store to a tenant of a vCloud Automation Center host. You can run this workflow only if you are a system administrator configuring a tenant.
Add an identity store to a vCAC host	Adds an identity store to a tenant that is configured as a vCloud Automation Center host. You can run this workflow only if you are a tenant administrator configuring an identity store for your tenant.
Create a tenant	Creates a tenant. You must select the vCloud Automation Center host added with the system administrator credentials.
Delete an identity store from a tenant	Deletes an identity store from a tenant of a vCloud Automation Center host. You can run this workflow only if you are a system administrator configuring a tenant.
Delete an identity store from a vCAC host	Deletes an identity store from a tenant that is configured as a vCloud Automation Center host. You can run this workflow only if you are a tenant administrator configuring identity stores for your tenant.

Table 3-9. Tenants (Continued)

Workflow	Description
Delete a tenant	Deletes a tenant.
Remove administrators	Removes one or more tenant administrators and infrastructure administrators from a tenant.
Update an identity store for a tenant	Updates an existing identity store for a tenant of a vCloud Automation Center host. You can run this workflow only if you are a system administrator configuring a tenant.
Update an identity store for a vCAC host	Updates an identity store for a tenant that is configured as a vCloud Automation Center host. You can run this workflow only if you are a tenant administrator configuring identity stores for your tenant.
Update a tenant	Updates the name, description, and contact email address of an existing tenant.

Using the vCloud Automation Center Plug-In Infrastructure Administration Workflows

You can use the infrastructure administration workflows to provision virtual machines and run basic or CRUD operations. You use the extensibility package to customize vCloud Automation Center with the ability to call vCenter Orchestrator workflows either as part of the provisioning process, or by using custom operation menus.

You can find the infrastructure administration workflows on the **Workflows** view of the Orchestrator client, in the **Infrastructure Administration** subdirectory of the plug-in library.

You can use the infrastructure administration workflows to provision virtual machines and run basic or CRUD operations.

Table 3-10. Infrastructure Administration

Workflow Name	Description
Await virtual machine state change	<p>Awaits a state change for a set of virtual machines. If all virtual machines are in the success state, a trigger is called and the workflow ends successfully. If any of the specified virtual machines gets into the fail state, or does not exist, the workflow fails. You must enter the success and fail states selecting from the following options:</p> <ul style="list-style-type: none"> ■ Requested ■ AwaitingApproval ■ RegisterMachine ■ BuildingMachine ■ AddingDisks ■ MachineProvisioned ■ MachineActivated ■ InstallTools (VMware only) ■ On ■ Off ■ TurningOn ■ TurningOff ■ ShuttingDown ■ Suspending ■ Resetting ■ Rebooting ■ Expired ■ DeactivateMachine ■ UnprovisionMachine ■ Disposing ■ Finalized
Create a vCAC model entity	Creates and persists an entity for a specified vCloud Automation Center model.
Delete a vCAC model entity	Deletes a specified vCloud Automation Center model entity.
Invoke a post-provisioning action	Invokes a specified post-provisioning action on a virtual machine.
Provision a virtual machine from a blueprint (Deprecated)	This workflow is deprecated. You must user either the Request a catalog item workflow, or the Request a catalog item on behalf of a user workflow. See <i>Using the vCloud Automation Center Plug-In Requests Workflows</i> .
Read a vCAC entity by custom filter	Reads a list of vCloud Automation Center entities by using a custom filter. If you do not specify a filter, all entities are returned as a result.
Read a vCAC entity by system query	Reads a list of vCloud Automation Center entities by using OData system filters. The system filters apply to the OData URI convention.

Table 3-10. Infrastructure Administration (Continued)

Workflow Name	Description
Read a vCAC model entity	Reads a vCloud Automation Center model entity by its ID.
Update a vCAC model entity	Updates a vCloud Automation Center model entity by its ID.

You use the workflows in the **Extensibility** subdirectory to customize vCloud Automation Center with the ability to call vCenter Orchestrator workflows either as part of the provisioning process, or by custom operation menus.

Table 3-11. Extensibility

Workflow Name	Description
Install vCO customization	Installs an Orchestrator customization including customized state change workflow stubs and Orchestrator menu operation workflows.
Uninstall vCO customization	Uninstalls an Orchestrator customization including customized state change workflow stubs and Orchestrator menu operation workflows.
Import vCAC Virtual Machine (deprecated Register vCAC Virtual Machine)	Imports a vCloud Automation Center virtual machine, maps it to a blueprint, and adds it to the catalog.
Import vCenter Virtual Machine (deprecated Register vCenter Virtual Machine)	Imports a vSphere virtual machine, maps it to a blueprint, and adds it to the catalog.
Unregister Virtual Machine	Removes a virtual machine from the vCloud Automation Center inventory.
Assign a menu operation to a blueprint and its virtual machines	Adds or updates a menu operation on virtual machines.
Assign a menu operation to virtual machines	Updates a vCloud Automation Center model entity by its ID.
Assign a state change workflow to a blueprint and its virtual machines	Adds or updates a state change workflow on several blueprints and their virtual machines.
Customize a menu operation	Updates a vCloud Automation Center operation menu.
Remove a menu operation from a blueprint and its virtual machines	Removes a menu operation from a blueprint and its virtual machines.
Remove a state change workflow from a blueprint and its virtual machines	Removes a state change workflow from a blueprint and its virtual machines.

Create a vCloud Automation Center Model Entity

You can run a workflow to create a simple or complex vCloud Automation Center entity, such as a blueprint reference to a business group.

Procedure

- 1 From the drop-down menu in the Orchestrator client, select **Run** or **Design**.
- 2 Click the **Workflows** view.
- 3 Select **Library > vCloud Automation Center > Infrastructure Administration**.

- 4 Right-click the **Create a vCAC model entity** workflow and select **Start workflow**.
- 5 Select a vCloud Automation Center host.
- 6 Enter the name of the model in the **Model name** text box.
- 7 Enter the name of the entity set, in the **Entity set name** text box.

You use scripting or a REST API to set the Simple properties, Links to complex properties, and HTTP headers properties.

- 8 Click **Submit** to run the workflow.

Read a vCloud Automation Center Model Entity

You can run a workflow to read a vCloud Automation Center model entity.

Procedure

- 1 From the drop-down menu in the Orchestrator client, select **Run** or **Design**.
- 2 Click the **Workflows** view.
- 3 Select **Library > vCloud Automation Center > Infrastructure Administration**.
- 4 Right-click **Read a vCAC model entity** and select **Start workflow**.
- 5 Select a vCloud Automation Center host.
- 6 Enter the name of the model in the **Model name** text box.
- 7 Enter the name of the entity set, in the **Entity set name** text box.
You use scripting or a REST API to set the HTTP headers property.
- 8 Click **Submit** to run the workflow.

Using the vCloud Automation Center Plug-In Requests Workflows

You can use the requests workflows to request catalog items and resource actions, and to complete or cancel work items.

You can access these workflows from the **Workflows** view of the Orchestrator client, in the **Requests** subdirectory of the plug-in library.

Workflow	Description
Cancel a work item	Cancels an active work item. You can use this workflow only if you are a system administrator.
Complete a work item	Completes a work item.
Request a catalog item	Requests a catalog item for the user running the workflow.
Request a catalog item on behalf of a user	Sends a request for a catalog item on behalf of a user. You can use this workflow only for catalog items entitled to both you and the user on behalf of whom you are sending the request.

Workflow	Description
Request a resource action	Requests a resource action for a catalog item owned by the user running the workflow.
Request a resource action on behalf of a user	Sends a request for a resource action on behalf of a user. You can use this workflow only for resource actions entitled to both you and the user on behalf of whom you are sending the request.
Wait for catalog item request	Waits for a catalog item request to complete.
Wait for resource action request	Waits for a resource action request to complete.
Wait for work item	Waits for a work item to complete.

Using the vCloud Automation Center Plug-In Sample Workflows

You can use the sample workflows as examples, or as starting points for creating your own custom workflows.

You can find these workflows on the **Workflows** view of the Orchestrator client, in the **Sample** subdirectory of the plug-in library.

Workflow Name	Description
Create a permission	Provides a sample script that interacts with the authorization client and permission service to create a new permission in vCloud Automation Center.
Create a tenant	Creates a tenant with the same vCAC host and Identity Store configuration as the default tenant. To run this workflow, select the vCAC host added with system administrator credentials. You can change the Identity Store settings before running the workflow.
List catalog items	Returns a list of catalog items for the selected tenant.

Access the vCloud Automation Center Plug-In API

Orchestrator provides an API Explorer to allow you to search the vCloud Automation Center plug-in API and see the documentation for JavaScript objects that you can use in scripted elements.

For updated vCloud Automation Center API documentation, see <https://www.vmware.com/support/pubs/vcac-pubs.html>.

Procedure

- 1 Log in to the Orchestrator client as an administrator.
- 2 Select **Tools > API Explorer**.
- 3 Double-click the **vCAC** and **VCACCAFE** modules in the left pane to expand the hierarchical list of vCloud Automation Center plug-in API objects.

What to do next

You can copy code from API elements and paste it into scripting boxes. For more information about API scripting, see *Developing with VMware vCenter Orchestrator*.

Common Tasks Example Scripts

You can cut, paste, and edit the JavaScript examples, or use them as samples to help you learn to develop your own scripts for common vCloud Automation Center tasks.

For more information about scripting in vCenter Orchestrator, see *Developing with VMware vCenter Orchestrator*.

Example: Create a vCloud Automation Center Advanced Service Blueprint

This example script performs the following actions:

- 1 Sets the vCenter Orchestrator workflow used to build the service blueprint.
- 2 Generates the content for the service blueprint based on the workflow.
- 3 Creates the service blueprint entity.
- 4 Publishes the service blueprint.

Table 3-12. Input Variables

Variable	Type
host	vCACCAFE:VCACHost

```
//ID of the workflow used to create the service blueprint
var workflowId = "44e42047-2fa0-4e4a-ba0c-12086540b28b";

var name = "MyBlueprint"
var description = "Blueprint description";
var workflowClient = host.createAdvancedDesignerClient().getAdvancedDesignerWorkflowService();

//Generate a service blueprint based on the workflow ID
var blueprint = workflowClient.generateServiceBlueprintByWorkflowId(workflowId);
blueprint.setTenant(host.tenant);
blueprint.setName(name);
blueprint.setDescription(description);

//Create the service blueprint
var blueprintService =
host.createAdvancedDesignerClient().getAdvancedDesignerServiceBlueprintService();
var uri = blueprintService.createServiceBlueprint(host.tenant , blueprint);

//Publish the service blueprint
var createdBlueprint = blueprintService.getServiceBlueprintByUri(uri);
blueprintService.updateServiceBlueprintStatus(host.tenant, createdBlueprint.getId(),
vCACCAFEDesignerPublishStatus.PUBLISHED);
```

Example: Create a vCloud Automation Center Approval Policy

This example script performs the following actions:

- 1 Gets the approval policy type.
- 2 Sets the user and group whose approval is required.
- 3 Sets the approval levels.
- 4 Defines the pre-provisioning approval phase.
- 5 Defines the post-provisioning approval phase.
- 6 Defines the approval policy specifications such as name, description, and type.
- 7 Creates the approval policy.
- 8 Publishes the approval policy. Once an approval policy is published, it becomes read-only.

Table 3-13. Input Variables

Variable	Type
host	vCACCAFE:VCAHost

```
// Get the type of approval policy by ID
var typeService = host.createApprovalClient().getApprovalApprovalPolicyTypeService();
var type = typeService.getApprovalPolicyType("com.vmware.cafe.catalog.request");

// Set the user and group required to complete the approval
var user = new vCACCAFEApprovalPrincipal();
user.setValue("user@domain.com");
user.setType(vCACCAFEApprovalPrincipalType.USER);

var group = new vCACCAFEApprovalPrincipal();
group.setValue("group@domain.com");
group.setType(vCACCAFEApprovalPrincipalType.GROUP);

// Set the level of the approval
var level = new vCACCAFEApprovalLevel();
level.setName("IT Approval Level");
level.setDescription("IT Approval Level description");
level.setApprovalMode(vCACCAFEApprovalMode.ALL);
System.getModule("com.vmware.library.vcaccafe.util").addElementToList(level, "getApprovers", user);
System.getModule("com.vmware.library.vcaccafe.util").addElementToList(level, "getApprovers", group);
level.setLevelNumber(1);

// Set pre-provisioning phase type and the phase of the approval
var phase1Type = new vCACCAFEApprovalPhaseType();
phase1Type.setId("com.vmware.cafe.catalog.request.pre");
phase1Type.setName("Pre-Provisioning type");
phase1Type.setDescription("Pre-Provisioning type description");
phase1Type.setPhaseOrder(1);

var phase1 = new vCACCAFEPhase();
```

```

phase1.setName("Pre-Provisioning");
phase1.setDescription("Pre provisioning phase");
phase1.setPhasetype(phase1Type);
System.getModule("com.vmware.library.vcaccafe.util").addElementToList(phase1, "getLevels", level);

// Set post-provisioning phase type and the phase of the approval
var phase2Type = new vCACCAFEApprovalPhaseType();
phase2Type.setId("com.vmware.cafe.catalog.request.post");
phase2Type.setName("Post-Provisioning type");
phase2Type.setDescription("Post-Provisioning type description");
phase2Type.setPhaseOrder(1);

var phase2 = new vCACCAFEPhase();
phase2.setName("Post-Provisioning");
phase2.setDescription("Post provisioning phase");
phase2.setPhasetype(phase2Type);
System.getModule("com.vmware.library.vcaccafe.util").addElementToList(phase2, "getLevels", level);

// Create the approval policy specifications
var spec = new vCACCAFEApprovalPolicy();
spec.setName("New Policy");
spec.setDescription("New Policy description");
spec.setPolicyType(type);
System.getModule("com.vmware.library.vcaccafe.util").addElementToList(spec, "getPhases", phase1);
System.getModule("com.vmware.library.vcaccafe.util").addElementToList(spec, "getPhases", phase2);

// Create the approval policy
var approvalPolicyService = host.createApprovalClient().getApprovalApprovalPolicyService();
var approvalPolicy = approvalPolicyService.createPolicy(spec);

// Publish the approval policy
approvalPolicy.setState(vCACCAFEApprovalPolicyState.PUBLISHED);
approvalPolicy = approvalPolicyService.update(approvalPolicy);
System.log("New approval policy id: " + approvalPolicy.getId());

```

CRUD Infrastructure Administration Tasks Example Scripts

You can cut, paste, and edit the JavaScript examples to write scripts for CRUD vCloud Automation Center tasks.

For more information about scripting in vCenter Orchestrator, see *Developing with VMware vCenter Orchestrator*.

Example: Create a vCloud Automation Center Model Entity

This example script performs the following actions:

- 1 Defines the model name and the entity set name.
- 2 Defines the properties of the host prefix.
- 3 Saves the host prefix entity.

- 4 Defines the properties of the provisioning group.
- 5 Defines the provisioning group as a link.
- 6 Saves the provisioning group entity, by linking it with the host name prefix.

Table 3-14. Input Variables

Variable	Type
host	vCAC:VcacHost

```

var modelName = 'ManagementModelEntities.svc';
var entitySetName = 'HostNamePrefixes';
var links = null;
var headers = null;
//Create properties for prefix entity
var prefixInputProperties = {
    MachinePrefix:'test-prefix',
    NextMachineNo:1,
    MachineNumberLength:3
};
//Save the prefix
var prefixEntity = vCACEntityManager
    .createModelEntity(host.id, modelName, entitySetName, prefixInputProperties, links, headers);
entitySetName = 'ProvisioningGroups';
//Create properties for the provisioning group entity
inputProperties = {
    GroupName:'TestGroupName',
    GroupDescription:'This group was generated with a vCO workflow',
    AdministratorEmail:'test@test.com',
    AdContainer:'AD',
    IsTestGroup:false,
    Flags:2,
    GroupType:1};
//Add a reference to the newly created prefix entity
links = {
    HostNamePrefix:prefixEntity
};
//Save the provisioning group
var entity = vCACEntityManager.createModelEntity(host.id, modelName, entitySetName, inputProperties,
links, headers);

```

Example: Update a vCloud Automation Center Model Entity

This example script performs the following actions:

- 1 Gets the host ID from the provided entity.
- 2 Gets the model name from the provided entity.
- 3 Gets the entity set name from the provided entity.
- 4 Gets the entity ID from the provided entity.
- 5 Defines a set of properties that will be updated.

6 Starts the action responsible for updating the entity.

Table 3-15. Input Variables

Variable	Type
entity	vCAC:Entity
updatedDescription	String

```

var hostId = entity.hostId;
var modelName = entity.modelName;
var entitySetName = entity.entitySetName;
var entityIdString = entity.keyString;
var links = null;
var headers = null;
var updateProperties = new Properties();
updateProperties.put("UserNameDescription", updatedDescription);
//Update the user description
System.getModule("com.vmware.library.vcac")
    .updateVCACEntity(hostId, modelName, entitySetName, entityIdString, updateProperties, links,
headers);

```

Example: Read a vCloud Automation Center Model Entity

This example script performs the following actions:

- 1 Defines the model name and the entity set name.
- 2 Defines the blueprint ID with a property object.
- 3 Reads the entity.

Table 3-16. Input Variables

Variable	Type
host	vCAC:VcacHost
blueprintID	String

```

var modelName = 'ManagementModelEntities.svc';
var entitySetName = 'VirtualMachineTemplates';
var links = null;
var headers = null;
//Create properties for the prefix entity
var blueprintId = {
    VirtualMachineTemplateID:blueprintId,
};
//Read the blueprint
var entity = vCACEntityManager
    .readModelEntity(host.id, modelName, entitySetName, blueprintId, headers);

```


Example: Delete a vCloud Automation Center Model Entity

This example script performs the following actions:

- 1 Gets the host ID from the provided entity.
- 2 Gets the model name from the provided entity.
- 3 Gets the entity set name from the provided entity.
- 4 Gets the entity ID from the provided entity.
- 5 Starts the action responsible for deleting the entity.

Table 3-17. Input Variables

Variable	Type
entity	vCAC:Entity

```
var hostId = entity.hostId;
var modelName = entity.modelName;
var entitySetName = entity.entitySetName;
var entityKeyString = entity.keyString;
var headers = null;
//Delete the entity
System.getModule("com.vmware.library.vcac")
    .deleteVCACEntity(hostId, modelName, entitySetName, entityKeyString, headers);
```

Example: Read a vCloud Automation Center Entity by Custom Filter

This example script performs the following actions:

- 1 Defines the model name and the entity set name.
- 2 Defines the properties by which the entities are filtered.
- 3 Reads a list of entities.

Table 3-18. Input Variables

Variable	Type
host	vCAC:VcacHost
templateName	String

```
var modelName = 'ManagementModelEntities.svc';
var entitySetName = 'VirtualMachineTemplates';
var headers = null;
//Create properties for prefix entity
var properties = {
    VirtualMachineTemplateName:templateName,
```

```
};
//Read a list of entities
var entities = vCACEntityManager
    .readModelEntitiesByCustomFilter(host.id, modelName, entitySetName, properties, headers);
```

Example: Read a vCloud Automation Center Entity by System Query

This example script performs the following actions:

- 1 Defines the model name and the entity set name.
- 2 Defines the system queries by which the entities are filtered and selects the top ten results of all virtual machines, filtered by the machine state and component flag.
- 3 Reads a list of entities.

Table 3-19. Input Variables

Variable	Type
host	vCAC:VcacHost

```
var modelName = 'ManagementModelEntities.svc';
var entitySetName = 'VirtualMachines';
var filter = "VirtualMachineState eq 'Off' and IsComponent eq true";
var orderBy = 'VirtualMachineName asc';
var top = 10; {
var skip = 0;
var headers = null;
var select = null;
var entities = vCACEntityManager
    readModelEntitiesBySystemQuery(host.id, modelName, entitySetName, filter, orderBy, select, top,
    skip, headers);
```

Finding vCloud Automation Center Entities Example Scripts

You can cut, paste, and edit the JavaScript examples to write scripts for finding vCloud Automation Center entities by using the vCACCAFEEntitiesFinder scripting utility object.

For more information about scripting in vCenter Orchestrator, see *Developing with VMware vCenter Orchestrator*.

Example: Find Catalog Resources Filtered by Name

Table 3-20. Input Variables

Variable	Type
host	vCAC:VcacHost

You can use one the following examples:

- This example script gets all catalog resources for the target host matching the query of *name_of_the_resource* by name and description.

```
var items = vCACCAFEEntitiesFinder.findCatalogResources(host, "name_of_the_resource");
```

- This example script performs the following actions:
 - a Gets the Consumer Resource service and invokes the get method passing as a Pageable parameter an instance of the vCACCAFEPageOdataRequest object.
 - b Creates the vCACCAFEPageOdataRequest object by providing an OData query as a single filter of the name attribute matching the *name_of_the_resource* string.

```
var service = host.createCatalogClient().getCatalogConsumerResourceService();

var filter = new Array();
filter[0] = vCACCAFEFILTERParam.equal("name", vCACCAFEFILTERParam.string("name_of_the_resource"));
var query = vCACCAFEOdataQuery.query().addFilter(filter);

var items = service.getResourcesList(new vCACCAFEPageOdataRequest(query));
```

Example: Find Catalog Resources Filtered by Owner

This example script performs the following actions:

- 1 Gets the Consumer Resource service and invokes the get method passing as a Pageable parameter an instance of the vCACCAFEPageOdataRequest object.
- 2 Creates the vCACCAFEPageOdataRequest object by providing an OData query as a single filter of the owner/ref attribute matching the *user@domain.com* string.

The owners/ref attribute is a composition based on the internal structure and fields of the catalog resources. The vCACCAFECatalogResource entity has the owners attribute, which is a collection of vCACCAFECatalogPrincipal entities. The vCACCAFECatalogPrincipal entity has the ref property, which is a string representation of the principal id of the user.

```
var filter = new Array();
filter[0] = vCACCAFEFILTERParam.substringOf("owners/ref",
vCACCAFEFILTERParam.string("user@domain.com"));
var query = vCACCAFEOdataQuery.query().addFilter(filter);

var items = service.getResourcesList(new vCACCAFEPageOdataRequest(query));
```

Example: Find Catalog Resources Filtered by Name and Owner

This example script combines the OData queries from the previous two examples into a single one condition by using the `vCACCAFEFilterParam.and(array of conditions)` logic operator.

```
var conditions = new Array();
conditions[0] = vCACCAFEFilterParam.equal("name",
vCACCAFEFilterParam.string("name_of_the_resource_here"));
conditions[1] = vCACCAFEFilterParam.substringOf("owners/ref",
vCACCAFEFilterParam.string("user@domain.com"));

var filter = new Array();
filter[0] = vCACCAFEFilterParam.and(conditions);
var query = vCACCAFE0dataQuery.query().addFilter(filter);

var items = service.getResourcesList(new vCACCAFEPage0dataRequest(query));
```

You can define other conditions by using different logic operators such as `vCACCAFEFilterParam.group(array of parameters)`, `vCACCAFEFilterParam.not(parameter)`, `vCACCAFEFilterParam.startsWith(id, string)`, `vCACCAFEFilterParam.endsWith(id, string)`, `vCACCAFEFilterParam.greaterThan(id, number)`, `vCACCAFEFilterParam.lessThan(id, number)`, and so on.

Get a Resource Provisioned by vCloud Automation Center Example Script

You can cut, paste, and edit the JavaScript example to write scripts for retrieving the actual entities of vCloud Automation Center provisioned resources.

The `CatalogResource` type represents the provisioned resources in vCloud Automation Center. This type has an attribute of `ProviderBinding` type which represents the relation between the catalog resource and its provider with the following attributes:

- `bindingId` - represents the identifier of the entity which is unique for the provider
- `providerRef` - identifies the catalog provider which corresponds directly to a service registered in the vCloud Automation Center component registry

For more information about scripting in vCenter Orchestrator, see *Developing with VMware vCenter Orchestrator*.

Example: Get a Virtual Machine Provisioned as a vCloud Automation Center Catalog Resource

This example uses a vCloud Automation Center host and its `iaas` host as input parameters and, for a provided resource id, returns the corresponding `iaas` virtual machine. The scripting code takes only catalog resources of `Virtual Machine` type that are provisioned by the `iaas-service` provider.

Table 3-21. Input Variables

Variable	Type
vcacHost	vCACCAFE:VCACHost
iaasHost	vCAC:VCACHost

```
// Id of the catalog resource (or vCACCAFECatalogResource_instance.getId())
var resourceId = "c222629c-6f90-4458-8c92-8ece0ba06173";

var resource = vCACCAFEEntitiesFinder.getCatalogResource(vcacHost, resourceId);

var resourceType = resource.getResourceTypeRef().getLabel();
System.log("resource type: " + resourceType);

var providerBinding = resource.getProviderBinding();

var bindingId = providerBinding.getBindingId();
System.log("provider binding id: " + bindingId);

var provider = providerBinding.getProviderRef();
System.log("provider id: " + provider.getId());
System.log("provider name: " + provider.getLabel());

if ((resourceType == "Virtual Machine") && (provider.getLabel() == "iaas-service")) {
    System.log("It is an IaaS VM!");

    // IaaS virtual machine
    var vm = Server.findForType("vCAC:VirtualMachine", bindingId);
    System.log("IaaS VM id: " + vm.virtualMachineID);
    System.log("IaaS VM name: " + vm.displayName);

    // IaaS Entity
    var entity = System.getModule("com.vmware.library.vcac").getVirtualMachineEntityFromId(iaasHost,
bindingId);
    System.log("IaaS entity id: " + entity.keyString);
}
}
```

Transfer Content Between vCloud Automation Center Hosts Example Script

You can cut, paste, and edit the JavaScript example to write scripts for content transfer between vCloud Automation Center hosts by using the Content service.

For more information about scripting in vCenter Orchestrator, see *Developing with VMware vCenter Orchestrator*.

Example: Transfer Content from One vCloud Automation Center Host to Another vCloud Automation Center Host

This example script performs the following actions:

- 1 Defines the source host that is based on a tenant administrator role.
- 2 Exports the content of the source host.
- 3 Defines the target host that is based on a tenant administrator role.
- 4 Imports the content to the target source.

Table 3-22. Input Variables

Variable	Type
sourceHost	string
targetHost	string

```

var CONTENT_MANAGEMENT_SERVICE = "com.vmware.csp.core.content.service.api";
var content;

System.log("Source host: " + sourceHost.url);
System.log("Source tenant: " + sourceHost.tenant);

var contentClientExport = sourceHost.createRestClient(CONTENT_MANAGEMENT_SERVICE);
System.log("Exporting...");
content = contentClientExport.getFile("content");
System.log("Export successful");

System.log("Target host: " + targetHost.url);
System.log("Target tenant: " + targetHost.tenant);

var contentClientImport = targetHost.createRestClient(CONTENT_MANAGEMENT_SERVICE);
System.log("Importing...");
contentClientImport.postFile("content", content);
System.log("Import successful");

```