Using and Managing vRealize Automation Service Broker

9 SEPTEMBER 2021

vRealize Automation 8.5

vmware®

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

https://docs.vmware.com/

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

Contents

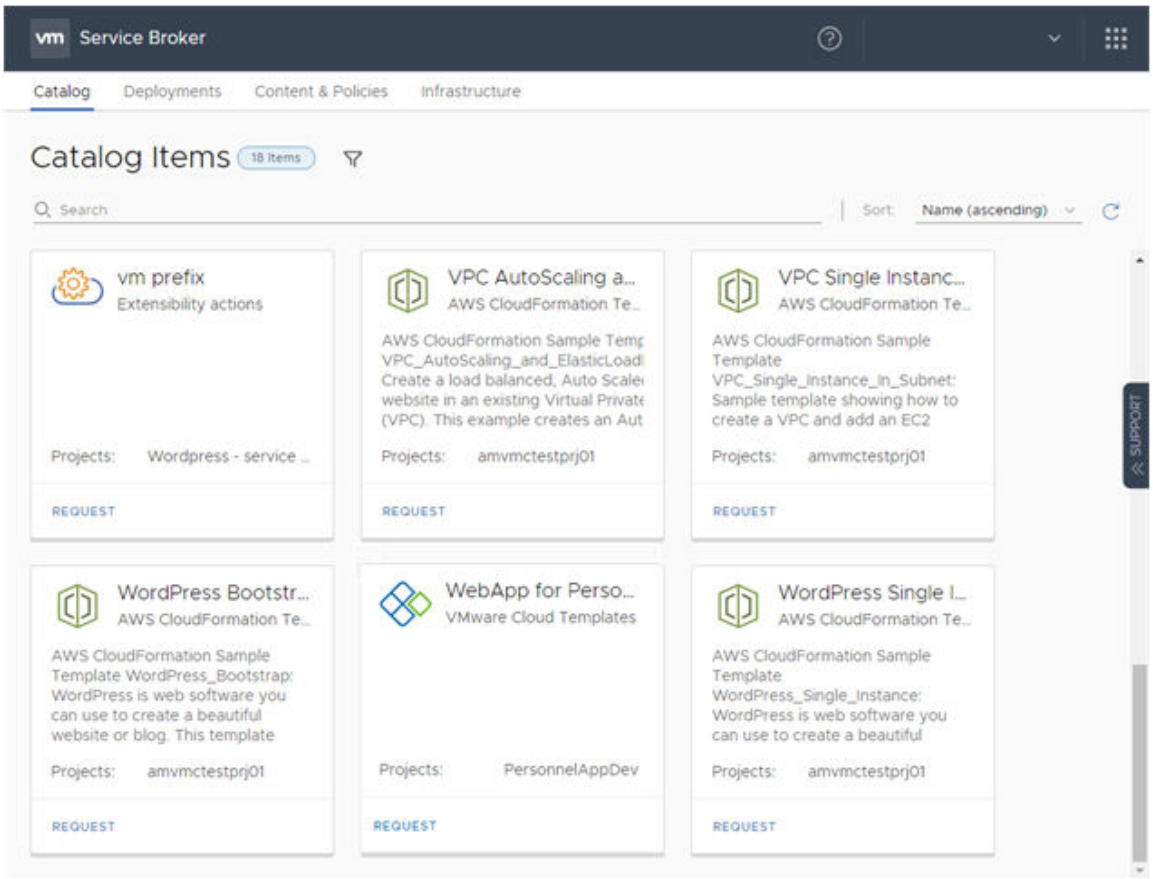
What is vRealize Automation Service Broker

1

The vRealize Automation Service Broker provides a single point where you can request and manage catalog items.

As a cloud administrator, you create catalog items by importing released vRealize Automation Cloud Assembly cloud templates and Amazon Web Services CloudFormation templates that your users can deploy to your cloud vendor regions or datastores.

As a user, you can request and monitor the provisioning process. After deployment, you manage the deployed catalog items throughout the deployment lifecycle.



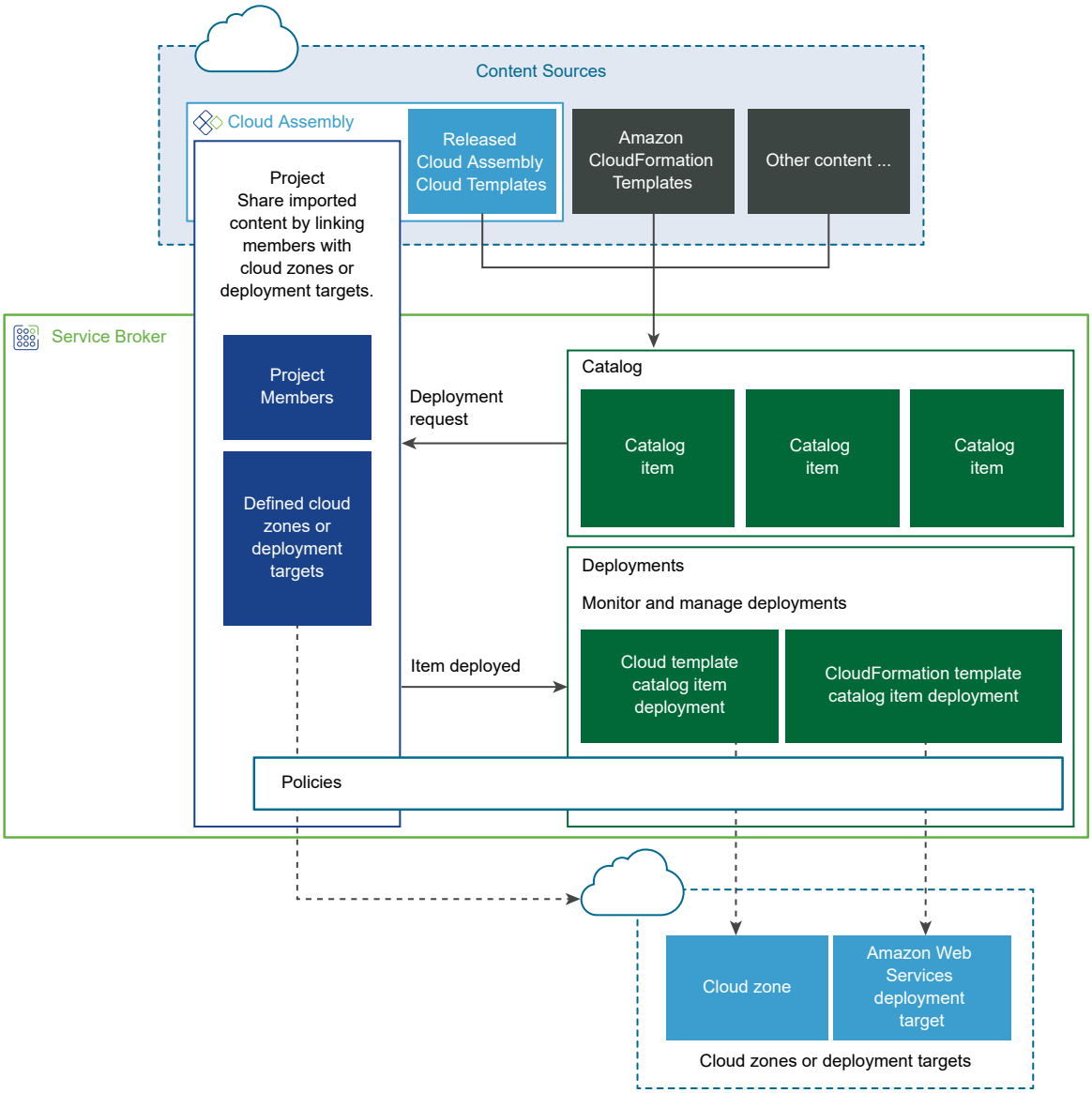
This chapter includes the following topics:

- How does Service Broker work

How does Service Broker work

The Service Broker is the simplified user interface that cloud administrators make available to users when the administrator's teams do not need full access to developing and building and the templates.

You use Service Broker to deploy templates to cloud regions or datastores associated with projects.



To provide the templates, the cloud administrator configures content sources. The content sources can include Cloud Assembly templates and Amazon CloudFormation templates. The imported templates become catalog items.

- The content sources are entitled to projects. Projects link a set of users with one or more target cloud zone regions or datastores.

- For example, UserA is a member of ProjectA and ProjectB, but not ProjectC. She sees only the imported templates that were entitled to ProjectA and ProjectB.

When users requests a catalog item, where it deployed depends on the project selected. Projects might have one or more cloud zones.

- If UserA and UserB are members of ProjectA, they see the imported templates as catalog items. And at deployment time they can deploy to ProjectA, which determines which cloud regions or datastores the catalog item is deployed to.

The availability of the catalog items is determined by project membership. Projects link users, catalog items, and cloud resources where the items are deployed.

After a successful request, your users can then manage their deployments by running actions, including dismiss or delete.

What are the Service Broker user roles

2

Your user role in Service Broker determines what you can see and do. Some roles are defined at the service organization level, and some are specific to Cloud Assembly.

User Roles

User roles are defined for the organization in the vRealize Automation console. There are two types of roles, organization roles and service roles.

The organization roles are global and apply to all services in the organization. A user is assigned an Organization owner or Organization Member role.

For more information about the organization, service, and custom roles, start with the cloud user roles.

The Service Broker service roles, which are service-specific permissions, are also assigned at the organization level in the console.

Service Broker Service Roles

The Service Broker service roles determine what you can see and do in Service Broker. These service roles are defined in the console by an organization owner.

Table 2-1. Service Broker Service Role Descriptions

Role	Description
Service Broker Administrator	Must have read and write access to the entire user interface and API resources. This is the only user role that can perform all tasks, including creating a new project and assigning a project administrator.
Service Broker User	Any user who does not have the Service Broker Administrator role. In a Service Broker project, the administrator adds users to projects as project members, administrators, or viewers. The administrator can also add a project administrator.
Service Broker Viewer	A user who has read access to see information but cannot create, update, or delete values. Users with the viewer role can see all the information that is available to the administrator. They cannot take any action unless you make them a project administrator or a project member. If the user is affiliated with a project, they have the permissions related to the role. The project viewer would not extend their permissions the way that the administrator or member role does.

In addition to the service roles, Service Broker has project roles. Any project is available in all of the services.

The project roles are defined in Service Broker and can vary between projects.

In the following tables, which tells you what the different service and project roles can see and do, remember that the service administrators have full permission on all areas of the user interface.

Use the following descriptions of project roles will help you as you decide what permissions to give your users.

- Project administrators leverage the infrastructure that is created by the service administrator to ensure that their project members have the resources they need for their development work.
- Project members work within their projects to design and deploy cloud templates.
- Project viewers are restricted to read-only access.
- Project supervisors are approvers in Service Broker for their projects where an approval policy is defined with a project supervisor approver. To provide the supervisor with context for approvals, consider also granting them the project member or viewer role.

Table 2-2. Service Broker Service Roles and Project Roles

UI Context	Task	Service Broker Administrator	Service Broker Viewer	Service Broker User User must be a project administrator to see and do project-related tasks.			
				Project Administrator	Project Member	Project Viewer	Project Supervisor
Access Service Broker							
Console	In the console, you can see and open Service Broker	Yes	Yes	Yes	Yes	Yes	Yes
Infrastructure							
	See and open the Infrastructure tab	Yes	Yes				
Configure - Projects	Create projects	Yes					
	Update, or delete values from project summary, provisioning, Kubernetes, integrations, and test project configurations.	Yes					
	Add users and groups, and assign roles in projects.	Yes		Yes. Your projects.			
	View projects	Yes	Yes	Yes. Your projects	Yes. Your projects	Yes. Your projects	
Configure - Cloud Zones	Create, update, or delete cloud zones	Yes					
	View cloud zones	Yes	Yes				
Configure - Kubernetes Zones	Create, update, or delete Kubernetes zones	Yes					
	View Kubernetes zones	Yes	Yes				
Connections - Cloud Accounts	Create, update, or delete cloud accounts	Yes					

Table 2-2. Service Broker Service Roles and Project Roles (continued)

UI Context	Task	Service Broker Administrator	Service Broker Viewer	Service Broker User User must be a project administrator to see and do project-related tasks.			
				Project Administrator	Project Member	Project Viewer	Project Supervisor
	View cloud accounts	Yes	Yes				
Connections - Integrations	Create, update, or delete integrations	Yes					
	View integrations	Yes	Yes				
Activity - Requests	Delete deployment request records	Yes					
	View deployment request records	Yes					
Activity - Event Logs	View event logs	Yes					
Content and Policies							
	See and open the Content and Policies tab	Yes	Yes				
Content Sources	Create, update, or delete content sources	Yes					
	View content sources	Yes	Yes				
Content Sharing	Add or remove shared content	Yes					
	View shared content	Yes	Yes				
Content	Customize form and configure item	Yes					
	View content	Yes	Yes				
Policies - Definitions	Create, update, or delete policy definitions	Yes					
	View policy definitions	Yes	Yes				
Policies - Enforcement	View enforcement log	Yes	Yes				

Table 2-2. Service Broker Service Roles and Project Roles (continued)

UI Context	Task	Service Broker Administrator	Service Broker Viewer	Service Broker User User must be a project administrator to see and do project-related tasks.			
				Project Administrator	Project Member	Project Viewer	Project Supervisor
Notifications - Email Server	Configure an email server	Yes					
Catalog							
	See and open the Catalog tab	Yes	Yes	Yes	Yes	Yes	Yes
	View available catalog items	Yes	Yes	Yes. Your projects	Yes. Your projects	Yes. Your projects	
	Request a catalog item	Yes		Yes. Your projects	Yes. Your projects		
Deployments							
	See and open the Deployments tab	Yes	Yes	Yes.	Yes	Yes	Yes
	View deployments, including deployment details, deployment history, price, monitor, alerts, optimize, and troubleshooting information	Yes	Yes	Yes. Your projects	Yes. Your projects	Yes. Your projects	
	Manage alerts	Yes		Yes. Your projects	Yes. Your projects		
	Run day 2 actions on deployments based on policies	Yes		Yes. Your projects	Yes. Your projects		
Approvals							
	See and open the Approvals tab	Yes	Yes	Yes	Yes	Yes	Yes
	Respond to approval requests	Yes		Yes. Your projects and the policy approver is Project Administrator	Only if you are a named approver	Only if you are a named approver	Yes. Your projects and the policy approver is Project Supervisor

Setting up Service Broker for your organization

3

To fully configure Service Broker, you need to determine your catalog sources and apply governance using projects. As a cloud administrator, you can also apply policies and customize the catalog request form.

As a cloud administrator, you can also apply policies and customize the catalog request form.

This chapter includes the following topics:

Adding Content to the Service Broker Catalog

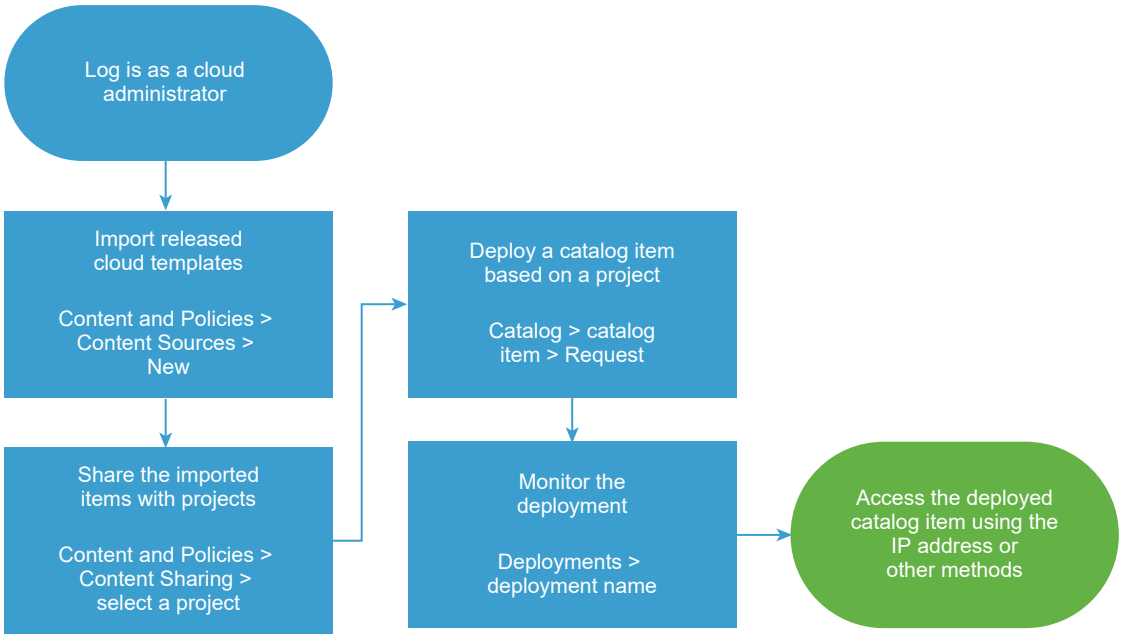
The requirements and process for setting up your Service Broker catalog depends on the content that you are providing to your users.

Each process is provided as an end-to-end procedure. Identify the content that you are providing and add each relevant type. Ensure that the imported content is working properly outside of Service Broker before you add it to the catalog.

After you add the content sources, the templates are refreshed every six hours. Any changes to the templates in your external sources are reflected in the catalog after a refresh.

Add Cloud Assembly cloud templates to the Service Broker catalog

As a cloud administrator, you can make Cloud Assembly cloud templates available in the Service Broker catalog by adding a Cloud Assembly content source and sharing the templates. The cloud templates are the specifications for services or applications that you can deploy to your cloud providers.



After you import the cloud templates, you share them with project members so that they can deploy the templates. At the request time, the cloud template is deployed to cloud zone account region or datastore that supports the cloud template requirements.

Prerequisites

- Verify that the cloud templates that you are importing are deployable and released in Cloud Assembly before you import them. See How to save different versions of a cloud template in *Using and Managing vRealize Automation Cloud Assembly*.

Procedure

1. Import cloud templates from Cloud Assembly.
   a. Select **Content and Policies > Content Sources**.
   b. Click **New**, and then click **VMware Cloud Templates**.
   c. Enter the **Name** for this content source.
   d. Select the **Source project** and then click **Validate**.

      The validation process tests the connection and provides the number of released cloud templates that are associated with the project in Cloud Assembly.

   e. Click **Create and Import**.

   The Content Sources page lists your new source and the number of discovered and imported items.

2 Share the imported items with a project.

   a Select **Content and Policies > Content Sharing**.

   b Select the project that includes the users who should be able to deploy the cloud templates.

   c Click **Add Items** and then select one or more cloud templates to share with the project.

   The list of possible templates includes the cloud templates associated with the current project in Cloud Assembly and any cloud templates for other projects where sharing is enabled.

   You can select all the items imported from a content sources or you can expand the source trees and select individual items.

   d Click **Save**.

   The Content Sharing page lists all the items entitled to the selected project. The cloud templates are also added to the catalog where the project members can request them.

3 Verify that the cloud template is available in the catalog to the members of the selected projects.

   a Click **Catalog**, locate the imported cloud template, and review the projects to ensure that the project you configured is included.

   b Click **Request** and provide any required information.

   If the cloud template has more than one released version, select the version that you want to deploy.

   c Click **Submit**.

   The provisioning process begins and the Deployments tab opens with your current request at the top.

4 Monitor the provisioning process to ensure successful deployment.

   a Click **Deployments** and locate your deployed catalog item.

   b Monitor the card status until it is successful.

Results

The released cloud templates are imported into Service Broker, shared in the catalog, and deployable.

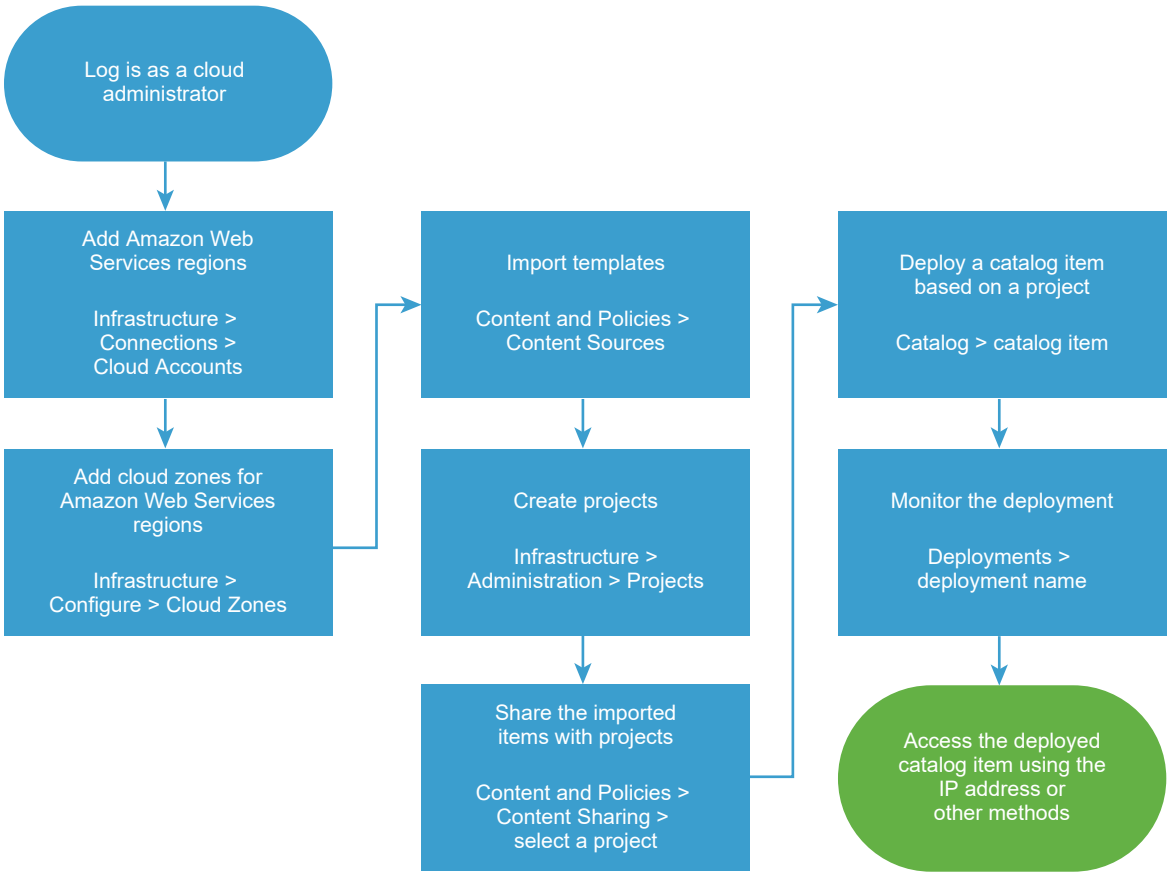
What to do next

- If the deployment fails, click the deployment name and begin troubleshooting. See What can I do if a Service Broker deployment fails. If you are a Cloud Assembly cloud administrator, you can also do more extensive troubleshooting in Cloud Assembly What can I do if a Cloud Assembly deployment fails in *Using and Managing VMware Cloud Assembly*.

- If you want to control how long a deployment can exist, create a lease. See Setting up Service Broker policies.
- To provide more or fewer user inputs at request time, you can create a custom form. See Customize a Service Broker icon and request form.

Add CloudFormation templates to the Service Broker catalog

As a cloud administrator, you can populate the Service Broker catalog with Amazon CloudFormation templates by adding one or more Amazon S3 buckets as content sources and sharing them with project members. The templates are the specifications for the services or applications that you can deploy to Amazon Web Services.



You can only add one bucket as a content source. To add multiple buckets, you create a content source for each bucket.

After you add the templates, you entitle project members to deploy the cloud templates. At the request time, the cloud template is deployed to the cloud account region that you define when you add the content source.

Prerequisites

- Ensure that you know to name of the S3 bucket that contains your CloudFormation templates.
- If you are adding a private bucket, you must know the access key and the secret key.

Procedure

1 To deploy your CloudFormation templates, you must have at least one Amazon Web Services cloud account and select the regions.
   a Select **Infrastructure > Connections > Cloud Accounts**.
   b Click **Add Cloud Account** and then click **Amazon Web Services**.
   c Enter the 20-digit **Access Key ID** and corresponding **Secret Access Key**.
   d To verify the credentials, click **Validate**.
   e Enter an account name.

      Provide a name that you can identify when you share templates with projects.
   f Select one or more regions in this account that you want to deploy templates to.
   g Click **Create**.
2 Define cloud zones for the Amazon Web Services cloud account regions.
   a Select **Infrastructure > Configure > Cloud Zones**, and then click **New Cloud Zone**.
   b Select the **Account/region**, the **Name**, and the **Placement policy**.
   c Click the **Compute** tab and verify or modify the resources that are included in the cloud zone.
   d Click **Create**.
3 Import the templates.
   a Select **Content and Policies > Content Sources**.
   b Click **New**, and then click **AWS CloudFormation Template**.
   c Enter the **Name** for this content source.
   d Add the S3 bucket information.
   e Click **Validate**.

      If the bucket is public, the validation process verifies the name and the number of templates. If the bucket is private, the validation process verifies the name, the keys, and the number of templates.
   f Select the **Deployment Target** Amazon Web Services cloud account and a region.
   g Click **Create and Import**.
4 Add a project so that you can share the templates with project members.
   a In Service Broker, select **Infrastructure > Administration > Projects**, and then click **New Project**.
   b Enter the project information on the **Summary** tab.

c Click the **Users** tab and then click **Add Users**.

To add project users, the individuals or the groups must already be active service organization users.

d If this project supports only CloudFormation templates, ignore the Provisioning tab.

CloudFormation templates are deployed to the target account and region that you defined when you imported the templates. If the project members can deploy other templates or content, you must add the target cloud zones for the content to the project.

e Click **Create**.

The new project is added to your projects. It is also added to your associated Cloud Assembly instance. If the project is for VMware Cloud Templates, you can add cloud zones in Cloud Assembly. If the project is for templates, you do not need to add cloud zones.

5 Share the imported templates with a project.

a Select **Content and Policies > Content Sharing**.

b Select the project that includes the users who should be able to deploy the templates.

c Select one or more Amazon Web Services content sources to share with the project.

d Click **Save**.

Content Sharing page lists all the items entitled to the selected project. The templates are also added to the catalog where the project members can request them.

6 Verify that the template is available in the catalog to the members of the selected projects.

a Click **Catalog**, locate the imported CloudFormation templates, and review the projects to ensure that the project you configured is included.

b Click **Request** and provide any required information.

c Click **Submit**.

The provisioning process begins and the Deployments tab opens with your current request at the top.

7 Monitor the provisioning process to ensure successful deployment.

a Click **Deployments** and locate your deployed catalog item.

b Monitor the card status until it is successful.

Results

The templates are imported into Service Broker and shared in the catalog.

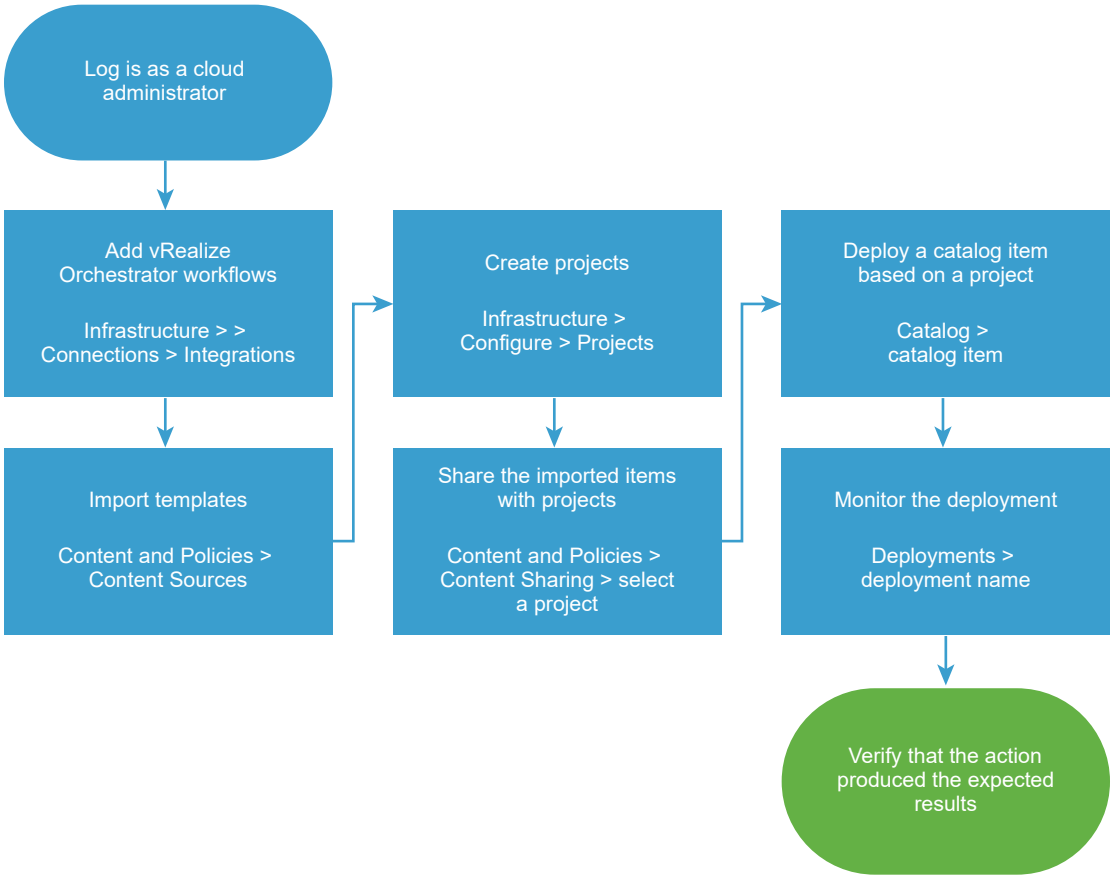
What to do next

- If the deployment fails, click the deployment name and begin troubleshooting. See What can I do if a Service Broker deployment fails. If you are a Cloud Assembly cloud administrator, you can also do more extensive troubleshooting in Cloud Assembly What can I do if a Cloud Assembly deployment fails in *Using and Managing VMware Cloud Assembly*.
- If you want to control how long a deployment can exist, create a lease. See Setting up Service Broker policies.
- To provide more or fewer user inputs at request time, you can create a custom form. See Customize a Service Broker icon and request form.

Add vRealize Orchestrator workflows to the Service Broker catalog

As a cloud administrator, you can add vRealize Orchestrator workflows to the catalog. The workflows are created in vRealize Orchestrator to accomplish a simple or complex task.

In addition to the regular input parameters, the workflows can include composite types as input parameters.



Prerequisites

- Verify that you have vRealize Orchestrator workflows that can perform required tasks. See Managing Workflows.

Procedure

1 If you do not have a vRealize Orchestrator integration configured in Cloud Assembly, you can add the integration in Service Broker.
   a Select **Infrastructure > Connections > Integrations**.
   b Click **Add Integration** and then click **vRealize Orchestrator**.
   c Enter the URL for your vRealize Orchestrator instance.
   d Select or add a **Cloud Proxy**.
   e Enter a user name and password.
   f To validate the credentials and URL, click **Validate**.
   g Enter a name that identifies this instance when you create the content source.
   h Click **Add**.
2 Import the workflow.
   a Select **Content and Policies > Content Sources**.
   b Click **New**, and then click **vRealize Orchestrator Workflow**.
   c Enter the **Name** for this content source so that you can identify it when you share the content.
   d Click Add and select the workflows that you want to make available in Service Broker.
   e Click **Create and Import**.
3 Share the imported workflow with a project.
   a Select **Content and Policies > Content Sharing**.
   b Select the project that includes the users who should be able to deploy the workflows.
   c Click **Add Items** and then select one or more workflows to share with the project members.

      You can select all the items imported from a content source or you can expand the source trees and select individual items.
   d Click **Save**.
4 Verify that the workflow is available in the catalog to members of the selected project.
   a Click **Catalog**, locate the imported workflow, and review the projects to ensure that the project you configured is included.
   b Click **Request** and provide any required information.
   c Click **Submit**.

   The provisioning process begins and the Deployments tab opens with your current request at the top.

5 Monitor the provisioning process to ensure that the workflow runs successfully.

   a Click **Deployments** and locate your deployed request.

   b Monitor the card status until it is successful.

Results

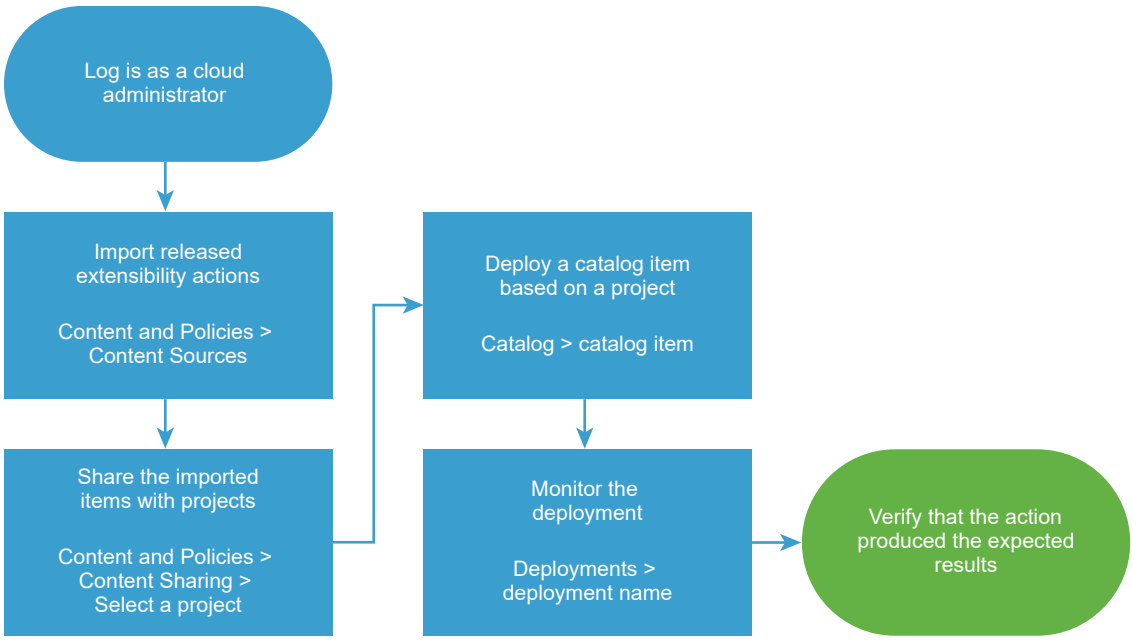
The vRealize Orchestrator workflows are imported into Service Broker and shared in the catalog.

What to do next

- If the deployment fails, click the deployment name and begin troubleshooting. See What can I do if a Service Broker deployment fails. If you are a Cloud Assembly cloud administrator, you can also do more extensive troubleshooting in Cloud Assembly What can I do if a Cloud Assembly deployment fails in *Using and Managing VMware Cloud Assembly*.
- If you want to control how long a deployment can exist, create a lease. See Setting up Service Broker policies.
- To provide more or fewer user inputs at request time, you can create a custom form. See Customize a Service Broker icon and request form. If a workflow includes data grids, do not change the column IDs in the custom form. Use the IDs provided in the workflow.
- To learn more about working with workflows from more than one vRealize Orchestrator instance, consider this blog post from a VMware solution architect.

Add extensibility actions to the Service Broker catalog

As a cloud administrator, you can add Cloud Assembly extensibility actions to Service Broker as a content source. The extensibility actions are created and managed in Cloud Assembly.



The actions are small scripts that perform lightweight tasks or steps. For example, rename a virtual machine or assign an IP address.

Prerequisites

- Verify that the actions you are adding are associated with a project, and that they are released. See How do I create extensibility actions.

Procedure

1 Import the released extensibility actions.
   a Select **Content and Policies > Content Sources**, and click **New**.
   b Click **New**, and then click **Extensibility actions**.
   c Enter the **Name** for this content source.
   d Select the **Source project** and then click **Validate**.

      The validation process verifies the number of released extensibility actions that are associated with the project in Cloud Assembly.
   e Click **Create and Import**.
2 Share the imported actions with a project.
   a Select **Content and Policies > Content Sharing**.
   b Select the project that includes the users who should be able to deploy the extensibility actions.
   c Click **Add Items** and then select one or more actions to share with the project.

      You can select all the items imported from a content source or you can expand the source trees and select individual items.
   d Click **Save**.

   Content Sharing page lists all the items entitled to the selected project. The actions are also added to the catalog where the project members can request them.
3 Verify that the action is available in the catalog to the members of the selected projects.
   a Click **Catalog**, locate the imported extensibility action, and review the projects to ensure that the project you configured is included.
   b Click **Request** and provide any required information.
   c Click **Submit**.

   The provisioning process begins and the Deployments tab opens with your current request at the top.

4 Monitor the provisioning process to ensure that the action runs successfully.

   a Click **Deployments** and locate your deployed request.

   b Monitor the card status until it is successful.

Results

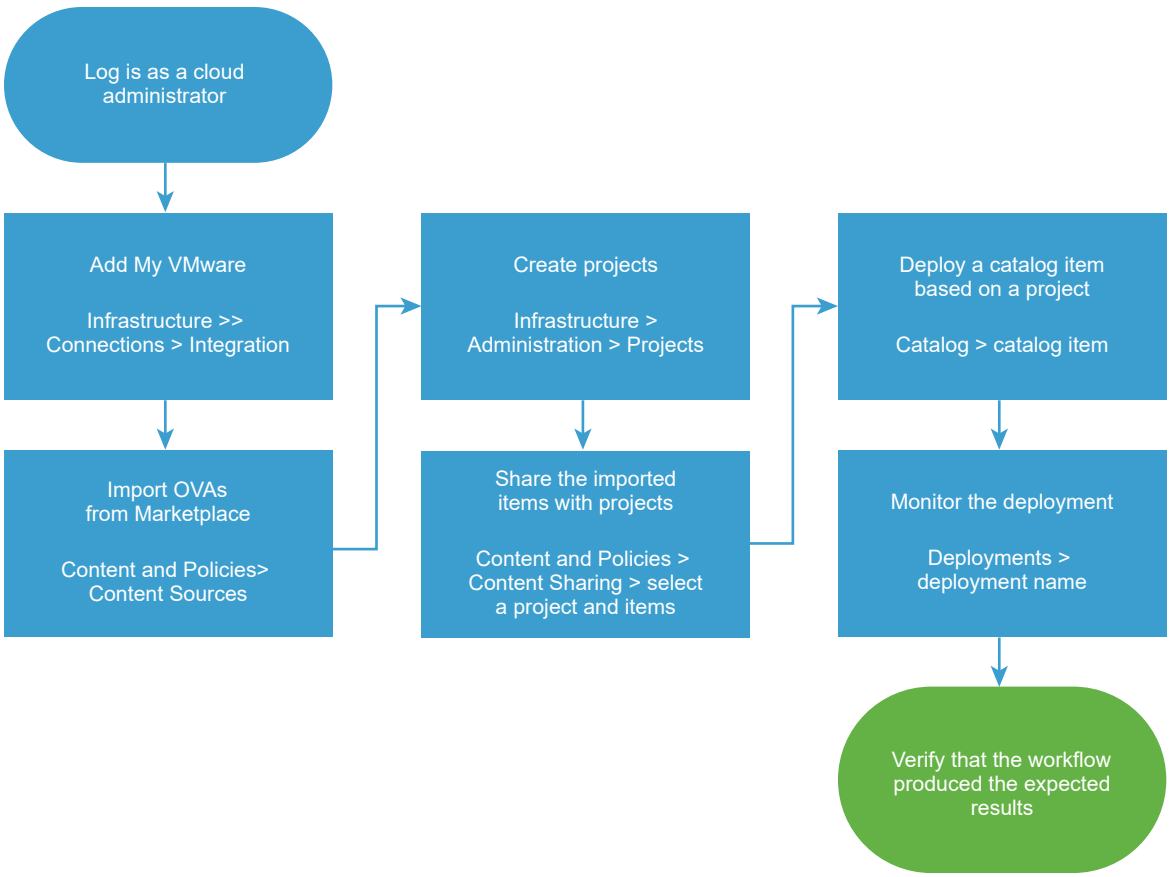
The extensibility actions are imported into Service Broker and shared in the catalog.

What to do next

- If the deployment fails, click the deployment name and begin troubleshooting. See What can I do if a Service Broker deployment fails. If you are a Cloud Assembly cloud administrator, you can also do more extensive troubleshooting in Cloud Assembly What can I do if a Cloud Assembly deployment fails in *Using and Managing VMware Cloud Assembly*.
- If you want to control how long a deployment can exist, create a lease. See Setting up Service Broker policies.
- To provide more or fewer user inputs at request time, you can create a custom form. See Customize a Service Broker icon and request form.

Add VMware Marketplace templates to the Service Broker catalog

As a cloud administrator, you can add Marketplace OVA files to the Service Broker catalog.



Prerequisites

- Verify that you have a My VMware account.

Procedure

1. If you do not have a My VMware integration configured in Cloud Assembly, you can add the integration in Service Broker.

   You can configure only one My VMware integration.

   a. Select **Infrastructure > Connections > Integrations**.

   b. Click **Add Integration** and then click **My VMware**.

   c. Enter a name that identifies this instance when you create the content source.

   d. Enter the My VMware credentials and click **Validate**.

   e. Click **Add**.

2. Import the OVAs.

   You can configure only one **Marketplace VM templates - OVA** content source.

   a. Select **Content and Policies > Content Sources**.

   b. Click **New**, and then click **Marketplace VM templates - OVA**.

   c. Enter the **Name** for this content source.

   d. Select the My VMware account to use to import the templates and click **Validate**.

   e. Click **Create and Import**.

3. If you do not have a project, add a project so that you can share the OVAs with project members.

   a. In Service Broker, select **Infrastructure > Administration > Projects**, and then click **New Project**.

   b. Enter the project information on the **Summary** tab.

   c. Click the **Users** tab and then click **Add Users**.

      To add project users, the individuals or the groups must already be active service organization users.

   d. Click the **Provisioning** tab and select the cloud zones that the OVAs can be deployed to.

      The cloud zones must include the resources that support an OVA when a catalog consumer deploys it.

   e. Click **Create**.

4 Share the imported OVA files with a project.

   a Select **Content and Policies > Content Sharing**.

   b Select the project that includes the users and the infrastructure resources that support the OVA.

      The project gives members permission to deploy the OVAs, and it specifies what infrastructure resources the OVA can be deployed to.

   c Click **Add Items** and then select one or more OVA files to share with the project members.

      You can select all the items imported from a content source or you can expand the source trees and select individual items.

   d Click **Save**.

5 Verify that the OVA file is available in the catalog to members of the selected project.

   a Click **Catalog**, locate the imported OVA, and review the projects to verify that the project you configured is included.

      Alternatively, you can filter the catalog based on the project name.

   b Click **Request** and provide any required information.

   c Click **Submit**.

   The provisioning process begins and the Deployments tab opens with your current request at the top.

6 Monitor the provisioning process to verify that the OVA runs successfully.

   a Click **Deployments** and locate your deployed request.

   b Monitor the card status until it is successful.

Results

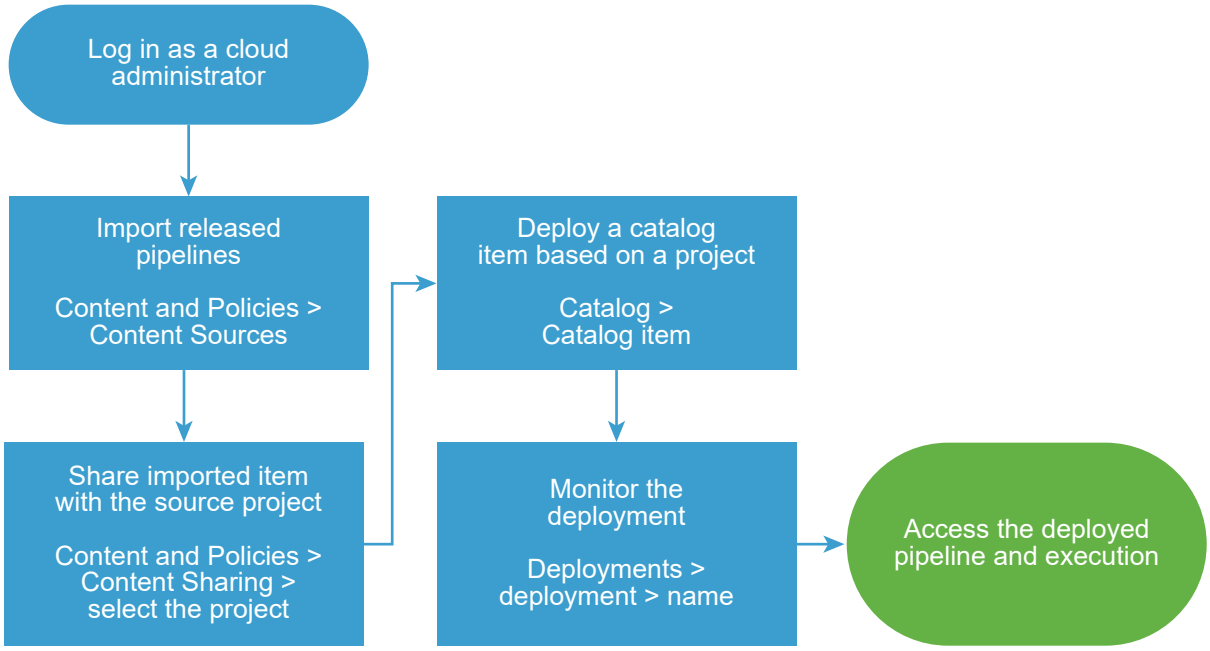
The OVAs are imported and available in the Service Broker catalog for deployment.

What to do next

- If the deployment fails, click the deployment name and begin troubleshooting. See What can I do if a Service Broker deployment fails. If you are a Cloud Assembly cloud administrator, you can also do more extensive troubleshooting in Cloud Assembly What can I do if a Cloud Assembly deployment fails in *Using and Managing VMware Cloud Assembly*.
- If you want to control how long a deployment can exist, create a lease. See Setting up Service Broker policies.
- To provide more or fewer user inputs at request time, you can create a custom form. See Customize a Service Broker icon and request form.

Add Code Stream pipelines to the Service Broker catalog

As a service administrator, you can make Code Stream pipelines available in the Service Broker catalog by adding a Code Stream content source and sharing the pipelines. The pipelines are the continuous integration and delivery model of your software release process.



After you import the pipelines, you share them with project members so that they can deploy the pipelines from the catalog. After the pipeline deployment execution completes, the users can access review the inputs and outputs, and use the output, pipeline, and execution links.

Prerequisites

- Verify that the pipelines that you are importing are enabled and released in Code Stream before you import it. See How do I run a pipeline and see results in *Using and Managing vRealize Automation Code Stream*.

Procedure

1. Import pipelines from Code Stream.
   a. Select **Content and Policies > Content Sources**.
   b. Click **New**, and then click **Code Stream Pipelines**.
   c. Enter the **Name** for this content source.
   d. Select the **Source project** and then click **Validate**.

      The validation process tests the connection and provides the number of released pipelines that are associated with the project in Code Stream.
   e. Click **Create and Import**.

The Content Sources page lists your new source and the number of discovered and imported items.

2 Share the imported items with the source project so that they appear in the catalog.

   a Select **Content and Policies > Content Sharing**.

   b Select the source project that includes the users who have permission to request the pipelines.

   c Click **Add Items** and then select one or more pipelines to share with the project.

   You can select all the items imported from a content source or you can expand the source tree and select individual items.

   d Click **Save**.

   The Content Sharing page lists all the items entitled to the selected project. The pipelines are also added to the catalog where the project members can request them.

3 Verify that the pipeline is available in the catalog to the members of the selected projects.

   a Click **Catalog**, locate the imported pipeline.

   b Click **Request** and provide any required information.

   c Click **Submit**.

   The provisioning process begins and the Deployments tab opens with your current request at the top.

4 Monitor the provisioning process to ensure successful deployment.

   a Click **Deployments** and locate your deployed catalog item.

   b Monitor the card status until it is successful.

   You can open the deployment, review the inputs and outputs, use the links to access the output URL, and use the links to the pipeline and execution in Code Stream.

Results

The released pipelines are imported into Service Broker, shared in the catalog, and deployable.

What to do next

- If the deployment fails, click the deployment name and begin troubleshooting. See What can I do if a Service Broker deployment fails. If you are a Cloud Assembly cloud administrator, you can also do more extensive troubleshooting in Cloud Assembly What can I do if a Cloud Assembly deployment fails in *Using and Managing VMware Cloud Assembly*.
- If the deployment fails, review the failed execution in Code Stream.
- If you want to control who must approve a pipeline request before it provisions, create an approval policy. See How do I configure Service Broker approval policies. The lease and day 2 policies do not apply to pipelines.

- To provide more or fewer user inputs at request time, you can create a custom form. See Customize a Service Broker icon and request form.

Setting up Service Broker policies

To provide the background management of your deployments, you set up policies. Each Service Broker policy is a set of rules or parameters that are applied to deployments, freeing the cloud administrator for other tasks.

Any policies that you create in Service Broker are applied to the deployments in Service Broker and in Cloud Assembly.

Getting started with policies

To begin creating policies, select **Content and Policies > Policies > Definitions**. Any policy that you add is applied to current deployments and any new deployments.

To get you started, use the full use cases that are provided for each policy type. The use cases guide you through the process of creating more than one policy. The use case provides contextual explanations of the choices and the desired behavior.

The use cases are followed by more in-depth information about how multiple policies are processed.

How do I configure Service Broker approval policies

Approval policies are a level of governance that you add to exercise control over deployment and day 2 action requests before they are run. You define approval policies in Service Broker so that you, or others that you designate, review requests before resources are consumed or destroyed. The approval policy use cases in this procedure are an introduction that you can use as you explore your governance options.

If you have only a small team adding and deploying catalog items, then approval policies might be less useful. But as you make the catalog available to a larger group of developers and general consumers, you can use the approval policies to ensure that someone reviews a request before the resources are consumed or changes are made to the provisioned items.

For example, you have a catalog item that is important, but it consumes a significant amount of resources. You want one of your IT administrators to review any deployment requests to ensure that the request is needed. Another example applies to day 2 actions. Making changes to a deployment that is used by many might be devastating. You want the project administrator who manages the deployment for that team by reviewing all changes to the deployed catalog item.

Who works with or is affected by approval policies?

- Service Broker administrator. Configures the policies.
- Catalog consumers. Users who request catalog items or day 2 actions to which one or more policies apply.

- Users deploying cloud templates in Cloud Assembly. Users who request templates or day 2 actions in Cloud Assembly to which one or more policies apply.
- Designated approvers. Users who must review and then approve or reject a request. You can grant approver rights to selected users, or you can choose from the following approver roles.
  - AD Manager. Active Directory user with manager attributes. See Configure Active Directory attributes for the AD Manager approver role
  - Project Administrators. Administrators of projects within the policy scope are automatically assigned as approvers. If a project does not have a dedicated administrator, the approval policy is not applied to that project.
  - Project Supervisors. Members of projects within the policy scope who are assigned the Supervisor role. Supervisor access rights are limited to approving and rejecting deployment requests for a project. If a project does not have a dedicated supervisor, the approval policy is not applied to that project.

What happens when approval policies are enforced?

Multiple approval policies might be enforceable. The approval policies are evaluated, and an enforced policy is applied to the request. When there are multiple valid policies, where the approvers are different people, all the approvers are added. When you have multiple policies, it is important to understand this process. For more information, see Approval policy goals and enforcement examples.

1 Approval policies are defined.

2 A user requests a catalog item or day 2 action. At request time, Service Broker evaluates the catalog item to see if any policies apply.

3 An approval policy is enforced.

   a The deploy card displays the status. For example, Create - Approval Pending.

   b An email notification is sent to the requester. See How do I track my requests that require approval in Service Broker.

   c An email notification is sent to the approvers. See How do I respond to an approval request in Service Broker.

      The deployment does not begin deploying and consuming infrastructure resources, or make changes to a deployed system, until the request is approved. The requesting user is notified by email that the request is waiting for approval.

   d The approvers respond to the request using the Approvals tab in Service Broker.

4 The approval process is completed.

   a If the request is rejected, the requesting user is notified and the deployment request is canceled.

   b If the request is approved, the deployment proceeds.

c It is possible that the enforced policy is configured to automatically approve or reject a request if the approver does not take any action.

How can I use the deployment criteria?

To limit what items or activities the policy applies to, you can define the deployment criteria. For more about the criteria, see How do I configure deployment criteria in Service Broker policies.

Approval policy constraints

- The change lease action is not available to include in an approval policy.
- Using custom resources as resource type in the policy criteria is not supported.

As you review the approval policies use case and create your own policy, consult the signpost help on the key text boxes for more information.

Prerequisites

- An approver, who might not be a regular Service Broker or vRealize Automation Cloud Assembly user, must have one of the following combination of roles:
  - Organization member and Service Broker user
  - Organization member and the Manage Approvals custom role

  These roles provide the minimum level of permissions and still allow them to approve or reject a request.
- Verify that the email notification server is defined. See Add an email server in Service Broker to send notifications.
- If you plan to use the Active Directory manager as the role-based approval type, you must use the Workspace One Access VMware Identity Manager integration configured for vRealize Automation. You must also include the Active Directory manager attributes in the user attributes. See Configure Active Directory attributes for the AD Manager approver role

Procedure

1 Select **Content and Policies > Policies > Definitions > New Policy > Approval Policy**.

2 Configure Approval Policy 1.

As an administrator, you have an important catalog item that also consumes a significant amount of your cloud resources. You want at least one of your two IT administrators to review any deployment requests to ensure that the request is really needed and that the resources exist to support it.

a Define when the policy is valid.

Setting	Sample Value
Scope	Organization This policy is applied to all projects in your organization.
Criteria	`Catalog Item equals CompanyApplication`

b Define the approval behavior.

Setting	Sample Value
Approval type	Select **User based**. You select the users who are the request approvers.
Approver mode	All You want all your IT managers to agree that the deployment request does not waste resources.
Approvers	{approvername1}@YourCompany, {approvername2}@YourCompany
Auto expiry decision	Reject The possible load on your cloud resources means that you do not want to inadvertently deploy the item without approval.
Auto expiry trigger	3 This value should carry you over a long weekend when the managers might not be available.
Actions	Deployment.Create

In this scenario, if any catalog consumer requests this catalog item, Approver 1 and Approver 2 must both approve the request within 3 days or the request is rejected.

3 Configure Approval Policy 2.

As an administrator, you have several projects where you want the project administrators to approve any changes to deployments that might have catastrophic consequences. For example, deleting the deployment.

a Define when the approval policy is valid.

Setting	Sample Value
Scope	Multiple Projects `Project name contains Prod` The policy is applied to deployments associated with all projects that match the scope criteria.
Criteria	None

b Define the approval behavior.

Setting	Sample Value
Approval type	Select **Role based**.
Approver role	Project Administrators If a project does not have a dedicated administrator, the approval policy is not applied to requests associated with that project.
Auto expiry decision	Reject
Auto expiry trigger	7
Actions	Deployment.Delete, Deployment.PowerOff, Deployment.Update, and any of the component-specific power, reboot, and delete actions.

In this scenario, when a member of one of the scoped projects submits a request to run the listed actions on a deployment, the request is rejected after seven days if the project administrator does not respond.

4 Configure Approval Policy 3.

As an administrator, you want to maintain a little control over resource consumption. For example, when a user requests a catalog item where the size is large, you want to evaluate and approve the request. Size is defined by the flavor mappings.

a Define when the approval policy is valid.

Setting	Sample Value
Scope	Organization
Criteria	`Resources has any` `  Flavor equals large`

b Define the approval behavior.

Setting	Sample Value
Approval type	Select **User based**.
Approver mode	Any
Approvers	{AdminName}@YourCompany
Auto expiry decision	Reject The possible consumption of your cloud resources means that you do not want to inadvertently deploy the item without approval.
Auto expiry trigger	5
Actions	Deployment.Create and any applicable *.Machine.Resize actions. For example, Cloud.vSphere.Machine.Resize.

In this scenario, when any user submits a request for a large deployment or to resize a deployment to large, the request is rejected after 5 days if the cloud administrator does not respond.

What to do next

- For more information about how approval policies are processed, see Approval policy goals and enforcement examples.
- For more about the consumer and approver experience, see How do I track my requests that require approval in Service Broker and How do I respond to an approval request in Service Broker.

Configure Active Directory attributes for the AD Manager approver role

You must have the manager Active Directory attributes configured in Workspace ONE Access VMware Identity Manager if you plan us use role-based approvers for approval policies in Service Broker. To do this you must have permission to configure the VMware Identify Manager instance that you use with vRealize Automation.

This procedure primarily covers work that you perform outside of vRealize Automation. Links to relevant procedure are provided.

Prerequisites

- Verify that you have administrator credentials in Workspace ONE Access and VMware Identity Manager.

Procedure

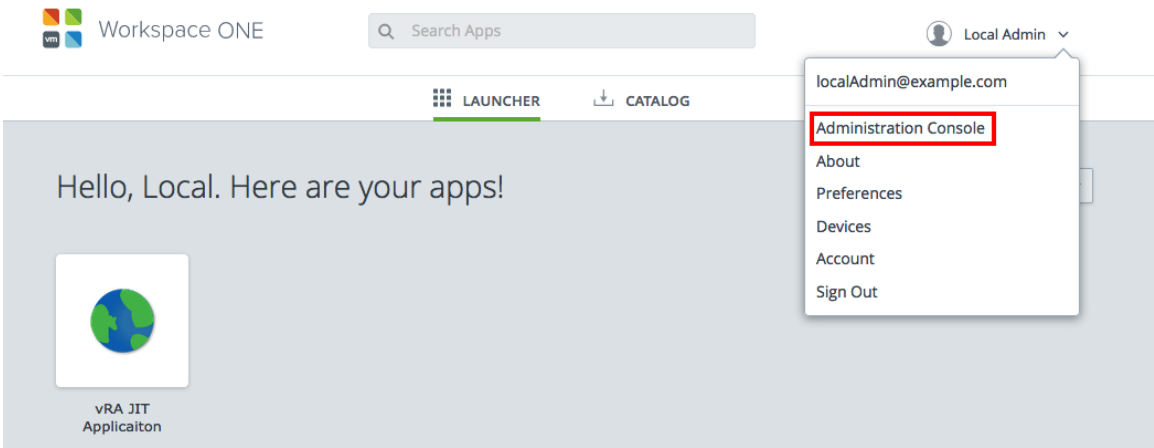
1. In the VMware Identity Manager instance that you use with vRealize Automation, verify that your are integrating Active Directory with Identity Manager.

   See Integrating with Active Directory.

2. Configure the user attributes.

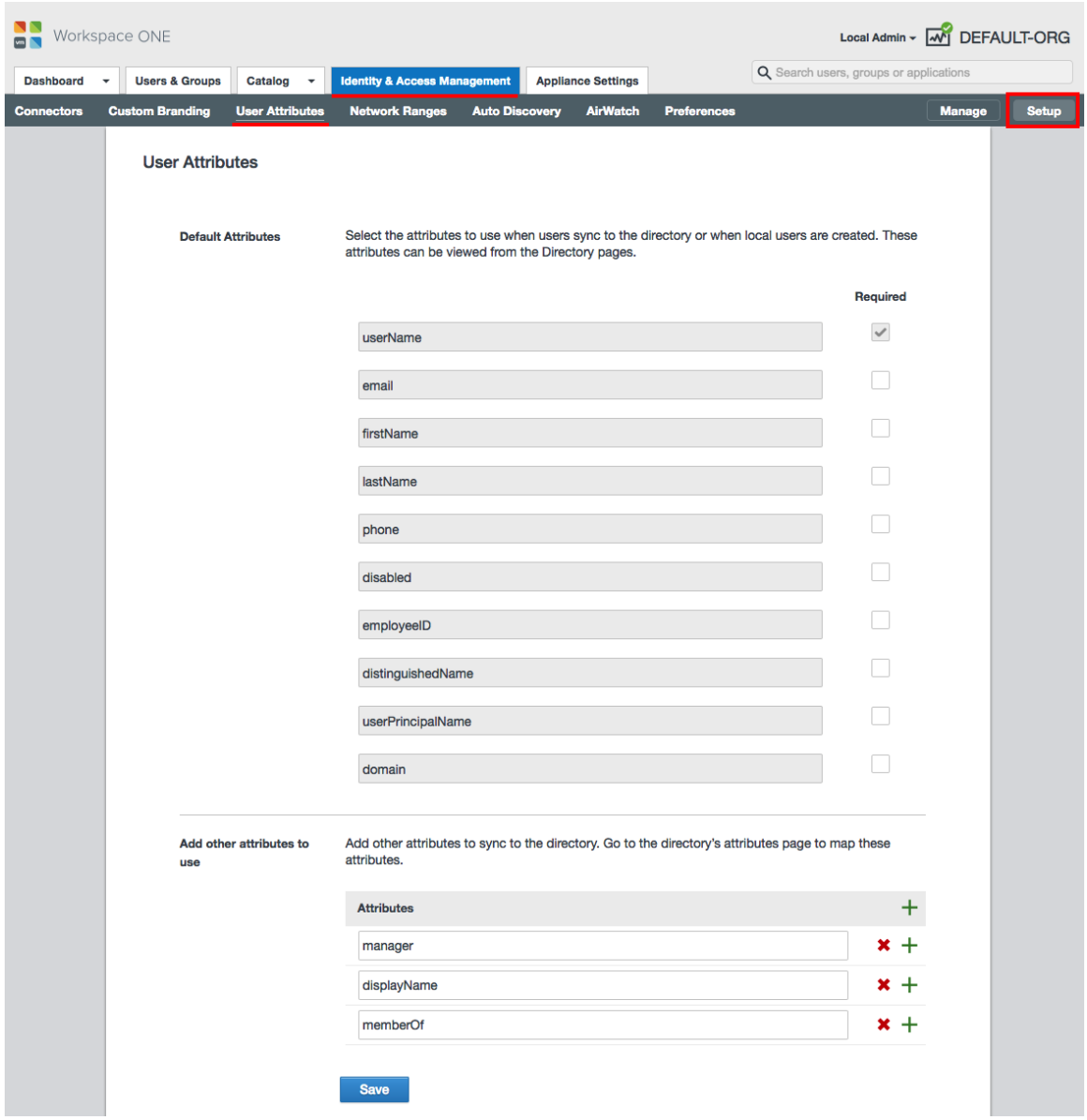
   The basic steps are provided below. For more information, see Managing User Attributes that Sync from Active Directory.

   a. In Identity Manager, click your local administrator login and click **Administration Console**.



   b. Select the **Identity and Access Management** tab and click **Setup**.

c Click **User Attributes**.



d Verify that the following attributes exist in the **Default Attributes** section.

- userName
- email
- firstName
- LastName
- phone
- disabled
- employeeID
- distinguishedName
- userPrincipalName
- domain

e In the **Add other attributes to use** section add the following attribute.

- manager

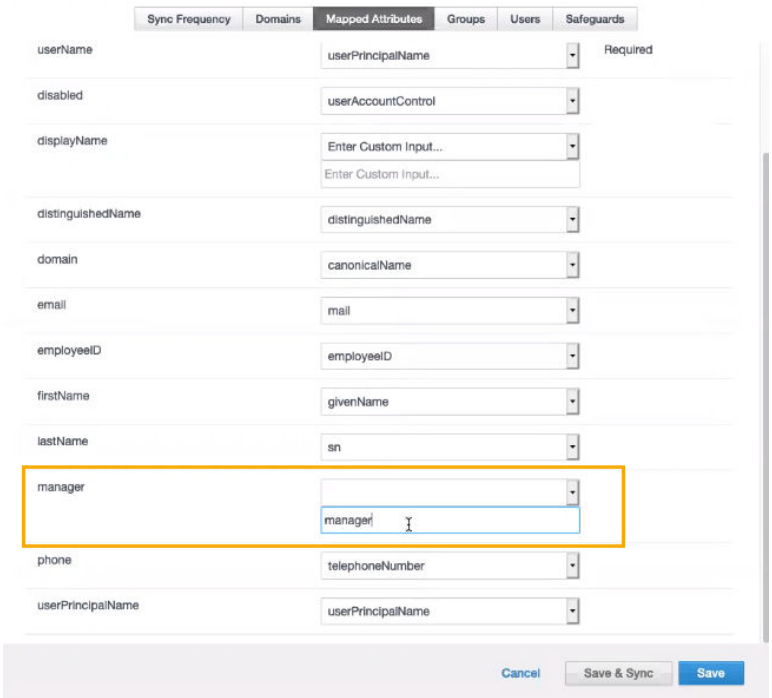
f Click **Save**.

3 After you make any changes, you must synchronize the affected directories.

a Click **Manage**.

b Select the **Directories** tab.

c Open the directory by clicking the directory name and click **Sync Settings**.



d Click **Mapped Attributes** and verify that the manager attribute is defined as **`manager`**.

e Click **Save and Sync**.

f Click **Sync Directory**.

Results

You can now use the AD Manager role in you approval policies.

How do I entitle deployment users to Service Broker day 2 actions using policies

You define day 2 action policies so that you can control what changes your users can make to deployments and their component resources. By creating a list of permitted actions that all or some users can run on deployments, you ensure that the users cannot initiate any destructive or costly changes. The use cases related to day 2 actions policies are an introduction to the procedure.

When you entitle users to run day 2 actions, you select the individual actions that they can run. You are creating an inclusion list, not an exclusion list.

When does a day 2 actions policy go into effect?

- If you do not have any Day 2 Action policies defined, then no governance is applied and all users have access to all the actions. This initial lack of governance as you are starting out ensures that you and your users can exercise the day two actions in Service Broker and Cloud Assembly without the need to understand day 2 policies.
- After you determine that you are ready to control who has access to what actions, you add governance in the form of a single Day 2 Action policy. When the first policy goes into effect, the Day 2 Action policies are enforced for all users in Service Broker and Cloud Assembly. As a result, only the users for whom the first policy is true can run the selected actions. All others are excluded. They are excluded because the actions policies include the trusted users. By excluding all others, you are able to craft the policies to match your governance goals.
- To entitle other users, you must create policies that entitle them to run the actions you select.

Deployment sharing in projects affects how you configure the day 2 actions entitlements. If the project is not set to share, then only the requesting user can see a deployment. If the project shares deployments, then all the members of the project can see the deployment, and run any actions that they are entitled to run by a Day 2 Action policy. Deployment sharing is configured in a project. Select **Infrastructure > Administration > Projects**, then select the project and click the **Users** tab.

As you create your policies, the way that you define Day 2 Actions policies must take sharing status into consideration.

To focus when the Day 2 Actions policies are applied, you can configure scope, role, and criteria. These configurations control what deployments the policy is applied to and who can run the actions when the policy is enforced.

- What deployments the policy is applied to.
  - Scope determines whether the policy is applied to deployments at the organization or project level.
  - Criteria narrows the scope of the policy to particular aspects of deployments.
- Who can run what actions on those deployments.
  - Role entitles the members of the selected role, within the selected scope and criteria, to run the selected actions. The role can be project administrator, project member, or a named custom role.

Day 2 policies are enforced when a user tries to manage a deployment using the Actions menu on the deployment or on the component resources.

In this use case, which is used to illustrate a collection of day 2 action policies, the assumption is that you enabled deployment sharing in the project.

As you review the day 2 actions policies use case, you must also select the actions. You must select the actions that support your cloud accounts.

- Actions are cloud specific. When you are entitling the users to make changes, consider what cloud accounts the entitled users are deploying to and ensure that you select all the cloud-specific versions of the actions. For example, add Cloud.AWS.EC2.Instance.Resize, Cloud.GCP.Machine.Resize, and Cloud.Azure.Machine.Resize to entitle users to resize those machines.
- Cloud agnostic actions, for example, Cloud.Machine.Resize, exist to accommodate resources where the on-boarding or migration process cannot identify the machine type. If you entitle users to the cloud agnostic actions, you have not entitled them to run the cloud-specific action that will make the changes to the deployed resources. The agnostic actions might appear in the action menu, but running the actions has no effect. You should avoid entitling the agnostic actions and only entitle cloud-specific actions to ensure that actions are available to the users for your various cloud platforms.

Prerequisites

- For a list of possible actions, see What actions can I run on Service Broker deployments.
- For more information about constructing deployment criteria, see How do I configure deployment criteria in Service Broker policies.
- Custom roles are used in Day 2 Policy 4. Create a Deployment Troubleshooter role, but with the Manage Deployment role in the custom Deployment Troubleshooting role does

not limit the members by project. The Manage Deployment role allows the assignees to see all deployments and run all actions. If the Troubleshooting Deployments role does not include Manage Deployments, then the assignees see deployments based on their project membership. For more information about custom roles, see custom role use case.

Procedure

1 Select **Content and Policies > Policies > Definitions > New Policy > Day 2 Actions Policy**.

2 Configure Day 2 Policy 1.

As an administrator, you want to control storage costs by restricting the ability of users to request snapshots.

a Define when the policy is valid.

Setting	Sample Value
Scope	Organization This policy applied to all deployments in your organization.
Criteria	None
Enforcement type	Soft This enforcement type allows you to create other policies related to the snapshot actions that override this policy.
Role	Member This role applies the policy to all project members.

b Select the actions that the users can run, but do not select any snapshot actions.

You explicitly entitle users to run actions. To exclude users from running snapshot actions, ensure that the actions are not selected.

In this scenario, none of the project members in your organization are entitled to create snapshots. Nor can your project administrators. Your next step is to create a policy that entitles the project administrators to create and manage snapshots.

3 Configure Day 2 Policy 2.

As an administrator, you want to give the project administrators the ability to create and manage snapshots.

a Define when the policy is valid.

Setting	Sample Value
Scope	Organization This policy is applied to all deployments in your organization.
Criteria	None
Enforcement type	Soft This enforcement type allows you to create other policies related to the snapshot actions that override this policy.
Role	Administrator This role applies the policy to the project administrators.

b Select the snapshot actions that you want the administrators to run.

Project administrators are also entitled to run any actions that the members of their projects are entitled to run. You do not need to give them permission to member actions.

In this scenario, the project administrators are entitled to run the snapshot-related actions and all the actions that their project members are entitled to run.

4 Configure Day 2 Policy 3.

As a project administrator, you have two developers who are doing work that potentially makes a deployment unusable. You want to entitle them to snapshot and revert without your intervention. You entitle the two project members to use the snapshot actions.

a Define when the policy is valid.

Setting	Sample Value
Scope	Project MT5 This policy applied to deployments associated with this project.

Criteria	`Catalog Item equals Multi-tier five machine with LB`
	`AND`
	`(Created By equals jan@mycompany.com`
	`OR`
	`Created By equals kris@mycompany.com)`
	Based on this criteria expression, only the deployments where Jan or Kris deployed a catalog item named Multi-tier five machine with LB are considered for policy enforcement.

Enforcement type	Hard This enforcement type ensures that the policy is enforced based on the definition.
Role	Member This role applies the policy to the catalog item defined in the deployment criteria.

b Select the snapshot actions that you want the specified users to run.

Project administrators are also entitled to run any actions that the members of their projects are entitled to run.

In this scenario, Jan and Kris can use the snapshot actions on the Multi-tier 5 Machines with LB catalog item that either of them deploy. Although other members of the project can see the deployment, only Jan, Kris, and the project administrator can use the snapshot actions.

5 Configure Day 2 Policy 4.

As an administrator, you want to assign the permissions to run most of the day 2 actions to the users who are assigned to a Deployment Troubleshooter custom role. While most custom role permissions go across projects, what users can see in the Deployments tab is based on their project membership. To see the deployments, the users who are assigned the custom roles must be members of the projects that deployed them.

a Define when the policy is valid.

Setting	Sample Value
Scope	Organization
Criteria	None
Enforcement type	Soft This enforcement type allows you to create other policies related to the extended day 2 that override this policy.
Role	Select the **Deployment Troubleshooter** role.

b Select all the actions that you want the members of this custom role to be able to run.

In this scenario, all the users with the Deployment Troubleshooting role can manage all deployments and run all selected day 2 actions across projects. The Manage Deployments role grants service administrator privileges on deployments so that they can run any action that a service administrator can run. If the Deployment Troubleshooting custom role does not include the Manage Deployments role, the users can run all the selected day 2 actions for the deployments belonging to their projects.

What to do next

- For more examples of how the policies are processed and enforced, see How are Service Broker policies processed.
- Configure policies that are relevant to your organizations and projects.

How do I configure Service Broker deployment leases using policies

By using policy-based leases, you reduce the need to intervene manually to reclaim resources. You define lease policies so that you can control the amount of time that a deployment is available to your users. The lease policy use cases in this procedure provide a beginning point for learning about and implementing policies for your organization.

If you do not have any lease policies defined, then the deployments never expire. To reclaim the resources, you must manually destroy the deployments.

When does a lease policy go into effect?

- If the policy scope is Organization, then all the deployments in your organization are managed based on the defined policies.

- If the policy scope is a project, then the deployments that are associated with that project are managed based on the defined lease. Other projects are not affected.

Lease policies are applied when you:

- Create or update a lease policy. After lease policies are applied, they continuously evaluate the deployments in the background to ensure that they are in compliance with the defined leases.
- Request a catalog item in Service Broker or a cloud template in Cloud Assembly. The maximum lease and maximum total lease values go into effect when the deployment is created.
- Onboard workloads or resources in Cloud Assembly so that you can manage them using Service Broker, Cloud Assembly, or Code Stream.

In this use case, there are three policy definitions that illustrate how you can construct policies and the results when they are enforced. The last policy is not enforced, but the reasons are provided in the scenario results.

As you review the lease policies use case, you must also configure lease-specific options. The following descriptions provide a brief summary. Consult the signpost help for more information.

- Maximum Lease (days). The number of days that the deployment resources are active without being renewed. If they are not renewed, the lease expires and the deployment is destroyed. If a grace period is specified, the user can renew the lease for up to the same number of days that the lease has been active.
- Maximum Total Lease (days). The combined total number of days that the deployment can be active, including lease renewals. Each renewal cannot exceed the maximum lease, and the cumulative renewal value cannot exceed the maximum total lease. After the total lease is reached, the deployment is destroyed and the resources within that deployment are reclaimed.
- Grace period (days). The number of days the user has to renew an expired lease before the deployment is destroyed. The grace period is not included in the total lease days. If you don't define a grace period, it defaults to 1 day.

Procedure

1 Select **Content and Policies > Policies > Definitions > New Policy > Lease Policy**.

2 Configure Lease Policy 1.

As an administrator, you want to control costs by limiting the starting lease time for all deployments to 30 days, with the option to renew the lease for a total of 90 days.

a Define when the policy is valid.

Setting	Sample Value
Scope	Organization This policy is applied to everyone in your organization.
Criteria	None
Enforcement type	Soft This enforcement type allows you to create other policies related to this lease that override this policy.

b Define the lease.

Setting	Sample Value
Maximum lease (days)	30
Maximum total lease (days)	90
Grace period (days)	10

In this scenario, the deployment is shut down after 30 days and an email is sent to the user. During the grace period, the user extends the lease by 30 days. After the lease expires again, the user renews it for another 30 days. At the end of the third extension, the lease reaches the maximum total lease period of 90 active days and the user cannot extend it anymore. The deployment is shut down and destroyed 10 days later.

3 Configure Lease Policy 2.

As an administrator, you want to control costs by limiting the lease time on an expensive template to two weeks. For this example, the template name is `Multi-tier 5 machine with LB`.

a Define when the policy is valid.

Setting	Sample Value
Scope	Project MT5 This policy applied to deployments associated with this project.
Criteria	`Cloud Template equals Multi-tier 5 machine with LB` Based on this criteria expression, only deployments for the referenced template are considered for policy enforcement.
Enforcement type	Soft This soft enforcement still overrides the organization policy of 90 days in Policy 1 because the values are more meaningful at the project level.

b Define the lease policy.

Setting	Sample Value
Maximum lease (days)	14
Maximum total lease (days)	28
Grace period (days)	3

In this scenario, both policies are applied, but Policy 2 takes precedence over Policy 1 because it is more specific. When applied, the deployment is shut down after 14 days. If the user does not extend the lease, it is destroyed three days later. If the user extends the lease for up to another 14 days, the deployment is shut down at the end of the second extension and it is destroyed three days later.

4 Review the configuration of Lease Policy 3.

As a project manager, you realize that one of your developers is working on a complex application. The developer requires the `Multi-tier 5 Machines with LB` template and another template, `Distributed Database Across Clouds`, but for a longer lease than defined in Policy 2.

Unless you understand how the policies are processed based on how they are defined, you might encounter unexpected results. Policy 3 is an example of how processing and precedence affect the result.

This policy, as provided, will not be enforced. This example provides an opportunity for you to see how leases are applied and enforced when there is more than one that applies.

a Define when the policy is valid.

Setting	Sample Value
Scope	Project MT5 This policy is applied to deployments in this project.

Criteria	`(Cloud Template equals Multi-tier five machine with LB`
	`OR`
	`Catalog Item equals Distributed Database Across Clouds)`
	`AND`
	`Created By equals jan@mycompany.com`
	You use Catalog Item because it is a non- Cloud Assembly template.

Enforcement type	Soft This soft enforcement still overrides the organization policy of 90 days in policy 1 because the values are more meaningful at the project level.

b Define the lease policy.

Setting	Sample Value
Maximum lease (days)	21
Maximum total lease (days)	50
Grace period (days)	3

In this scenario, Lease Policy 2 is applied, not Lease Policy 3.

- Lease 3 has a lease time that is less than or equal to 21 days, and the policy is applied. Lease 2 has a lease time that is less than or equal to 14 days, and the policy is applied.
- Lease 2 is applicable and it does not violate the lease 3 policy. But, lease 2 is more restrictive, so it takes precedence. Lease policy 2 is more restrictive because it is for a shorter period of time.
- When both lease definitions are true and applicable, the more restrictive policy is the one that is enforced.

5 To resolve the unexpected behavior in Lease Policy 3, you can implement one of the following solutions.

- To ensure that you can provide Jan with the needed policy, change the enforcement type to hard.

- Alternatively, you could create a new project with access to the same resources, and then create Lease Policy 3 for that project. While this solution isolates the working policy, you must maintain a parallel project. The effort needed to set up and maintain the content sources, content sharing, and so on, are time consuming and subject to error.

What to do next

- For more examples of how the lease policies are processed and enforced, see How are Service Broker policies processed.
- Configure policies that are relevant to your organizations and projects. If you are just getting started with lease policies, begin with one lease policy at the organization level.
- To send an email to the deploying user, configure the email server for notifications. See Add an email server in Service Broker to send notifications.

How do I configure Service Broker resource quotas using policies

Resource quota policies control the amount of resources that are available to your users. You define resource quota policies so that you limit the resources that can be consumed by each user, project, or the organization. The use cases in this procedure are an introduction to resource quota policies.

If you do not have any resource quota policies defined, then no governance is applied and users can consume resources until all available resources are used up.

As a cloud administrator, you can create one or more resource quota policies and apply them, for example, at the organization level. As users across the organization request deploy resources, resource quota policies track the consumption of resources to ensure that new deployment requests do not exceed the resource limits defined in the policies.

As you create your policies, you must configure policy scope. The scope determines whether the policy is applied to resources at the organization or project level. For more information about policy scope, see How do I configure scope in Service Broker policies.

- If the policy scope is organization, then all resources in your organization are managed based on the defined policies.
- If the policy scope is multiple projects, then the resources that are associated with the specified projects are managed based on the defined policy.
- If the policy scope is a single project, then the resources that are associated with that project are managed based on the defined policy. Other projects are not affected.

When defining resource quotas, you must specify scope level limits for each resource. Level limits provide additional resource governance. For example, if you want to apply a resource quota policy to the whole organization, you can set the scope level to organization limits, or you can define limits for a smaller segment, such as projects or users within that organization.

You can set only one limit for a resource type per scope level in the same policy. For example, you can set a resource quota for storage consumption at the organization level and per user in the same policy. You cannot define two storage quotas at the organization level in the same policy.

Resource quota limits are dependent on the broad policy scope. If you change the scope after you define the resource quota limits, the resource quota settings are deleted and you must start over.

The scope level drop-down menu includes the following options.

Option	Description	Available at these policy scope levels
Organization Limits	Limits the amount of resources that are available for consumption at the organization level. Resource quotas with organization limits are distributed among all users or all projects in the organization.	■ Organization
Organization User Limits	Limits the total amount of resources that each user can consume within the organization.	■ Organization
Projects Limits	Limits the amount of resources that are available for consumption at the project level. Resource quotas with project limits are distributed among all users in the specified projects. Project limits are not cumulative. If the policy scope is set to multiple projects, the resource limits are applied per project.	■ Organization ■ Multiple projects ■ Project
Projects User Limits	Limits the total amount of resources that each user who belongs to the specified projects can consume at the project level.	■ Organization ■ Multiple projects ■ Project

How are resource quota policies enforced?

- Multiple resource quota policies might be enforceable. The resource quota policies are evaluated, and an enforced policy is applied to the deployment request. When there are multiple policies defined for a resource at the same scope level, the resource quota with the lowest limit value is enforced. The use case in this procedure provides more information about how resource quotas are processed.
- When a resource quota policy is enforced, all existing deployment resources are evaluated against the resource quota, except for deployment requests that are in-progress. Resource usage is updated after the deployment request is completed, so in-progress requests are not included in the evaluation.

- Concurrent deployment requests are not supported in resource quota policy enforcement. For example, a resource quota policy allows 15 GB of memory per user. A user triggers two concurrent deployment requests, each consuming 10 GB of memory. The policy allows both requests because at the time of requesting the deployments the user does not consume any memory and each request meets user level limit of 15 GB. After the requests are completed, resource usage is updated to reflect the two requests. If the user then creates a third deployment request, that request fails because no available resources are left.
- When deploying cloud templates, resource quota policies allow over-provisioning of storage because the system does not know the actual storage size of the deployment before the machine is provisioned in the endpoint. Similarly to concurrent requests, after the resource usage is updated and the system recognizes that the provisioning resources exceed the resource quota limit, the policy does not allow any subsequent requests.
- Resource quota policies are not enforced on day 2 actions. For example, if the resource quota limit is 2 CPUs at deployment, the user can deploy with 2 CPUs, and then they can run a day 2 action to increase the amount to 6 CPUs. After the day 2 action completes, CPU usage is updated to account for the newly added resources, which impacts the total amount of resources that are available for consumption.
- Resource quota policies support only VMware vSphere, Amazon Web Services, Microsoft Azure and Google Cloud Platform resources created from cloud templates.

Resource quota policies are applied when:

- A user requests a catalog item in Service Broker or a cloud template in Cloud Assembly.
- When you create a new policy or update an existing policy, the system can take up to two minutes to apply the changes. For example, if you create a new deployment within two minutes of updating a policy, the policy updates might not apply to the deployment request.

In this use case, there are three policy definitions that illustrate how you can construct resource quota policies and the results when they are enforced.

Procedure

1 Select **Content and Policies > Policies > Definitions > New Policy > Resource Quota Policy**.

2 Configure Resource Quota Policy 1.

As a cloud administrator, you want to control how resources are distributed among users and projects in the organization that you administer.

a Define when the policy is valid.

Setting	Sample Value
Scope	Organization This policy is applied to the whole organization.

b Define the resource quotas.

Scope Level	Resource and Limit
Organization Limits	CPU = 2000
Organization User Limits	CPU = 10
Project Limits	CPU = 200
Project User Limits	CPU = 5

In this scenario, the total amount that is available for consumption among all users in the organization is 2000 CPU and the total amount that is available per project is 200 CPU. Each user can use up to 5 CPU in each project that they belong to, but no more than 10 CPU, combined across all their deployments. Once a scope level limits is reached, any new deployment request that exceeds this limit fails.

3 Configure Resource Quota Policy 2.

As a project administrator, you want to control how resources are distributed among developers in several projects that you administer.

a Define when the policy is valid.

Setting	Sample Value
Scope	Multiple Projects Define the project criteria. For example, `Project name contains dev` This policy is applied only to projects whose name contains the phrase *dev*.

b Define the resource quotas.

Scope Level	Resource and Limit
Project Limits	CPU = 100
Project User Limits	CPU = 10

In this scenario, the resources that are available at each scope level are evaluated and both Policy 1 and Policy 2 are enforced. Between the two policies, the lowest limits are applied.

- Projects user limits in Policy 1 are applied because the defined value is lower than in Policy 2.
- Project limits in Policy 2 are applied because the defined value is lower than in Policy 1.
- Organization level limits defined in Policy 1 also apply to the projects specified in the scope of Policy 2.

4 Configure Resource Quota Policy 3.

As a cloud administrator, you want to distribute resources at the project and organization level evenly among users.

a Define when the policy is valid.

Setting	Sample Value
Scope	Organization This policy is applied to the whole organization.

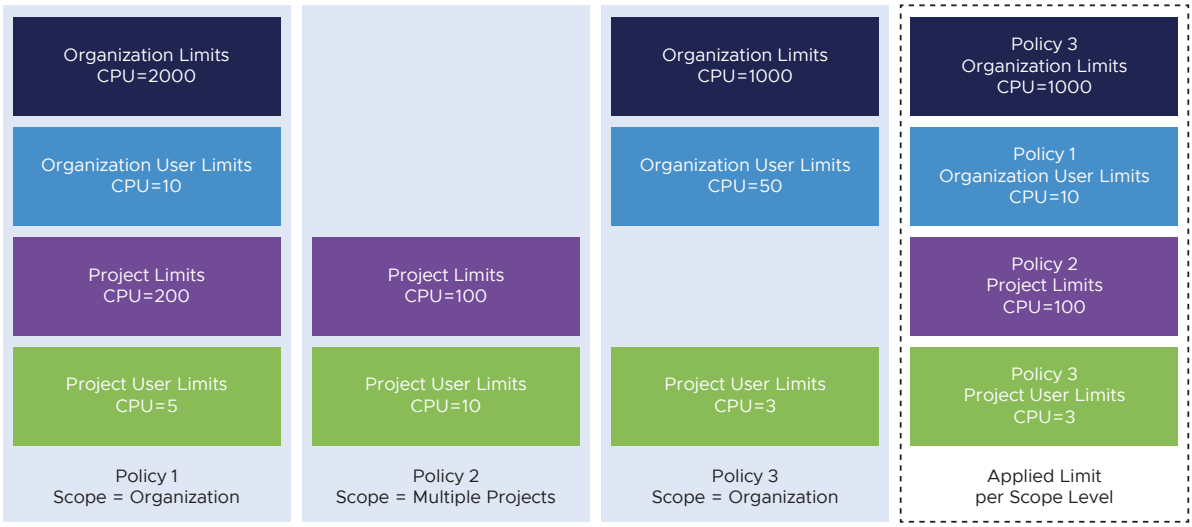
b Define the resource quotas.

Scope Level	Resource and Limit
Organization Limits	CPU = 1000
Organization User Limits	CPU = 50
Project User Limits	CPU = 3

In this scenario, the resources that are available at each scope level are evaluated and all three policies are enforced. Again, the lowest scope level limits between the three policies are applied.

- Projects User Limits in Policy 3 are applied because the defined value is lower than in Policy 1 and Policy 2.
- Organization User Limits in Policy 3 are not applied. Instead, the limit defined in Policy 1 is applied because the value is lower.
- Organization level limits defined in Policy 3 are applied because the value is lower than in Policy 1.

Based on the configuration examples above, the following diagram summarizes how resource quotas across multiple policies are applied.



What to do next

- For more examples of how other policies are processed and enforced, see How are Service Broker policies processed.
- Configure policies that are relevant to your organizations and projects.
- Monitor provisioned resources on the My Resource Usage dashboard. See Learn more about the Service Broker catalog items.

How do I configure scope in Service Broker policies

When you create a policy, you specify its scope to determine how the policy is applied. You can assign the policy to the whole organization, to multiple projects within the organization, or to a single project.

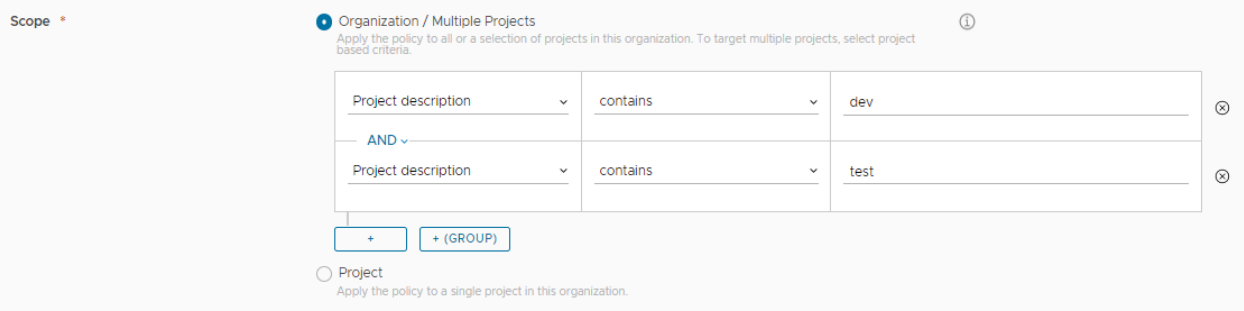
The scope options are the same for all policy types. After you create a policy, you cannot change the scope.

The following table provides more information about the application of each scope option.

Option	Project Criteria	Application
Organization/Multiple Projects	No	Organization. If no project criteria is defined, the policy is applied to all deployments in the organization.
	Yes	Multiple projects. If you define project specific criteria, the policy is applied to deployments associated with the projects that meet the specified criteria.
Project	Select a project.	Single project. The policy is applied only to deployments associated with the project that you select.

Setting policy scope to multiple projects

If you want to apply a policy to multiple projects in your organization, you set the policy scope to Organization/Multiple Projects and specify project specific criteria.



When you define project criteria, you can filter projects based on project name, description, and ID.

Project criteria work in the same way as deployment criteria. For more information about constructing criteria, see How do I configure deployment criteria in Service Broker policies.

The following table gives more information about how you can use each property to refine the scope of your policy.

Property	Supports these operators	Example

Project description	■ equals
	■ not equal to
	■ matches Regex
	■ contains
		You create a policy and you want to limit the application to developer projects in the organization that you administer.
		You set the scope to Organization/Multiple Projects and you add a project description expression that looks like the following example.
		`Project description contains dev`
		`AND`
		`Project description contains test`

Project ID	■ equals
	■ not equal to
		You want to apply a policy only to two or three projects, you set the scope to Organization/Multiple Projects and you add a project ID expression that looks like the following example.
		`Project ID equals proj123`
		`OR`
		`Project ID equals proj456`
		`OR`
		`Project ID equals proj789`

Project name	■ equals ■ not equal to ■ matches Regex ■ contains	You want to limit a policy to test projects in your organization, you set the scope to Organization/Multiple Projects and you add a project name expression that looks like the following example. `Project name matches Regex (t|T)est.*`

How do I configure deployment criteria in Service Broker policies

The deployment criteria narrows the scope of a policy so that it is applied only to the deployments where the criteria is true. For example, you can use the deployment criteria to create a policy that is applied only to a particular catalog item or template.

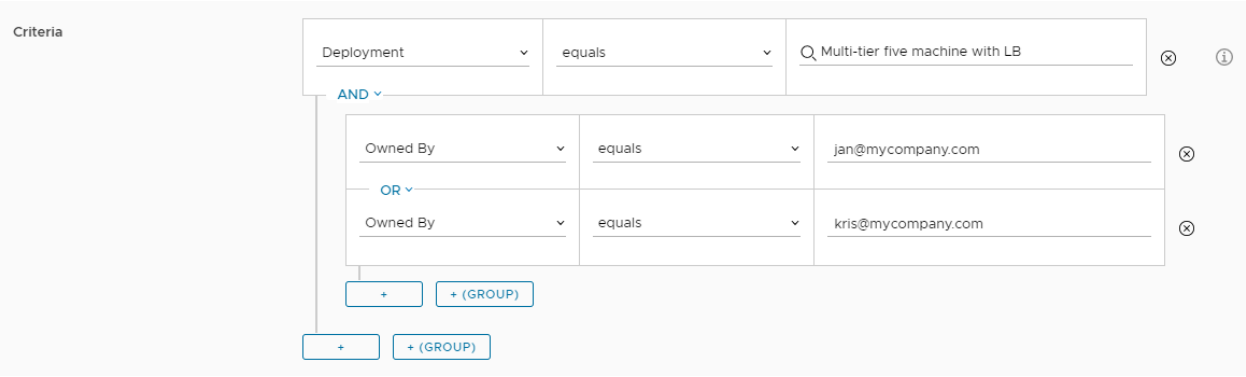
Constructing deployment criteria

You use the graphical interface to construct the deployment criteria expression. To construct complex expressions, you can use AND and OR. You can also group expressions as parenthetical operators. For more about how the expressions are processed, see Order of operations for the expression.

Here is an example of an expression.

```
Deployment equals Multi-tier five machine with LB AND (Owned By equals jan@mycompany.com OR
Owned By kris@mycompany.com)
```

Using the deployment criteria components, it looks like the following example.



Deployment criteria properties

To create a functional deployment criteria, you must understand the syntax.

The criteria text box has various drop-down menus that provide the available properties and operators. How you construct your expression depends on the available values and on the order of operations.

The drop-down menus include the following properties. Some properties vary between policy types.

Property	Description	Available in these policy types	Supports these operators
`Cloud Template`	Identifier for the Cloud Assembly cloud template that was used to create the deployment. Use `Cloud Template` rather than `Catalog Item` when your policy is specific to Cloud Assembly cloud templates. For example, an Amazon Web Services template does not have a `Cloud Template`.	■ Approvals ■ Day 2 ■ Lease	■ equals ■ not equal to
`Catalog Item`	Identifier for the Service Broker catalog item that was used to request the deployment. Use `Catalog Item` rather than `Cloud Template` when your policy can include Service Broker catalog items based on any template, extensibility workflow, or other content type. For example, Cloud Assembly cloud templates and Amazon Web Services CloudFormation templates deployed from the catalog have a `Catalog Item`.	■ Approvals ■ Day 2 ■ Lease	■ equals ■ not equal to
`Deployment Creation Cost`	Cost value. If the deployment matches the specified cost expression, it triggers an approval flow.	■ Approvals	■ equals ■ not equal to ■ greater than ■ greater than or equal ■ less than ■ less than or equal
`Deployment`	Identifier for the deployment. Use `Deployment` when you want to apply the policy to existing deployments.	■ Approvals ■ Day 2 ■ Lease	■ equals ■ not equal to
`Created By`	Name of the user who requested the deployment. The format is username@mycompany.com. This user is the user who requested the deployment.	■ Day 2 ■ Lease	■ equals ■ not equal to ■ matches Regex ■ contains

Property	Description	Available in these policy types	Supports these operators
`Name`	Deployment name. Use `Name` rather than `Deployment` when you want to apply the policy to existing policies and policies that can be created in the future that match the specified deployment name expression.	■ Approvals ■ Day 2 ■ Lease	■ equals ■ not equal to ■ matches Regex ■ contains
`Owned By`	Name of the current deployment owner.	■ Approvals ■ Day 2 ■ Lease	■ equals ■ not equal to ■ matches Regex ■ contains

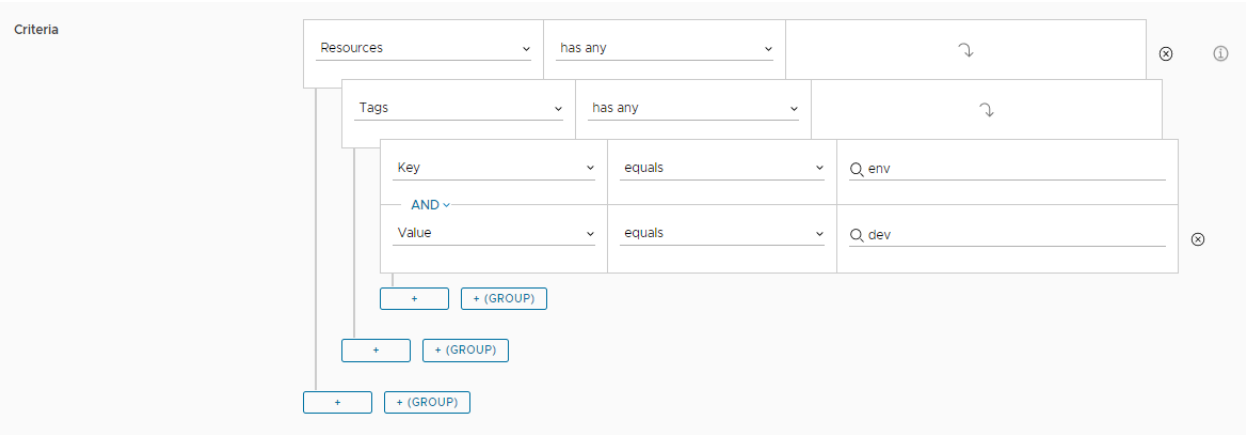
Property	Description	Available in these policy types	Supports these operators
`Requested By`	Name of the user who requested a day 2 action. The format is username@mycompany.com. When creating approval policies, the `Requested By` criteria is the user who requested a day 2 action, not the user who requested the deployment. The user who requested the deployment is the `Created By` criteria.	■ Approvals	■ equals ■ not equal to ■ matches Regex ■ contains
`Resources`	Resources that are part of a deployment. You can define the deployment criteria based on the following resources. ■ Cloud Zone ■ Cloud Account ■ CPU Count ■ Cloud Type ■ Disks ■ Flavor ■ Has Snapshots ■ Image ■ Image ID ■ OS Type ■ Power State ■ Region ■ Tags User-defined and discovered tags. ■ Total Memory (MB) ■ Resource Type	■ Approvals ■ Day 2 ■ Lease	

Criteria formats for resource tags

Resource tags are key value pairs. When you define deployment criteria based on the tags, you must define the key. Defining the value is optional. The criteria are based on user-defined tags and system tags.

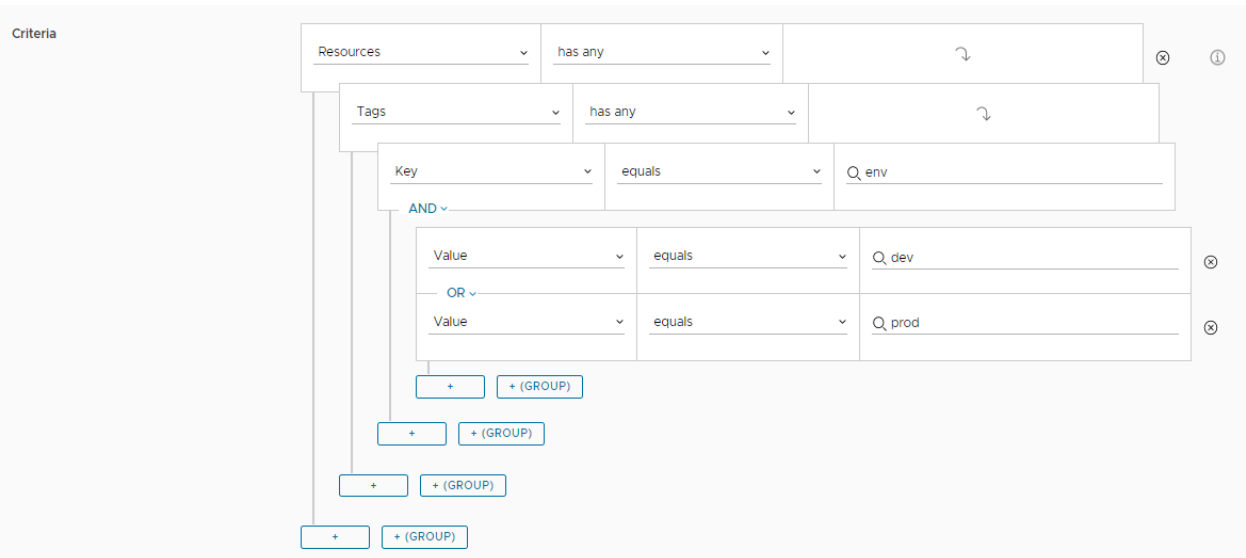
For example, to create criteria for one tag pair, the expression is similar to the following example.

```
Resources has any
    Tags has any
        Key equals env
        AND
        Value equals dev
```



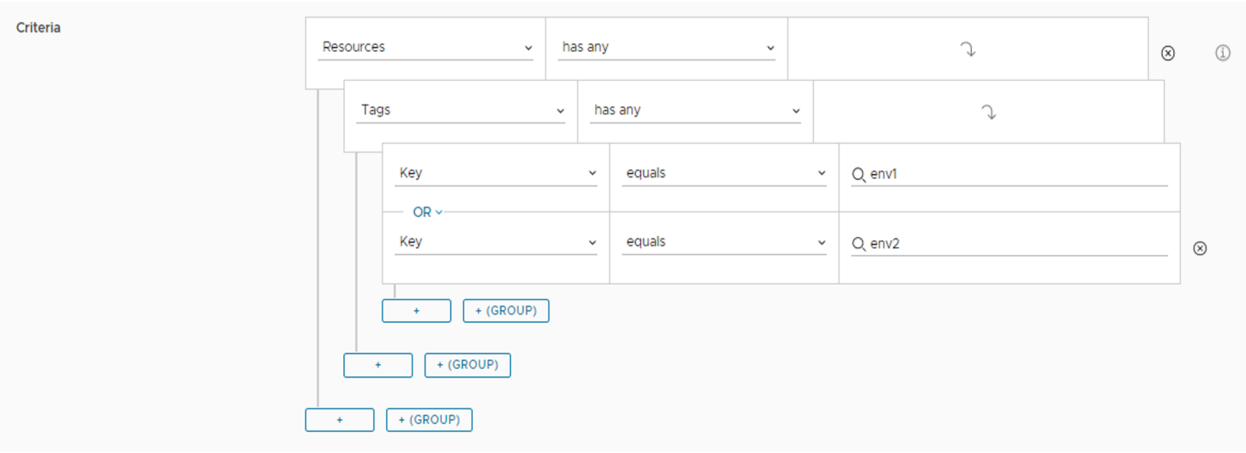
To create criteria based on one key but multiple values, the expression is similar to the following example.

```
Resources has any
    Tags has any
        Key equals env
            AND
                Value equals dev
                OR
                Value equals prod
```



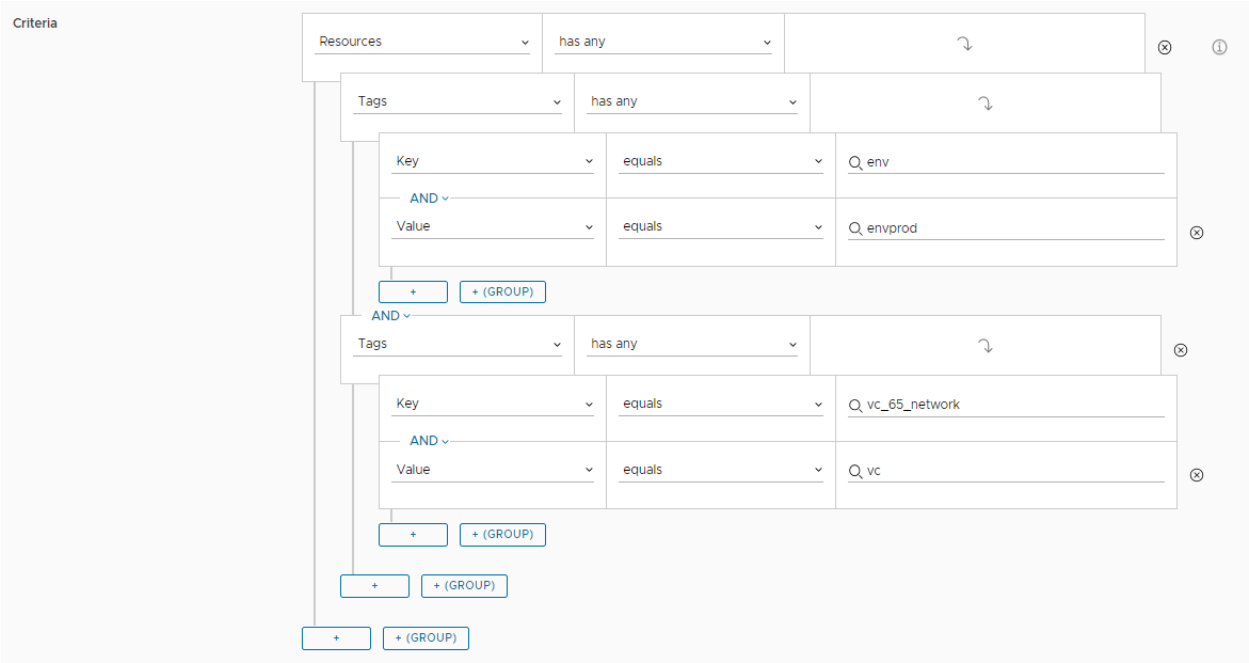
To create criteria based on multiple keys but no values, the expression is similar to the following example.

```
Resources has any
    Tags has any
        Key equals env1
        OR
        Key equals env2
```



If you want to create criteria that evaluate two different key value pairs, then you must add them as individual resource tags. For example,

```
Resources has any
    Tags has any
        Key equals env
        AND
        Value equals envprod
    AND
    Tags has any
        Key equals vc_65_network
        AND
        Value equals vc
```



Using the *contains* and *matches Regex* operators

The `contains` and `matches Regex` operators define a search for a specified set of characters within a property. You can apply these operators to string based properties that do not support a drop-down, such as `createdBy`, `name`, and `ownedBy`.

The `contains` operator searches for all instances of the value you specify in any context. The value input text box is case sensitive and space sensitive. If you want to account for context variation, you must set a value for each additional variant. Use the `contains` operator for simple searches for a limited number of values.

The `matches Regex` operator provides great flexibility when you use it for complex searches that must account for a lot of context variation. The regular expressions must follow ECMAScript syntax. When defining regular expressions, do not enter the forward slashes (`/`) at the beginning and at the end of the value.

The following table provides examples of expressions using the two operators and compares how they might be used to achieve the same goal.

Example with the `contains` operator	Example with the `matches Regex` operator	Field value matches
`Name contains test`	`Name matches Regex test*`	All deployment names that contain *test* in lowercase. For example, *test deployment*, *mytest*, *test-123*, and so on.

`Name contains test`
`OR`
`Name contains Test`
	`Name matches Regex (t|T)est.*`	All deployment names that contain *test* or *Test*.

`(group)`
`  Created By contains admin@`
`  (group)`
`  AND`
`  Created By contains .com`
`  OR`
`  Created By contains .org`
`(group)`
`AND`
`  Name contains test`
`  OR`
`  Name contains test-`
`  OR`
`  Name contains Test`
`  OR`
`  Name contains Test-`
`  OR`
`  Name contains deploy`
`  OR`
`  Name contains Deploy`
	`Created By matches Regex admin@\S+\.((com)|(org))`
	`AND`
	`Name matches ((t|T)est)|(d|D)epl.*.`
		All deployments that are created by users whose email address starts with *admin@* and ends with *.com* or *.org*.
		All deployment names that contain *test* and/or *deploy* in any configuration. For example, *test deployment*, *testdeployment*, *Test-Deployment*, and so on.

Order of operations for the expression

An expression is processed in the following order. Groups are illustrated as parentheses.

1 Expressions in groups

2 AND

3 OR

Use the following examples to understand the order.

- X OR Y AND Z. In this example, Y AND Z is evaluated before X OR Y. Next, the X OR is evaluated against the results of Y AND Z.
- (X OR Y) AND Z. In this example, X OR Y is evaluated before AND because the expression in the group is always evaluated first. Next the AND Z is evaluated against the results of X OR Y.

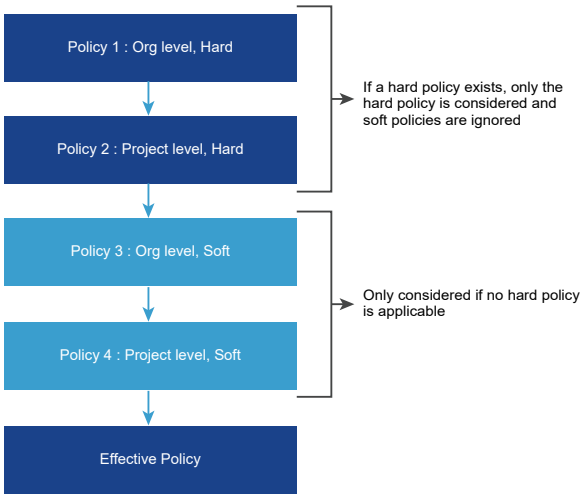
How are Service Broker policies processed

Policies are processed based on the policy definition. In particular, the scope and the enforcement level determine which policy is valid when you have multiple policies that might apply to a single deployment.

This article provides general information about policy processing, but it also includes more details for the different types of policies.

How policies are ranked based on organization level and enforcement type

When a user, who is a member of a project, creates a deployment, there might be more than one policy that applies to that deployment.



To evaluate the policies, the system first identifies and ranks policies.

1. Are there any hard policies at either the organization and project level. If there are hard and soft policies, then only the hard policies are considered and ranked. If there are only soft policies, then the soft policies are ranked.
2. The ranking of all hard or all soft policies is ordered by scope, with organizational policies having a higher rank than project policies.
3. The final discriminating characteristic is the creation date, with older dates be ranked higher than newer dates.

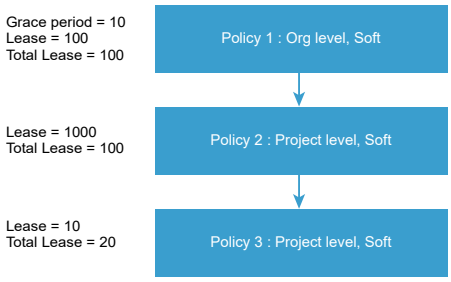
How policies are processed based on organization level and enforcement type

The policies are evaluated, ranked, and, where applicable, merged to produce an effective policy. An effective policy produces the intended results but is not always a specific named policy.

This section includes the following examples:

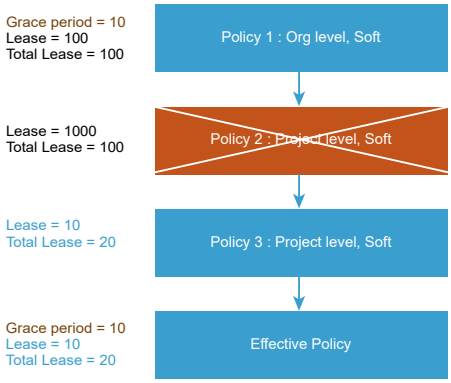
- Lease policies
- Day 2 actions policies

Review the following lease policy examples.



After identifying the policies to be considered and ranking them, the policies are then evaluated to identify the merge order.

- The highest ranking policy becomes the baseline. The second-level policy is applied on top of it, and so on.
- If a policy is incompatible with the preceding policies, then it is discarded from consideration. For example, the values are higher they the preceding policies.
- Any policy that is discarded is ignored. To see which policy is applied, select **Content and Policies > Policies > Enforcement**, locate the deployment, and review the decision notes.



Rather than applying one policy and excluding all the others above, the policies are merged and might include values from more than one individual policy.

In this example, the merging process excludes Policy 2 from consideration because the values are higher than Policy 1.

Next, Policy 3 is evaluated against Policy 1. The Lease and Total Lease values in Policy 3 are lower than Policy 1, so those values, in addition to the Grace period, become part of the effective policy.

Review the following day 2 actions policy examples.

- After identifying the policies to be considered and ranking them, the policies are then evaluated to identify the merge order.
  - The highest ranking policy becomes the baseline. The second-level policy is applied on top of it, and so on.
  - If a policy is enforced by preceding policies, for example, policy 3, then it is discarded from consideration.
  - Any policy that is discarded is ignored. To see which policy is applied, select **Content and Policies > Policies > Enforcement**, locate the deployment, and review the decision notes.

Lease policy management goal considerations

Now that you know how lease policies are processed, identify your policy management goals. By understanding how the policies are processed, you can meet your management goals without creating an excessive and unmanageable number of policies.

When deciding how to implement your policies, consider the following scenarios.

- Lease policy goals and enforcement examples
- Day 2 policy goals and enforcement examples

Table 3-1. Lease policy goals and enforcement examples

Management goal	Configuration Example	Behavior
Meaningful default organization-level policy that still allows the project-level policy values to influence the applied values.	Organization policy = Soft ■ Grace period: 10 ■ Lease: 100 ■ Total Lease: 100 Project 1 policy 1= Soft ■ Lease: 20 ■ Total Lease: 50 Project 2 policy 1= Soft ■ Lease: 10 ■ Total Lease: 30	A member of project 1 requests a catalog item. Project 2 is not considered because it is not applicable to project 1 deployments. The merged effective policy is: ■ Grace period: 10 ■ Lease: 20 ■ Total Lease: 50
Always default to the organization-level policy.	Organization policy = Hard ■ Grace period: 10 ■ Lease: 100 ■ Total Lease: 100 Project 1 policy 1= Soft ■ Lease: 20 ■ Total Lease: 50	A member of project 1 requests a catalog item. Project 1 policy 1 is not considered because the hard organization level project is a higher rank and the soft policy is not considered. The effective policy is: ■ Grace period: 10 ■ Lease: 100 ■ Total Lease: 100
All policies are defined at the project-level, with no organization-level default policy.	Project 1 policy 1 = Soft ■ Grace period: 10 ■ Lease: 100 ■ Total Lease: 100 Project 1 policy 2= Soft ■ Lease: 20	A member of project 1 requests a catalog item. They are both soft policies, and they are both for project 1. The values are merged. The effective policy is: ■ Grace period: 10 ■ Lease: 20 ■ Total Lease: 100

The day 2 actions policies are used in these examples.

Table 3-2. Day 2 policy goals and enforcement examples

Management goal	Configuration Example	Behavior
Meaningful default organization-level policy that still allows the project-level policy values to influence the applied values.	Organization policy = Soft ■ Actions : Deployment.* Project 1 policy 1= Soft ■ Actions: Cloud.vSphere.Machine.* Project 2 policy 1= Soft ■ Actions: Cloud.Azure.Machine.*	A member of project 1 requests a catalog item. Project 2 is not considered because it is not applicable to project 1 deployments. The merged effective policy is: ■ Action : {Deployment.* ,Cloud.vSphere.Machine.*}
Always default to the organization-level policy.	Organization policy = Hard ■ Action : Deployment.* Project 1 policy 1= Soft ■ Action : Cloud.vSphere.Machine.*	A member of project 1 requests a catalog item. Project 1 policy 1 is not considered because the hard organization level project is a higher rank and the soft policy is not considered. The effective policy is: ■ Action : {Deployment.* }
All policies are defined at the project-level, with no organization-level default policy.	Project 1 policy 1 = Soft ■ Actions : Deployment.ChangeLease Project 1 policy 2= Soft ■ Action : Deployment.Delete	A member of project 1 requests a catalog item. They are both soft policies, and they are both for project 1. The values are merged. The effective policy is: ■ Action : {Deployment.ChangeLease , Deployment.Delete}

Approval policy goals and enforcement examples

The approval policy evaluation follows this process.

1 A request for a deployment or day 2 action is submitted.

2 The approval service queries for policies that apply to the project that is requesting a catalog item or changing a deployed item.

3 All the applicable project- and organization-level scope policies are returned.

4 The approval policies are filtered based on the deployment criteria. Deployment criteria apply to deployments and day 2 actions.

5 If no matching policies are found, no approval is required and the deployment process proceeds.

6 If there are matching policies, for example, AP1, AP2, APn, then an approval item is created as:

- Enforced policies = AP1, AP2, APn.
- Approvers = A union of all the approvers in all the enforced policies.

- Auto expiry = Reject, if any policy has a reject value; otherwise, approve.
- Expiry = Minimum number of days of any of the enforced policies.

The following table provides a sample of multiple policies. The description of how they are processed is below the table.

Policy	Configuration example
AP1	Scope = Organization Auto expiry = Approve Expiry = 7 days
AP2	Scope = Project 1 Auto expiry - Approve Expiry = 3 days
AP3	Scope = Project 1 Auto expiry = Reject Expiry = 4 days
AP4	Scope = Project 2 Auto expiry = Approve Expiry = 5 days

Based on the policies and configuration examples above, the following information explains how a Project 1 request is processed.

1 The scope evaluation returns AP1, AP2, and AP3. AP4 is not included because it is a Project 2 policy.

2 Assuming that AP1, AP2, and AP3 satisfy the deployment and action criteria, then the approval item includes the following values:

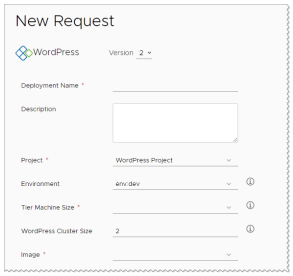
   - Approvers = Any or all the approvers from AP1, AP2, and AP3 are added as approvers.
   - Auto expiry = Reject. AP3 provides the more restrictive behavior.
   - Expiry = 3 days. AP2 provides the lowest value.

Customize a Service Broker icon and request form

In Service Broker, you can customize the icon that represents the content in the catalog, limit the number of deployed instances for a catalog item, and customize the request form for imported templates. When customizing the request form, you can also design the input parameters that allow the user requesting a catalog item to provide the values. You can customize how the custom options are presented in the form.

The icon that you provide helps you and your catalog consumers use visual queues to identify specific items. You are not required to customize a form if all you want is a custom icon. Nor are you required to customize the icon when you create a custom form.

When creating the custom form, the WordPress cloud template is used as the example in this use case. If you don't customize the request form, it is a simple list of parameters. See the following example.



In this use case, you customize the following options:

- Reduce the maximum number of WordPress Cluster Size from 5 to 3.
- Specify operating system based on Node Size. For example, if size is small, then the operating system is coreos. If it medium, then the operating system is ubuntu.
- Set the MySQL Data Disk Size value to 5 and hide the option from the requesting users.

Prerequisites

- To add an icon, verify that you have an image that does not exceed 100 KB. The optimal size is no larger than 100x100 pixels.
- This use case assumes that you imported the WordPress use case cloud template from Cloud Assembly, or that you have a cloud template or template that includes input parameters.

Procedure

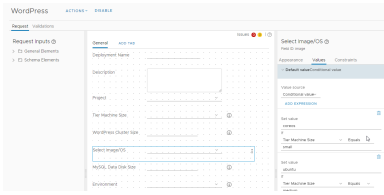
1. Select **Content and Policies > Content**.
2. Locate the WordPress cloud template, click the menu to the left of the name, select **Configure item**.
   a. Set the maximum number of deployment instances for this catalog item.

      If you select a value greater than one, the **Deployment count** field is added to the request form. This option allows the requesting user to do bulk deployments.

   b. Add a custom icon.

   If all you want is a custom icon, you can stop here.
3. Locate the WordPress cloud template, click the menu to the left of the name, and select **Customize form**.

   If the cloud template has input properties, they are listed in the Request Inputs pane on the left, and are added to the canvas.



4 Edit the form using the values provided in the following table.

For this field in the screenshot	Appearance	Values	Constraints
WordPress Cluster Size			Maximum value ■ Value source = Constant ■ Max value = 3
Select Image/OS		Default value ■ Value source = Conditional value ■ Expression = Set value = coreos If Tier Machine Size Equals small ■ Expression = Set value = ubuntu If Tier Machine Sieze Equals medium	
MySQL Data Disk Size	Visibility ■ Value source = Constant ■ Visible = No	Default value ■ Value source = Constant ■ Default value = 5	

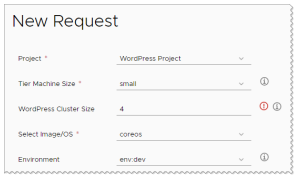
5 Click and drag the fields to rearrange them on the form.

6 To turn on the custom form, click **Enable**.

7 Click **Save**.

Results

The request form is now similar to the following example.



Notice that the Wordpress Cluster Size field indicates an error. The limit is 3, but the user entered a value of 4.

What to do next

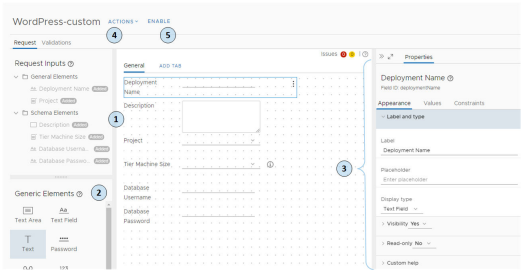
Request the item in the catalog and verify that the presentation and behavior is what you expected.

Learn more about Service Broker custom forms

To create useful forms based on input parameters, you can use Service Broker to design how the information appears at request time, how the parameters values are populated, and add any specialized constraints.

Custom request form designer

You use the form designer to create your custom form.



To create a custom form:

1 Notice that request inputs that are already on the canvas.

2 Drag any custom elements onto the design canvas.

3 Configure each element using the properties pane.

   For more about the fields properties, see Custom form designer field properties in Service Broker.

4 Use the Actions menu options to import or export the form, or import or export a CSS file. The following sections provide more information.

5 Enable the form.

The custom form designer supports data validation by adding constraints to a field. For constraints options that are applied as you create a form, see Custom form designer field properties in Service Broker. For a constraint example, see Customize a Service Broker icon and request form.

Catalog items can have a single custom form at a time. If you edit a catalog item, for example a cloud template, that already has a custom form defined, the changes are not reflected in the custom form. To be able to see the changes that you made to the cloud template, you must delete the old custom form, and create a new one.

Importing and exporting custom forms between templates

You might find, after you develop a custom form, that you want to use part or all of it with another template. You can export a form from one template and import it into another template, and then continue customizing the form for the new template.

To share the custom forms, you can click **Actions** on the custom form designer and select one of the following options.

Table 3-3. Action menu options for importing and exporting custom forms

Action Menu Item	Description
Import form	Imports a JSON or YAML file.
Export form	Exports your current custom form as a JSON file.
Export form as YAML	Exports your current custom form as YAML. Export the file as YAML when you want to move a custom from one Service Broker instance to another. For example, from your test environment to your production environment. If you prefer to edit the form as YAML, you can export the form, edit it, and then import it back into the template.

Adding your own style sheet to a custom form

You can use a custom cascading style sheet to refine how the text appears on the screen. You must create the CSS file outside of Service Broker. But you can export and import a CSS file from one template to another.

Table 3-4. Action menu options for importing and exporting CSS files

Action Menu Item	Description

Import CSS	Imports a CSS file that enhances the catalog request form.
	The file might be similar to the following example.
	`#<field_ID> {`
	`  font-size: 20px;`
	`  font-weight: bold;`
	`  color: red;`
	`  width: 600px;`
	`}`
	`#<field_ID> {`
	`  font-size: 20px;`
	`  font-weight: bold;`
	`  font-style: italic;`
	`  width: 600px;`
	`}`
	In this example, replace `<field_ID>` with the actual field IDs from the custom form. You can locate the values by selecting the field in the form, and then you can see the value in the properties pane, beneath the field name. For example, Field ID: `deploymentName` or Field ID: `textField_fe7cf66a`.

Export CSS	Exports your customized CSS.
Remove CSS	Discards your custom CSS. The discarded CSS is not recoverable.

Custom form designer field properties in Service Broker

The field properties in Service Broker determine how the fields looks and what default values are presented to the user. You can also use the properties to define rules that ensure that the users provide a valid entry when they request the item in the catalog.

You configure each field individually. Select the field and edit the field properties.

Value source

For many of the properties, you can select from various value source options. Not all source options are available for all field types or properties.

- **Constant.** The value does not change. Depending on the property, the value might be a string, an integer, a regular expression, or selected from a limited list, for example, Yes or No. For example, you can provide 1 as a default value integer, select No for the Read-only property, or provide the regular expression to validate a field entry.
- **Conditional value.** The value is based on one or more conditions. The conditions are processed in the order listed. If more than one condition is true, the last condition that is true determines the behavior of the field for that property. For example, you can create a condition that determines if a field is visible based on the value in another field.

- **External source.** The value is based on the results of a vRealize Orchestrator action. For example, calculate cost based on a scripted vRealize Orchestrator action. For an example, see Using vRealize Orchestrator actions in the custom form designer in Service Broker
- **Bind field.** The value is the same as the field to which it is bound. The available fields are limited to the same field type. For example, you bind default value for an authentication needed check box field to another check box field. When one target field check box is selected in the request form, the check box on the current field is selected.
- **Computed value.** The value is determined based on how the operator processes the selected fields and values. Text fields use the concatenate operator. Integer fields use the selected add, subtract, multiply or divide operations. For example, you can configure an integer field to convert megabytes to gigabytes using the multiply operation.

Field appearance

You use the appearance properties to determine whether the field appears on the form and what label and custom help you want to provide to your catalog users.

Table 3-5. Appearance Tab Options

Option	Description
Label and type	Provide a label and select a display type. The available display types depend on the element. Some some elements support multiple text types and others only support integers. Possible values: ■ Array Input ■ Checkbox ■ Combobox ■ Data Grid ■ Date Time ■ Decimal ■ Drop Down ■ Dual List ■ Image ■ Integer ■ Link ■ Multi Select ■ Multi Value Picker ■ Object Field ■ Password (Additional information below regarding password encryption.) ■ Radio Group ■ Text ■ Text Area ■ Text Fields ■ Value Picker Drop-down and data grid fields include a **Placeholder** setting. The entered value appears as an internal label or instructions in the drop-down menu, or as a general label or instructions in the data grid. To ensure that passwords are encrypted in the deployment request details page, the input property in the cloud template must include `encrypted:true`.
Visibility	Show or hide a field on the request form. ■ **Constant.** Select Yes to display the field on the form. Select No to hide the field. ■ **Conditional value.** Visibility is determined by the first expression that is true. For example, a field is visible if a check box is selected on a form. ■ **External source.** Visibility is determined by the results of the selected vRealize Orchestrator action.

Table 3-5. Appearance Tab Options (continued)

Option	Description
Read-only	Prevent users from changing the field values. ■ **Constant.** Select Yes to display the value but prevent changes. Select No to allow changes. ■ **Conditional value.** Status is determined by the first expression that is true. For example, a field is read-only if the value in a storage field is greater than 2 GB. ■ **External source.** Status is determined by the results of the selected vRealize Orchestrator action.
Rows per page	For data grid elements only. Enter the number of rows.
Custom help	Provide information about the field to your users. This information appears in signpost help for the field. You can use simple text or HTML, including href links. For example, `<a href="https://docs.vmware.com/en/VMware-Service-Broker/index.html">VMware Service Broker documentation</a>`.

Field values

You use the values properties to provide any default values.

Table 3-6. Values Tab Options

Option	Description
Columns	For the data grid element only. Provide the label, ID, and value type for each column in your table. The default value for the data grid must include the header data that matches the defined columns. For example, if you have user_name ID for one column and user_role ID for another, then the first row is user_name,user_role. For configuration examples, see Using the data grid element in the Service Broker custom form designer.
Default value	Populates the field with a default value based on the value source. Possible value sources depend on the field. ■ **Constant.** The entered string. ■ **Conditional value.** The default value is determined by the first expression that is true. For example, the default value of a storage field is 1 GB if the memory field is less than 512 MB. ■ **External source.** Value is based on the results of the selected vRealize Orchestrator action. ■ **Bind field.** Value is the same as the selected field. ■ **Computed value.** Value is based on the results of the provided field values and the selected operator. For example, the default value of memory in MB is based on the memory in GB multiplied by 1024.
Value option	Populates a drop-down, multi-select, radio group, or value picker fields. ■ **Constant.** The format for the list is Value|Label,Value|Label,Value|Label. For example, `2|Small,4|Medium,8|Large`. ■ **External source.** Value is based on the results of the selected vRealize Orchestrator action.
Step	For integer or decimal fields, define the incremental or decremental values. For example, if the default value is 1 and you set the step value to 3, then the allowed values are 4, 7, 10, and so on.

Field constraints

You use the constraint properties to ensure that the requesting user provides valid values in the request form.

Table 3-7. Constraints Tab Options

Option	Description
Required	The requesting user must provide a value for this field. ■ **Constant.** Select Yes to require that the requesting user provides a value. Select no if the field is optional. ■ **Conditional value.** Whether the field is required is determined by the first expression that is true. For example, this field is required if the operating system family starts with Darwin in another field. ■ **External source.** Status is based on the results of the selected vRealize Orchestratoraction.
Regular expression	Provide a regular expression that validates the value and a message that appears when the validation fails. The regular expressions must follow JavaScript syntax. For an overview, see Creating a regular expression. For more detailed guidance, see Syntax. ■ **Constant.** Provide a regular expression. For example, for an email address, the regular expression might be `^[A-Za-z0-9._%+-]+@[A-Zaz0-9.-]+\\.[A-Za-z]{2,}$` and the validation error message is `The email address format is not valid. Please try again.` ■ **Conditional value.** The regular expression that is used is determined by the first expression that is true.
Minimum value	Specify a minimum numeric value. For example, a password must have at least 8 characters. Provide an error message. For example, `The password must be at least 8 characters.` ■ **Constant.** Enter the integer. ■ **Conditional value.** The minimum value is determined by the first expression that is true. For example, a minimum CPU value is 4 if the operating system does not equal Linux. ■ **External source.** Value is based on the results of the selected vRealize Orchestrator action.

Table 3-7. Constraints Tab Options (continued)

Option	Description
Maximum value	Maximum numeric value. For example, a field is limited to 50 characters. Provide an error message. For example, `This description cannot exceed 50 characters.` ■ **Constant.** Enter the integer. ■ **Conditional value.** The maximum value is determined by the first expression that is true. For example, a maximum storage value is 2 GB if the deployment location equals AMEA. ■ **External source.** Value is based on the results of the selected vRealize Orchestrator action.
Match field	This field value must match the selected field value. For example, a password confirmation field must match the password field.

Using the data grid element in the Service Broker custom form designer

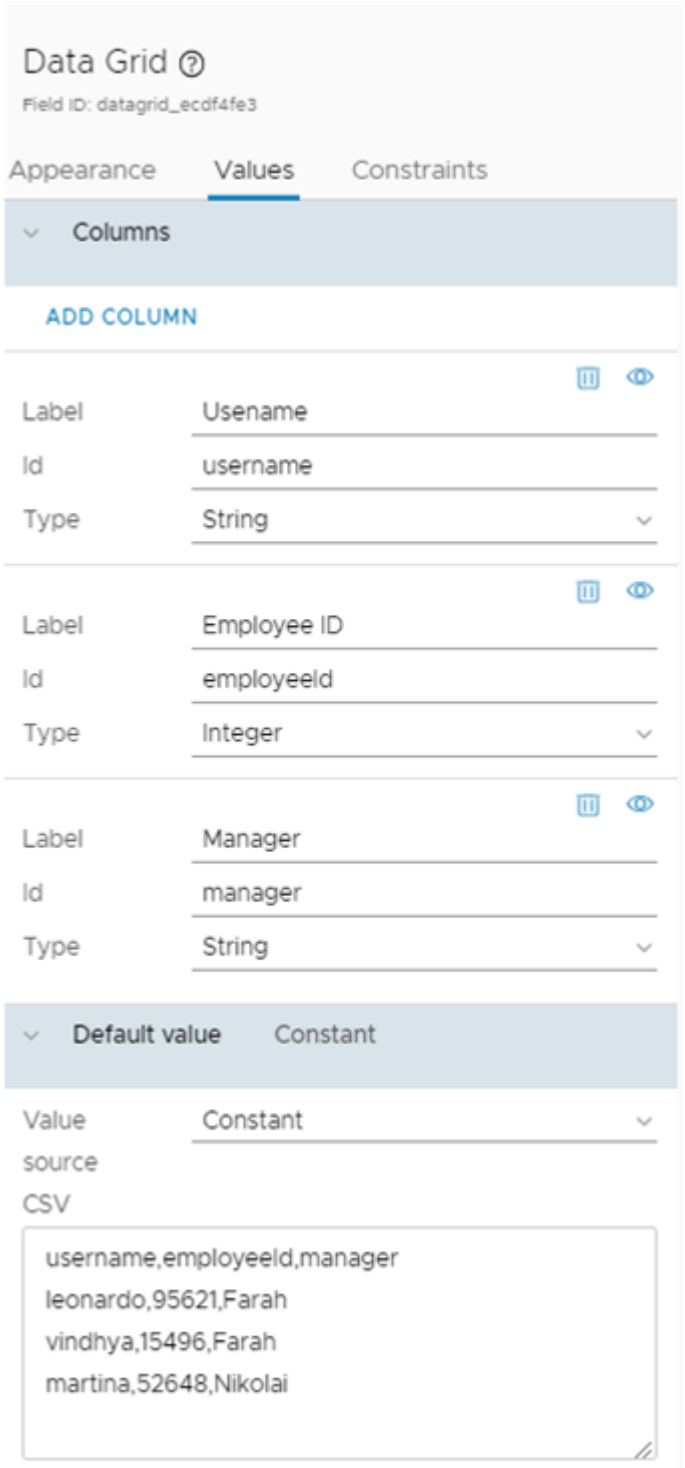
If you use a data grid element in a custom form, the data that is presented in the table might be manually provided.

Example: Provided CSV data example

In this use case, you have a table of values that you provide in the custom request form. You provide the information in the table as a constant value source. The source is based on a CSV data structure where the first row defines the grid headers. The headers are the column IDs separated by a comma. Each additional row is the data that appears in each row in the table.

1 Add the Data Grid generic element to the design canvas.

2 Select the data grid and define the values in the properties pane.

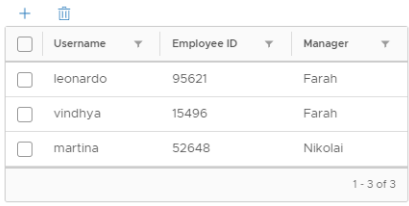


Label	ID	Type
Username	username	String
Employee ID	employeeId	Integer
Manger	manager	String

Define the CSV values.

```
username,employeeId,manager
leonardo,95621,Farah
vindhya,15496,Farah
martina,52648,Nikolai
```

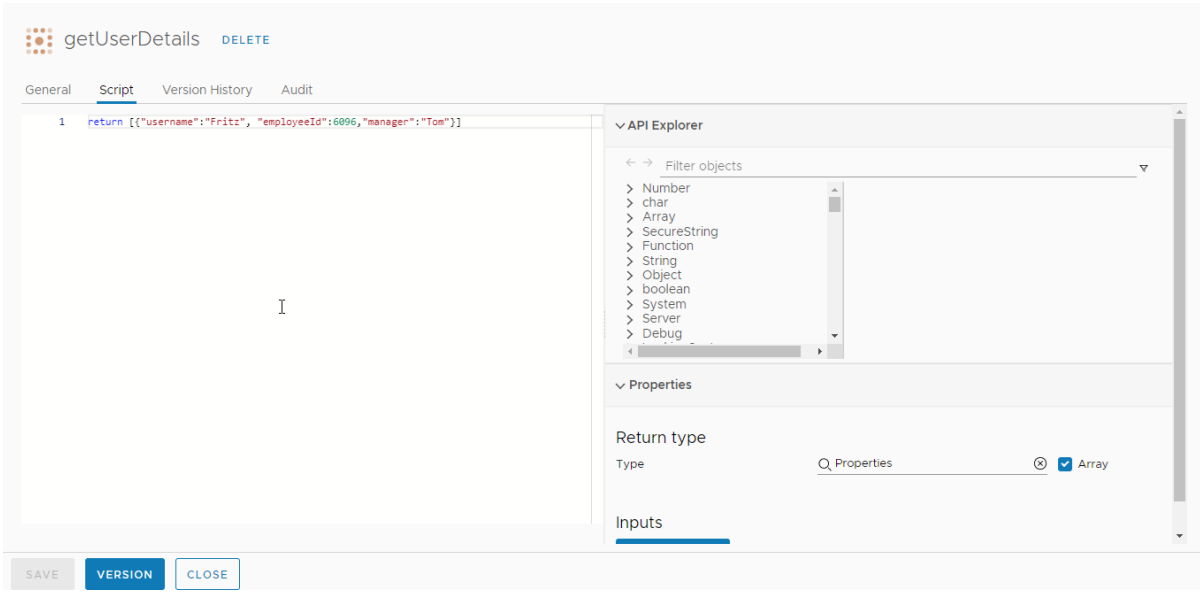
3 Verify that the data grid displays the expected data in the request form.



Example: External Source Example

This example uses the previous example but the values are based on a vRealize Orchestrator action. Although this is a simple action example, you can use a more complex action where you retrieve this information from a another database or system.

1 In vRealize Orchestrator, configure an action, `getUserDetails`, with an array similar to the following example.



a On the General tab, enter the name **`getUserDetails`** and provide a Module name.

b On the Script tab, use the following script example.

```
return [{"username":"Fritz", "employeeId":6096,"manager":"Tom"}]
```

c In the Return type area, enter or select **`Properties`** as the type, and click **Array**.

d Version and save the action.

2 In Service Broker, add the data grid and use the Values tab to configure the data grid columns with the following values.

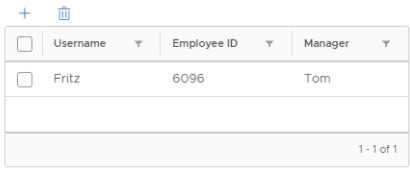
Label	ID	Type
Username	username	String
Employee ID	employeeId	Integer
Manger	manger	String

3 In the Default value, Value source list, select **External source**.

4 In Select action, enter **`getUserDetails`** and select the action you created in vRealize Orchestrator.

5 Save the form.

6 In the catalog, verify the table in the request form.



Using vRealize Orchestrator actions in the custom form designer in Service Broker

When you customize a Service Broker request form, you can base the behavior of some fields on the results of a vRealize Orchestrator action.

There are several ways that you can use vRealize Orchestrator actions. You might have an action that pulls the data from a third source, or you can use a script that defines the size and cost.

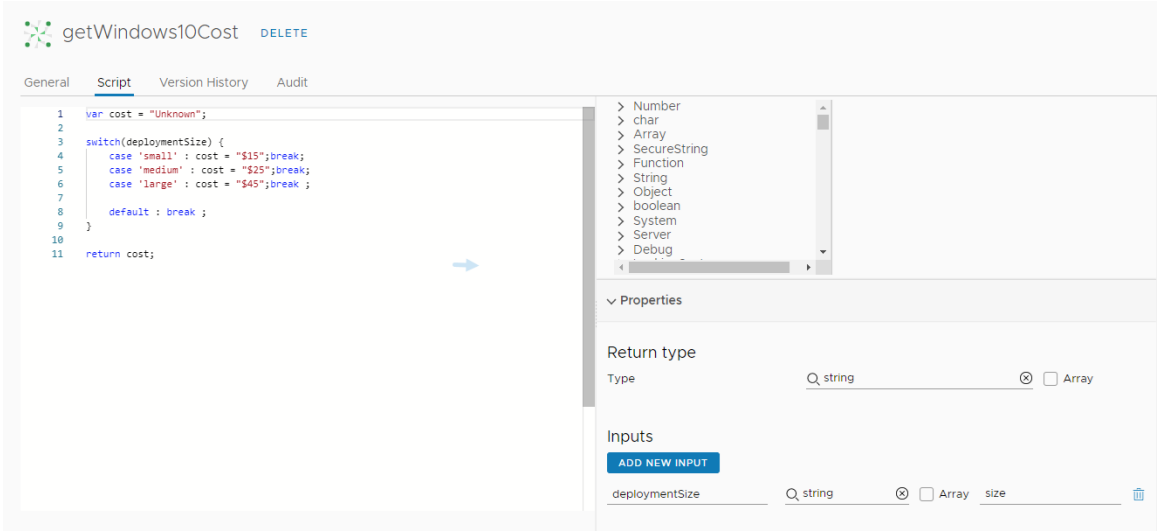
The first example is based on manually added fields so that you understand the underlying process. The second example uses the same premise, but instead relies on a template field.

In addition to the following examples, other examples are available in the VMware Cloud Management blog.

Example: Size and cost as manually added fields example

In this use case, you want the catalog user to select a virtual machine size, and then display the cost of that machine per day. To do this example, you have a vRealize Orchestrator script that correlates the size and cost. You then add a size field and a cost field to the template custom form. The size field determines the value that appears in the cost field.

1 In vRealize Orchestrator, configure an action named `getWindows10Cost`.



2 Add a script.

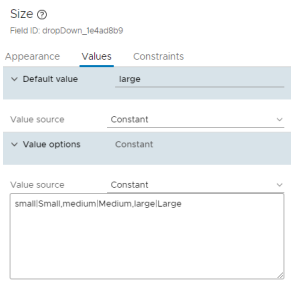
You can use the following example script.

```
var cost = "Unknown";

switch(deploymentSize) {
    case 'small' : cost = "$15";break;
    case 'medium' : cost = "$25";break;
    case 'large' : cost = "$45";break ;

    default : break ;
}

return cost;
```

3 Add `deploymentSize` as an input string.

4 In Service Broker, add and configure a Size field to a template custom form.

Configure the size field as a drop-down element with Small, Medium, and Large values.

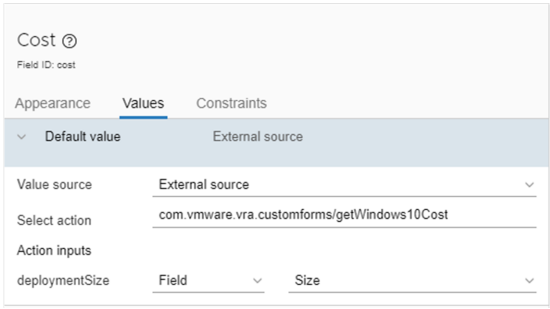


On the Values tab, configure the following property values.

- Default value = **`Large`**

- Value options
  - Value source = **`Constant`**
  - Value definition = **`small|Small,medium|Medium,large|Large`**

5 Add the cost field as a text field to display the cost as defined in the vRealize Orchestrator action based on the value selected in the size field.

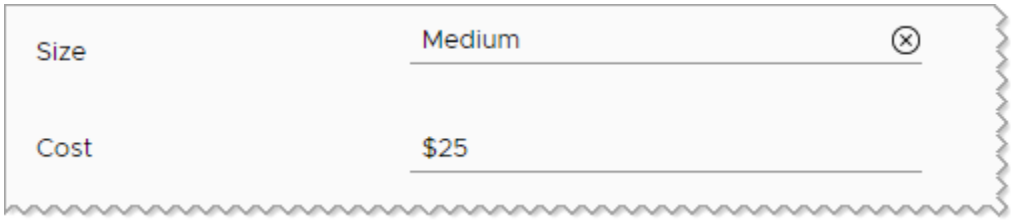


On the Values tab, configure the following property values.

- Default value = External source
- Select action = <your vRealize Orchestrator actions folder>/getWindows10Cost
- Action inputs
  - deploymentSize. This value was configured in the action as the input.
  - Field
  - Size. This is the field that you previously created.

6 Enable the custom form and save it.

7 To verify that it is working, request the item in the catalog. You should see the Cost field populated based on the selected Size value.



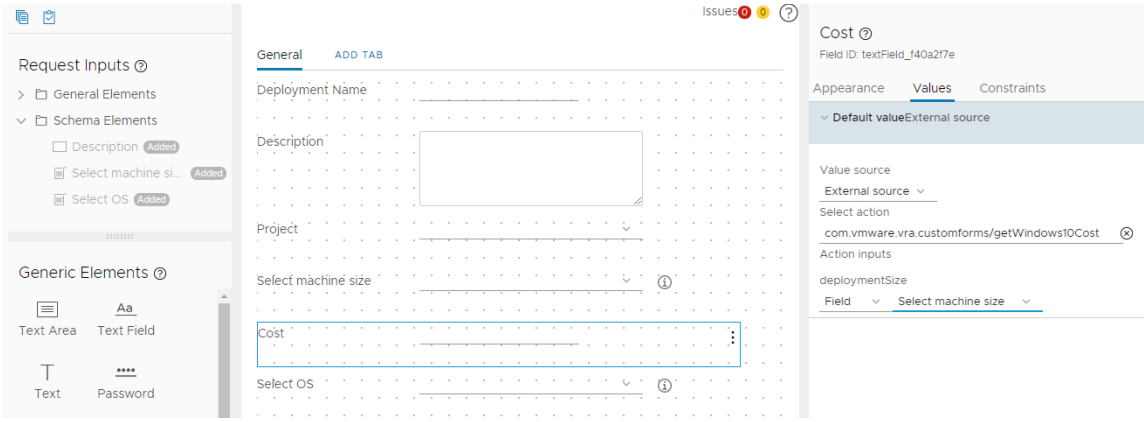
Example: Cost based on schema element example

In this use case, you want the catalog user to see the cost of that machine per day based on the flavor property in the template. To do this example, you use the vRealize Orchestratorscript from the previous example. But in this use case the cost is based on the flavor size that your user selected in the custom form when they request the Service Broker catalog item.

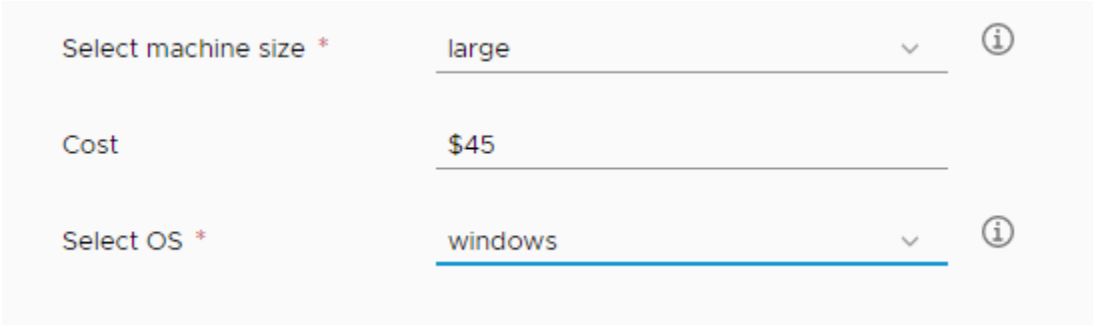
The simple example template includes a size input field where the user selects the flavor property.

```
formatVersion: 1
inputs:
  size:
    type: string
    enum:
      - small
      - medium
      - large
    description: Size of Nodes
    title: Select machine size
  image:
    type: string
    enum:
      - ubuntu
      - centos
      - windows
    description: OS image
    title: Select OS
resources:
  Cloud_vSphere_Machine_1:
    type: Cloud.vSphere.Machine
    properties:
      image: '${input.image}'
      flavor: '${input.size}'
```

The custom form uses the field, named `Select machine size` in this example.



The cost deploymentSize input is based on the Select machine size field.



Using value picker and multi value picker elements in the Service Broker custom form designer

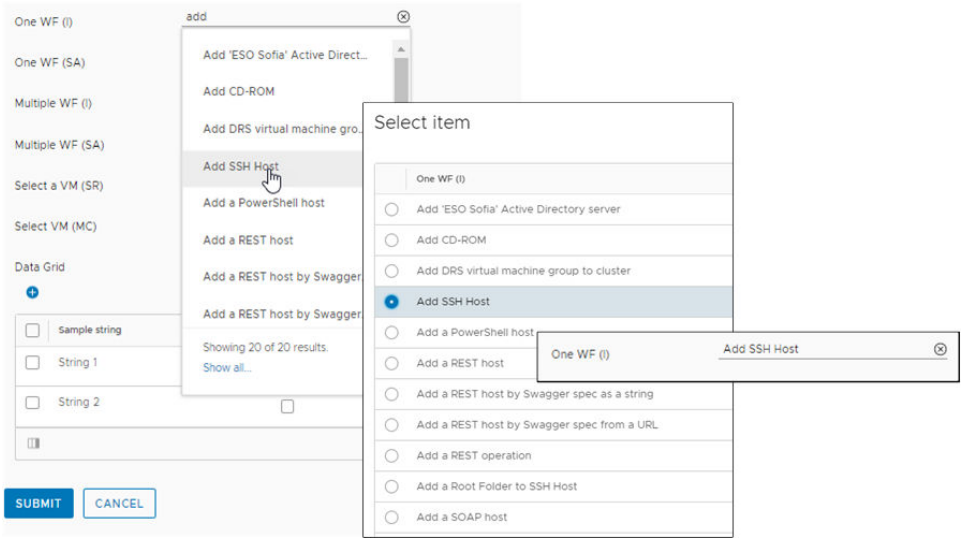
When you create a custom form, you can add elements where the user selects a value from a search results list. Using the value picker, the user selects a single value. Using the multi value picker, the user selects one or more values.

The value picker and multi value picker work with the Reference Type that is defined on the custom form Appearance tab. The Reference type is a vRealize Orchestrator resource. For example, AD:UserGroup or VC:Datastore. By defining the reference type, when the user enters a search string, the results are limited to the resources that have the matching parameter.

For the pickers, you can then further limit the possible values by configuring an external source.

Working with the Value Picker

The value picker appears in the form as a search option when users request the item in the catalog. The user enters a string and the picker provides list based on how you configured it.



You can use the picker based on the following use cases. The most valuable use of the value picker is pairing it with an external source value.

- Value picker with a constant value source.

  Use this method when you want the requesting user to select from a predefined static list of values. Similar to the combobox, drop down, multiselect, and radio group elements, this method provides search results in a list based on the defined constant values and labels.

- Value picker with no defined value source.

  Use this method when you want the requesting user to search the vRealize Orchestrator inventory for a specific object with the configured reference type. For example, the reference type is VC:Datastore and you want the users to select the datastore from the retrieved list.

- Value picker with an external value source.

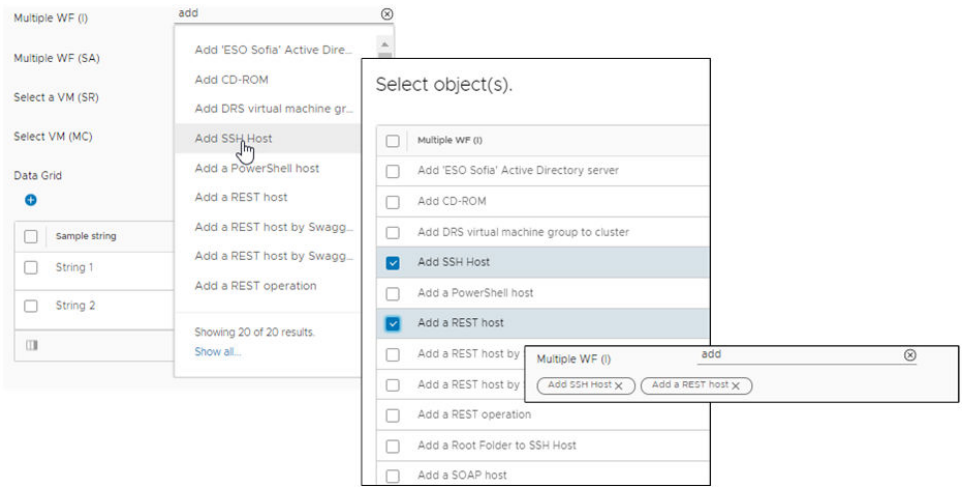
Use this method when you want the requesting user to select from results that are based on a vRealize Orchestrator action. For a value picker based on an external source, the action must return a properties array, not a string array. The following script provides an example of a basic vRealize Orchestrator action that works with the value picker.

```
var res = [];
res.push(new Properties({label: 'label1',value: 'value1'}));
res.push(new Properties({label: 'label2',value: 'value2'}));
res.push(new Properties({label: 'label3',value: 'value3'}));
return res;
```

Note The Properties input cannot be input to a workflow, only an intermediate value in the custom form.

Working with the Multi Value Picker

The multi value picker appears in the request form as a search option, similar to the value picker, but where you can select one or more values. The user enters a string and the picker provides list based on how you configured the element properties.



You can use the multi value picker based on the following use cases in addition to the use cases described for the value picker. The most valuable use of the multi value picker is using it with a reference data type and a vRealize Orchestrator reference.

- Multi value picker with a complex data type and constant value source.

  Use this method when you want the requesting user to select one or more values from a predefined static list of values. Similar to the data grid, this method provides search results in a list based on the defined constant values and labels.

- Multi value picker with a complex data type and an external source.

  Use this method when you want the requesting user to select one or more values from a list of values based on a vRealize Orchestrator action. You can use this method with vRealize Orchestrator composite types.

- Multi value picker with a reference data type and a vRealize Orchestrator reference type. Use this method when you want the requesting user to search the vRealize Orchestrator inventory for a specific object with the configured reference type. For example, the reference type is VC:Datastore and you want the users to select the datastore from the retrieved list. Or, if you have a workflow filter configured, you can use Workflow as the reference. To be retrieved, the filter must return values in a property array, not a string array. An example of a workflow filter is provided in the next section. In this example, the filtering is done in the UI when the user enters a search term.
- Multi value picker with a reference data type, a vRealize Orchestrator reference type, and an external source.

  Use this method when you want the requesting user to select from results that are first filtered by the reference type and then based on a vRealize Orchestrator action. This combination more thoroughly refines the results and populates the request form more quickly. Just as the reference type results must return a property array, so must the external source action. In this example, the filtering is done in vRealize Orchestrator and might improve the speed with which the list is populated, particularly if you have a large number of vRealize Orchestrator actions.

Limit the vRealize Orchestrator results for a multi value picker element results list

To limit the number of actions returned when the user searches for an action, you can create a filter action and bind the filter results to the search term.

1 In vRealize Orchestrator, create an action named filterWorkflow.

  a Select **Library > Actions**, and click **New Action**.

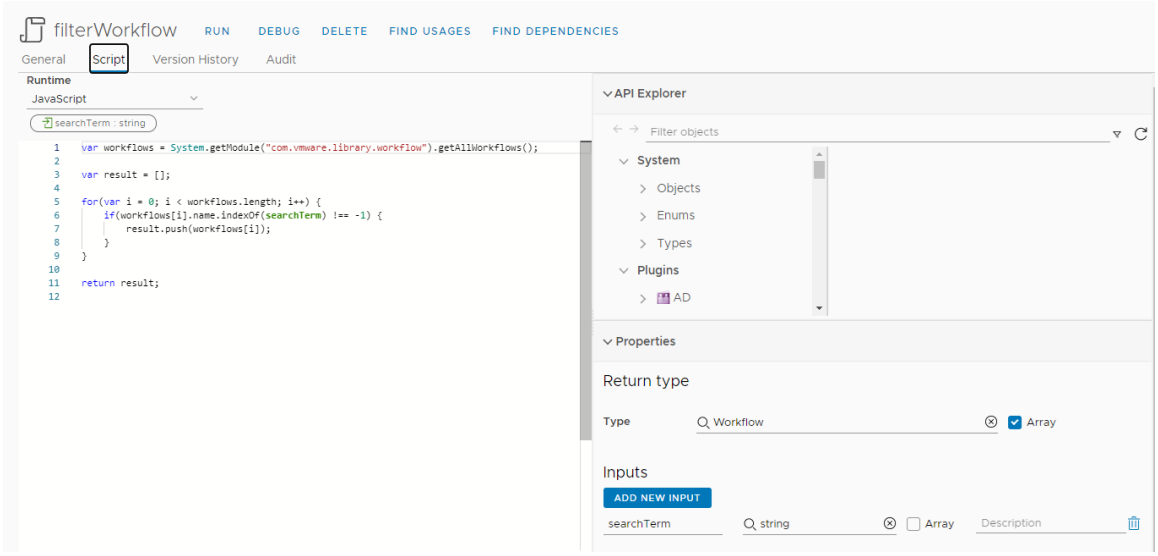
  b On the **General** tab, enter or select the following values.

Option	Value
Name	**filterWorkflow**
Module	**com.vmware.library.workflow**

  c Click the **Script** tab and add the following script.

```
var workflows = System.getModule("com.vmware.library.workflow").getAllWorkflows();

var result = [];

for(var i = 0; i < workflows.length; i++) {
    if(workflows[i].name.indexOf(searchTerm) !== -1) {
        result.push(workflows[i]);
    }
}

return result;
```

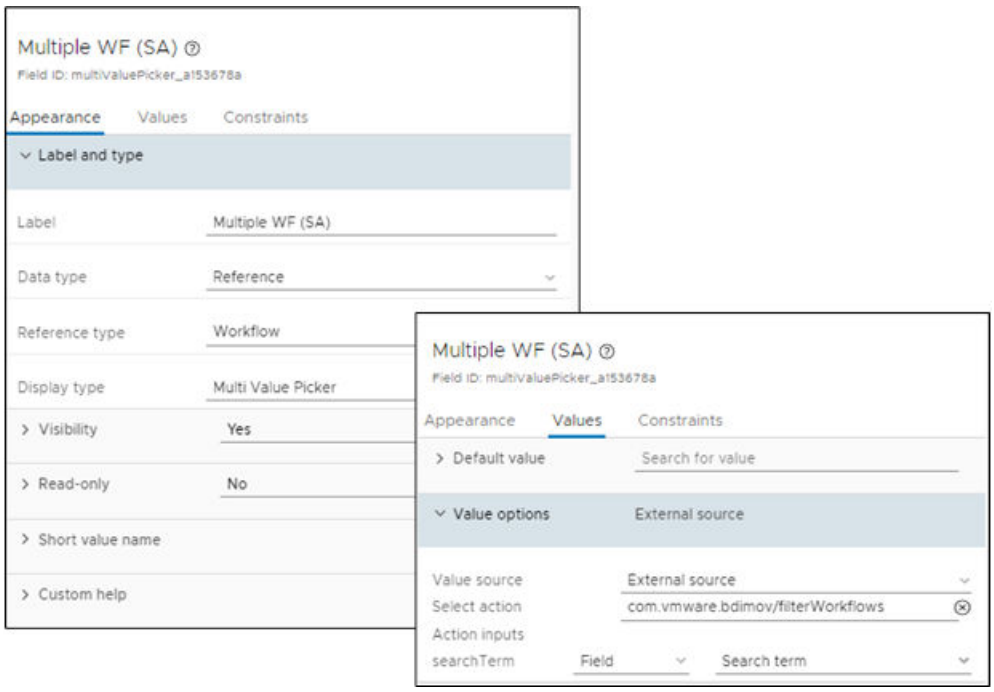
  d Configure the following properties.



Properties Option	Value
Return type	Enter **`Workflow`** and select **Array**. You can use any of the returned types when you run the search. The selected reference type in the custom form must match it. If this procedure, continue to use Workflow.
Inputs	Enter **`searchTerm`**. Notice that the input **`searchTerm`** matches the string used in the script.

e Click **Create**.

2 Configure the multi value picker properties in the custom form designer in Service Broker.



a In Service Broker, select **Content and Policies > Content** and click the vertical dots to the left of the template that you are modifying and click **Customize form**.

b Add or select the multi value picker element in the design canvas.

c In the Properties pane, click **Appearance** and configure the following values.

Property	Value
Data type	Reference
Reference type	Enter **`Workflow`**. Remember, this value is the return type selected for the filterWorkflow action in vRealize Orchestrator, and it must be an array.
Display type	Multi Value Picker

d Click the **Values** tab and configure the following values.

Property	Value
Value options > Value source	External source
Select action	Select the filter action. In this example, select **`filterWorkflows`**.
Action inputs searchTerm	Select **`Field`** and **`Search term`**.

3 Test the filter by requesting the catalog item.

You must ensure that the filter returns the expected values in the multi value picker list, and that the catalog item deploys correctly.

Send email notifications to Service Broker users

As a cloud administrator, you can configure vRealize Automation to send users notifications when specific events in Service Broker and Cloud Assembly occur.

You can send notifications for several types of events, called scenarios, such as the successful completion of a catalog request or a required approval.

Email messages are sent to users in the following scenarios.

Scenario	Description
Deployment Lease Expired	A deployment lease expired and the deployment is about to be deleted. The message is sent to the deployment owner 15–30 minutes before the deployment is destroyed.
Deployment Lease Expiring	A deployment lease expires soon. The message is sent to the deployment owner three days before the lease expires.
Deployment Request Approved	A request is approved. The message is sent to the user who requested the deployment.
Deployment Request Rejected	A request is rejected. The message is sent to the user who requested the deployment.
Deployment Request Waiting for Approval	A request awaits approval. The message is sent to the user who requested the deployment.
Pending Approval Request	A request requires approval. The message is sent to the user who must approve the request.

Prerequisites

- Verify that you configured an outbound email server. See Add an email server in Service Broker to send notifications.

Procedure

1. Log in to vRealize Automation as an administrator.
2. Select **Content and Policies > Notifications > Scenarios**.
3. Select one or more events to trigger user notifications.

Results

Users are subscribed to the notifications that you enabled.

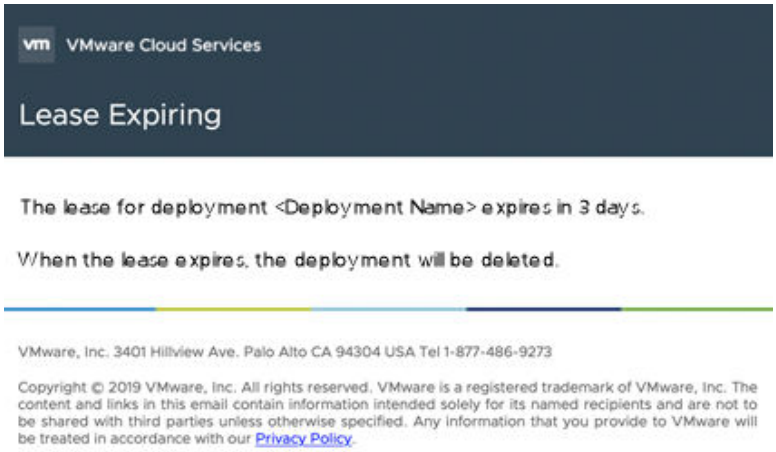
Add an email server in Service Broker to send notifications

As a cloud administrator, you configure an email server if you want to send messages to users about events in Service Broker and Cloud Assembly. The messages are a courtesy that improves the experience of your consumers.

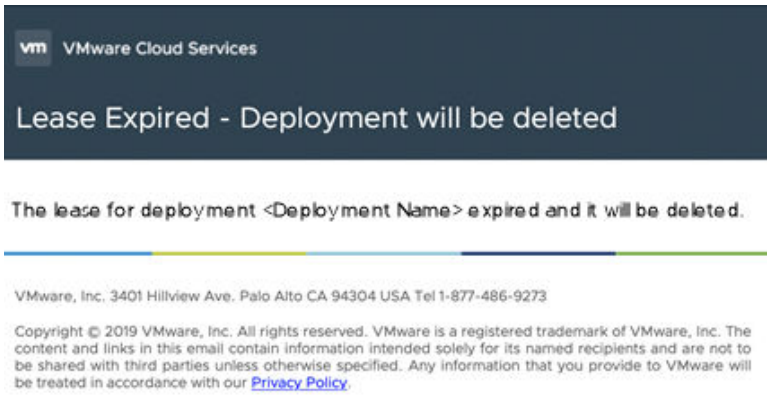
This email server is for outbound messages only.

Email messages are sent to users in the following scenarios.

- A deployment lease expires soon. The message is sent to the deployment owner three days before the lease expires.



- A deployment lease expired and the deployment is about to be deleted. The message is sent to the deployment owner 15–30 minutes before it is destroyed.



Prerequisites

- Verify that you know the credentials required to configure the email server. You must provide the server name and an email account that you want to be the message sender. If your email server requires authentication, you must also provide the user name and password.

Procedure

1. Select **Content and Policies > Notifications > Email Server**.
2. Enter the information for each setting.

   If you need assistance on a particular setting, consult the signpost help.
3. To verify the configured settings, click **Test Connection**.
4. To save, click **Create**.

What to do next

As the administrator, monitor the leases to ensure that the messages are sent to the deployment owners at the correct time.

Working with the Infrastructure options in Service Broker

The Infrastructure tab that is provided in Service Broker is available to administrators. As an administrator who is setting up the service catalog for your users, you use the options to create and manage configuration and connection information that is shared with Cloud Assembly.

For more information about the various connection options, see Setting up Cloud Assembly for your organization.

To better understand projects, and how it associates users with resources, see Adding and managing Cloud Assembly projects.

When working with cloud zones, see Learn more about Cloud Assembly cloud zones

How do I deploy a Service Broker catalog item

4

As a Service Broker consumer, you deploy a catalog item that was imported from Cloud Assembly, Amazon CloudFormation, and other sources so that you can deploy it as part of your work processes.

The catalog items are provided to you by your cloud administrator. The items that are available depend on your project membership. If you are member of one project, you can see only the catalog items for that project. If you are member of several projects, you can see the catalog items those projects.

Projects also determine your options at deployment time.

The information provided in this article is general because each catalog item is unique. The variation depends on how the template and other items were constructed, including what variables are made available to you at request time.

Procedure

1 Click **Catalog**.

The available catalog items are available to you based on your project membership.

2 Locate the catalog item you plan to deploy.

You can use the filter, search, or sorting options to find the catalog item.

3 Click **Request**.

4 Provide any required information.

If the template has more than one released version, select the version that you want to deploy.

A deployment name is required, as is a project. The project list includes those that you are a member of.

The form might have other options that you must configure, depending on how the template was designed.

5 Click **Submit**.

The provisioning process begins and the Deployments tab opens with your current request at the top.

What to do next

Monitor your request. See Monitoring Service Broker deployments.

Learn more about the Service Broker catalog items

Catalog items are imported templates that you can request for deployment. At request time, the information that you must provide or configure depends on how the template was designed by your administrator. When you deploy an item, it is provisioned to based on the cloud regions or datastores that are associated with the selected project.

For a general review of how to deploy, see Chapter 4 How do I deploy a Service Broker catalog item.

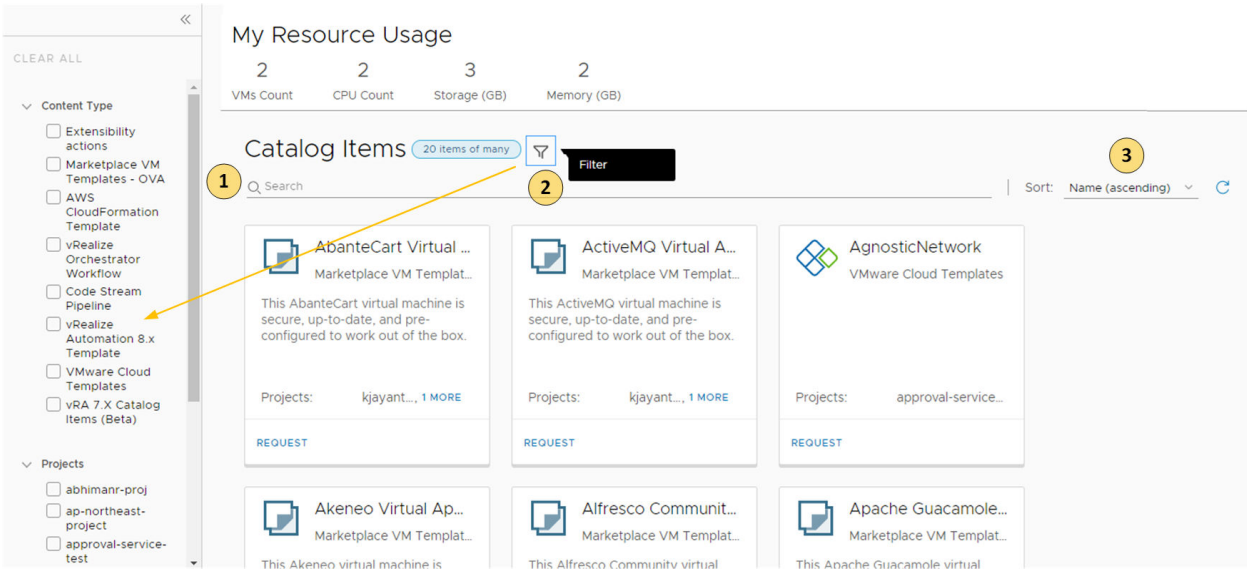
Using the filter and search to locate a catalog item

Depending on your company goals and project members, the catalog available to you can be extensive. You can use the following tools to locate a catalog item.

1 Search. Enter a search term.

2 Filter. Opens the left panel where you can filter by content type and projects.

3 Sort. If the list is still too long, you can sort in ascending or descending order.



My Resource Usage dashboard

The My Resource Usage dashboard provides the current number of VMs, CPUs, storage, and memory that your deployments consume. This information is provided so that you can understand how much you are consuming before you deploy another catalog item. If the numbers seem large, you might consider destroying some of your unused deployments.

The calculated resource usage is for all the deployments where you are the owner, including across projects.

The usage is calculated for resources provisioned by cloud templates for the following resource types:

- VMware vSphere
- VMware Cloud on AWS
- Amazon Web Services
- Microsoft Azure
- Google Cloud Platform

The usage is calculated when any of the following occurs:

- You deploy a catalog item that is provisioned on vSphere, AWS, Azure, or GCP.
- Your administrator onboards deployments where you are the owner. VMs, CPUs, storage, and memory are available for onboarded vSphere deployments. However, CPU and memory are not available for all the endpoints.
- You change a deployment by running a day 2 action. For example, if you add two CPUs to a machine in a deployment, the calculated number of CPUs increases by two.

Service Broker listens for events, such as deployment, onboarding, or day 2 actions, make the calculations, and then updates your resource usage. This usually takes one to two minutes after the change is finished.

The change might include you assigning the deployment to another user. When the change owner action is finished, the resources are subtracted from your resource usage board and added to the new owner's board.

How do I manage my Service Broker Deployments

5

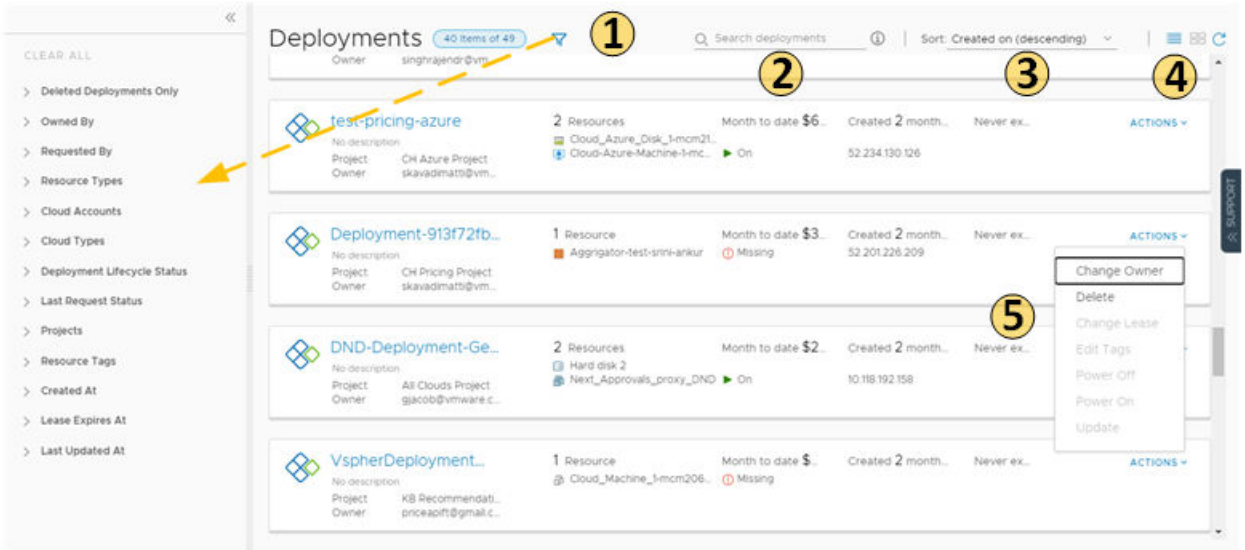
As a Service Broker consumer, you use the Deployment tab to manage your deployments and the associated resources, making changes to deployments, troubleshooting failed deployments, making changes to the resources, and destroying unused deployments.

The deployments are the provisioned instances of catalog items, cloud templates, and onboarded resources. If you manage a small number of deployments, the deployment cards provide a graphical view for managing them. If you manage a large number of deployments, the deployment list and the resource list provide more a more robust management view.

Working with deployment cards and the deployment list

You can locate and manage your deployments using the card list. You can filter or search for specific deployments, and then run actions on those deployments.

Figure 5-1. Deployments page card view



1 Filter your requests based on attributes.

For example, you can filter based on owner, projects, lease expiration date, or other filtering options. Or you might want to find all the deployments for two projects with a particular tag. When you construct the filter for the projects and tag example, the results conform to the following criteria: (Project1 OR Project2) AND Tag1.

The values that you see in the filter pane depend on the current deployments that you have permission to view or manage.

Most of the filters and how to use them are relatively obvious. Additional information about some of these filters is provided below.

2 Search for deployments based on keywords or requester.

3 Sort the list to order by time or name.

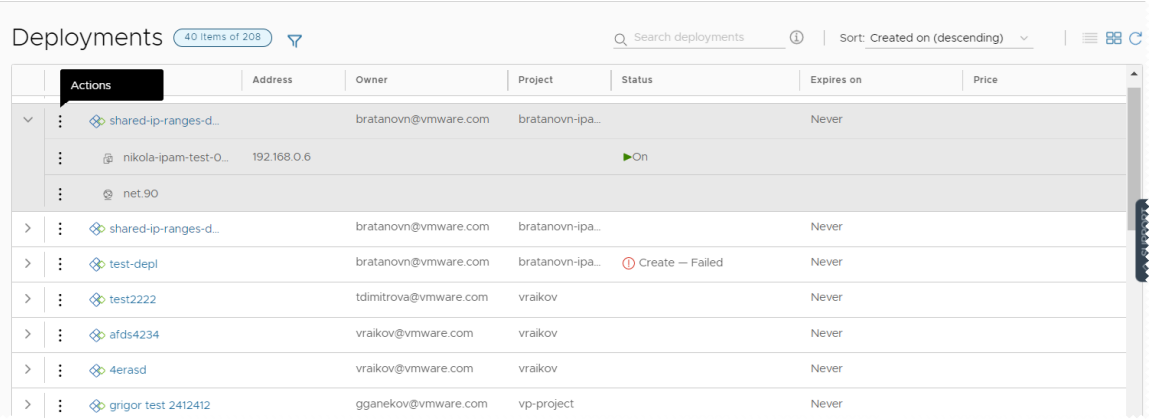
4 Switch between the deployment card and the deployment list views.

5 Run deployment-level actions on the deployment, including deleting unused deployments to reclaim resources.

You can also see deployment costs, expiration dates, and status.

You can switch between the card and list view in the upper right of the page, to the right of the Sort text box. You can use the list view to manage a large number of deployments on fewer pages.

Figure 5-2. Deployment page list view



Working with selected deployment filters

The following table is a not a definitive list of filter options. Most of them are self-evident. However, some of the filters require a little extra knowledge.

Table 5-1. Selected filter information

Filter name	Description
Optimizable Resources Only	If you integrated vRealize Operations Manager and are using the integration to identify reclaimable resources, you can toggle on the filter to limit the list of qualifying deployments.
Deployment Lifecycle Status	The Deployment Lifecycle Status and Last Request Status filters can be used individually or in combination, particularly if you manage a large number of deployments. Examples are included at the end of the Last Request Status section below. Deployment Lifecycle Status filters on the current state of the deployment based on the management operations. This filter is not available for deleted deployments. The values that you see in the filter pane depend on the current state of the listed deployments. You might not see all possible values. The following list includes all the possible values. Day 2 actions are included in the Update status. ■ Create - Successful ■ Create - In Progress ■ Create - Failed ■ Update - Successful ■ Update - In Progress ■ Update - Failed ■ Delete - In Progress ■ Delete - Failed
Last Request Status filters	Last Request Status filters on the last operation or action that ran on the deployment. This filter is not available for deleted deployments. The values that you see in the filter pane depend on the last operations that ran on the listed deployments. You might not see all possible values. The following list is all of the possible values. ■ Pending. The first stage of a request where the action is submitted but the deployment process has not yet started. ■ Failed. The request experienced a failure during any stage of the deployment process. ■ Cancelled. The request was cancelled by a user while the deployment process was processing and not yet completed. ■ Successful. The request successfully created, updated, or deleted a deployment. ■ In Progress. The deployment process is currently running. Additional deployment states, for example,

Table 5-1. Selected filter information (continued)

Filter name	Description
	Initialization and Completion that you see in the deployment History tab are not provided as filters, but you can use the In Progress filter to locate deployments in those states. ■ Approval Pending. The request triggered one or more approval policies. The process is waiting for a response to the approval request. ■ Approval Rejected. The request was denied by the approvers in the triggered approval policies. The request does not continue. The following examples illustrate how the how to use the Deployment Lifecycle Status and Last Request Status filters individually or together. ■ To find all delete requests that failed, select **Delete - Failed** in the Deployment Lifecycle Status filter. ■ To find all the requests waiting for approval, select **Approval Pending** in the Last Request Status filter. ■ To find the delete requests where the approval request is still pending, select **Delete - In Progress** in the Deployment Lifecycle Status filter and **Approval Pending** in the Last Request Status filter.

Working with the resource list

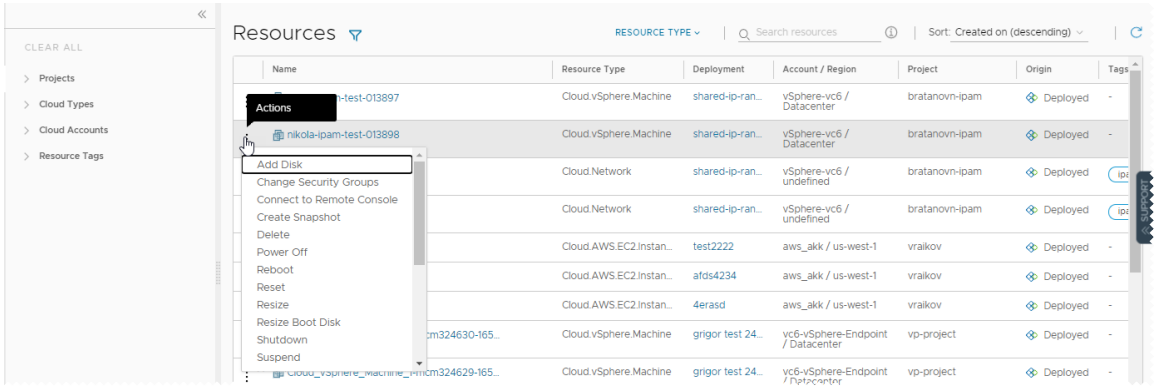
You can use the resource list to manage the machines, storage volumes, and networks that make up your deployments. In the resource list you can manage them in resource type groups rather than by deployments.

Similar to the deployment list view, you can filter the list, select a resource type, search , sort, and run actions.

If you click the resource name, you can work with the resource in the context of the deployment details.

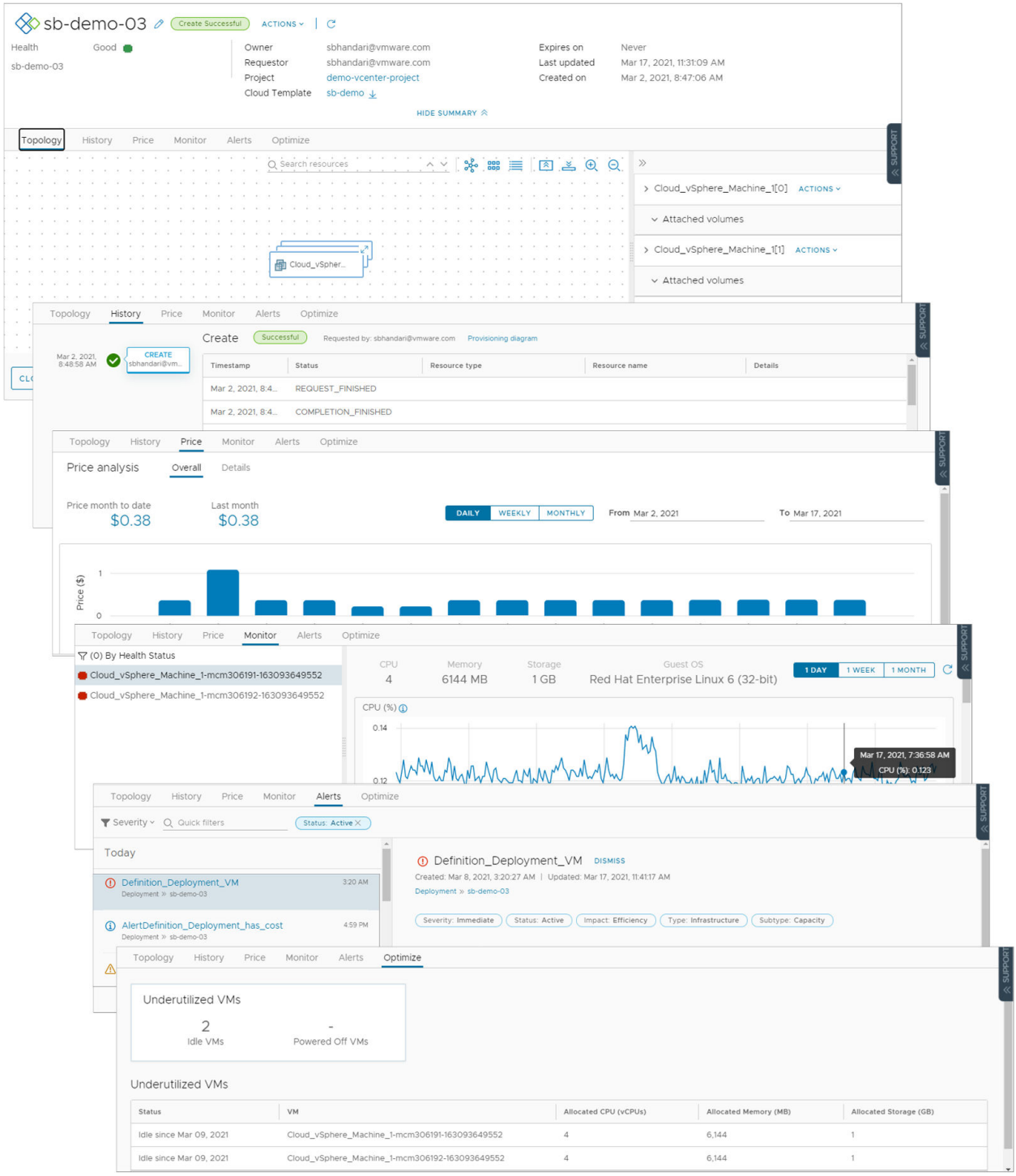
You can locate and manage your deployments using the card list. You can filter or search for specific deployments, and then run actions on those deployments.

Figure 5-3. Deployment resource page list



Working with deployment details

You use the deployment details to understand how the resources are deployed and what changes have been made. You can also see pricing information, the current health of the deployment, and if you have any resources that need to be modified.



- **Topology** tab. You can use the Topology tab to understand the deployment structure and resources.
- **History** tab. The History tab includes all the provisioning events and any events related to actions that you run after requested item is deployed. If there are any problems with the provisioning process, the History tab events will help you with troubleshoot the failures.

- **Pricing** tab. You can use the pricing card to understand how much your deployment is costing your organization. Pricing information is based on vRealize Operations Manager or CloudHealth integrations.
- **Monitor** tab. The Monitor tab data provides information about the health of your deployment based on data from vRealize Operations Manager.
- **Alerts** tab. The Alerts tab provides active alerts on the deployment resources. You can dismiss the alert or add reference notes. The alerts are based on data from vRealize Operations Manager.
- **Optimize** tab. The Optimize tab provides utilization information about your deployment and offers suggestions for reclaiming or otherwise modifying the resources to optimize resource consumption. The optimization information is based on data from vRealize Operations Manager.

This chapter includes the following topics:

Monitoring Service Broker deployments

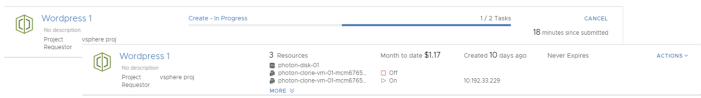
You monitor Service Broker deployment requests to ensure that the resources are provisioned, that the provisioned resources are running, and to resize or destroy the resources as needed.

The Deployment tab provides information about the current state of the deployment and where the resources are deployed in your provider clouds.

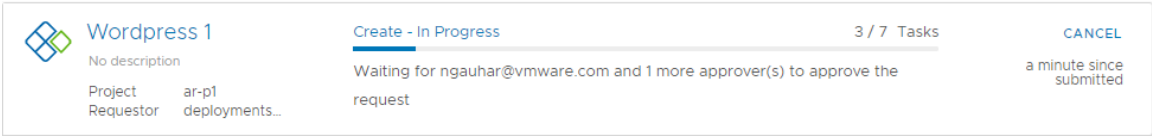
How do I know that my deployment request succeeded

The deployment cards that appear on the Deployments tab show the state of the deployment, including in-progress (top) and completed (below). The card includes the number of deployed resources, how long it has been deployed, and the lease expiration date.

The cards also provide the IP addresses and the actions that you can run on the deployment.

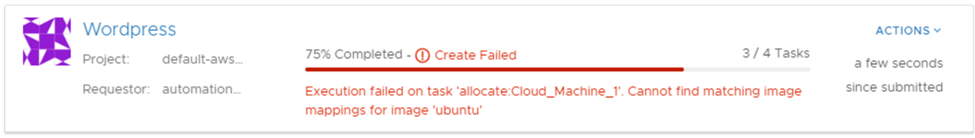


If an approval policy is triggered for your request, you might see the request in an in progress state with the name of at least one approver. How do I configure Service Broker approval policies are defined in Service Broker by your administrator. The approvers are defined in the policy. The approvers approve requests using an Approvals tab. You might also encounter approvals on day 2 actions.



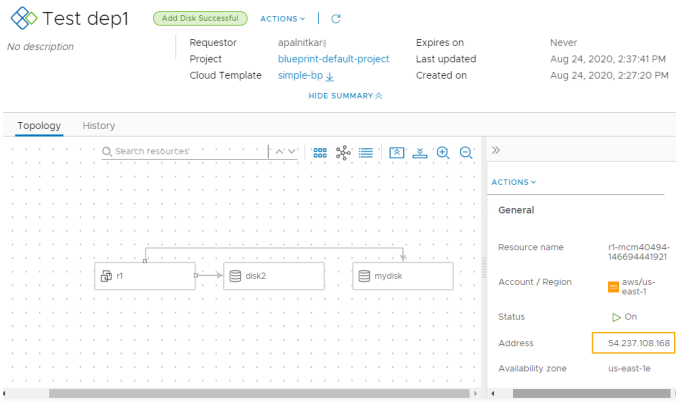
If a deployment fails, the cards show the error message for the point of failure and the process progress. To learn more about the failure, click the deployment name at review the History tab.

For more information about troubleshooting failed deployments, see What can I do if a Service Broker deployment fails.



Where are my resources deployed

To access your successfully provisioned deployments, you might need more than the IP address provided on the card. Click the deployment name and review the deployment details on the Topology tab.



You most likely need the IP address for the primary component. As you click on each component, notice the information that is provided is specific to that component.

The availability of the external link depends on the cloud provider. Where it is available, you must have the credential on that provider to access the component.

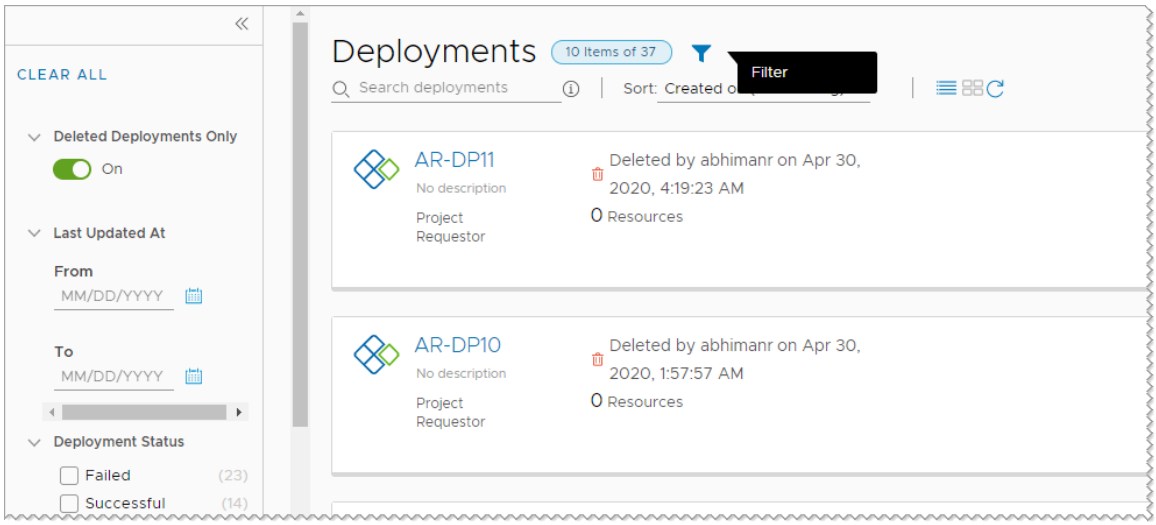
How do I track deleted deployments

After you delete a deployment, you might want to see a list or review the history of a particular deployment.

To view your deleted deployments, click the filter on the **Deployments** tab, and then turn on **Deleted Deployments Only** toggle. The list of deployments is now limited to those that are deleted.

If you need the name of delete machines, you can look at the history to retrieve the information.

The deleted deployments are available for 90 days.



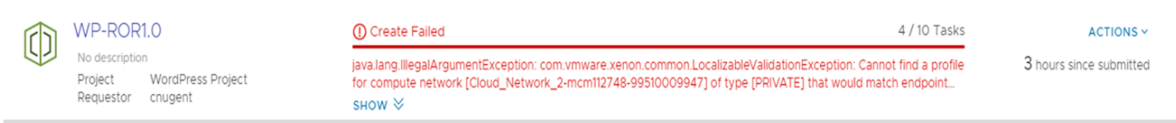
What can I do if a Service Broker deployment fails

Your deployment request might fail for many reasons. It might be due to network traffic, a lack of resources on the target cloud provider, or a flawed deployment specification. Or, the deployment succeeded, but it does not appear to be working. You can use Service Broker to examine your deployment, review any error messages, and determine whether the problem is the environment, the requested workload specification, or something else.

You use this workflow to begin your investigation. The process might reveal that the failure was due to a transient environmental problem. Redeploying the request after verifying the conditions have improved resolves this type of problem. In other cases, your investigation might require you to examine other areas in detail.

Procedure

1 To determine if a request failed, click the **Deployments** tab and locate the deployment card.

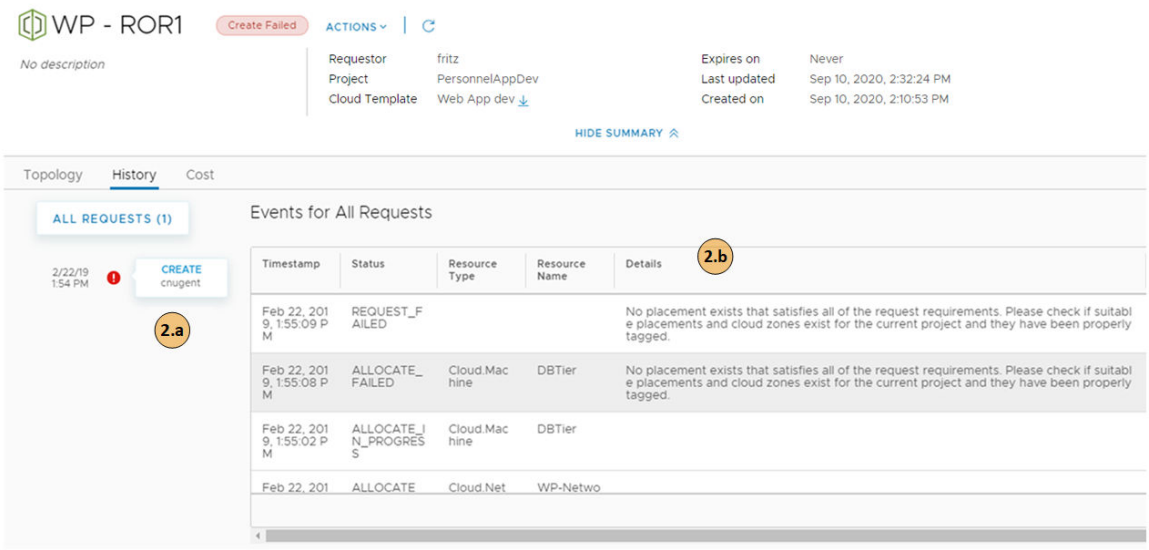


Failed deployments are indicated on the card.

a Review the error message.

b For more information, click the deployment name for the deployment details.

2 On the deployment details page, click the **History** tab.



a Review the event tree to see where the provisioning process failed. This tree is useful when you modify a deployment, but the change fails.

b The **Details** provides a more verbose version of the error message.

What to do next

If you are unable to resolve your problem, contact your cloud administrator for additional assistance.

What actions can I run on Service Broker deployments

After you deploy catalog items, you can run actions in Service Broker to modify and manage the resources. The available actions depend on the resource type and whether the action is supported on a particular cloud account or integrated platform.

The available actions also depend on what your administrator entitled you to run.

As an administrator or project administrator, you can set up Day 2 Actions policies. See How do I entitle deployment users to Service Broker day 2 actions using policies.

You might also see actions that are not included in the list. These are likely custom actions that your administrator configured in Cloud Assembly.

Table 5-2. List of possible actions

Action	Applies to these resource types	Deployed resource type	Description
Add Disk	Machines	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Add additional disks to existing virtual machines. If you add a disk to an Azure machine, the persistent disk or non-persistent disk is deployed in the resource group that includes the machine. When you add a disk to an Azure machines, you can also encrypt the new disk using the Azure disk encryption set configured in the storage profile. When you add a disk to vSphere machines, you can select the SCSI controller, the order of which was set in the cloud template and deployed. You can also specify the unit number for the new disk. You cannot specify a unit number without a selected controller. If you do not select a controller or provide a unit number, the new disk is deployed to first available controller and assigned then next available unit number on that controller. If you add a disk to a vSphere machine for a project with defined storage limits, the added machine is not considered as part of the storage limits. Only resized disks are considered. If you use VMware Storage DRS (SDRS) and the datastore cluster is configured in the storage profile, you can add disks on SDRS to vSphere machines.
Apply Salt Configuration	Machines	■ VMware vSphere	Install a Salt minion or update an existing minion on a virtual machine. The Apply Salt Configuration option is available if you configured the SaltStack Config integration. To apply a configuration, you must select an authentication method. The **Remote access with existing credentials** uses the remote access credentials that are included in the deployment. If you changed the credentials on the machine after deployment, the action can fail. If you know the new credentials, use the Password authentication method. The **Password** and **Private key** use the user name and the password or key to validate your credentials and then connect to the virtual machine using SSH. If you do not provide a value for the Master ID and Minion ID, Salt creates the values for you.

Table 5-2. List of possible actions (continued)

Action	Applies to these resource types	Deployed resource type	Description
Cancel	■ Deployments ■ Various resource types in deployments	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Cancel a deployment or a day 2 action on a deployment or a resource while the request is being processed. You can cancel the request on the deployment card or in the deployment details. After you cancel the request, it appears as a failed request on the **Deployments** tab. Use the **Delete** action to release any deployed resources and clean up your deployment list. Canceling a request that you think has been running too long is one method for managing deployment time. However, it is more efficient to set the **Request Timeout** in the projects. The default timeout is two hours. You can set if for a longer period of time if the workload deployment for a project requires more time.
Change Lease	Deployments	■ Amazon Web Service ■ Microsoft Azure ■ VMware vSphere	Change the lease expiration date and time. When a lease expires, the deployment is destroyed and the resources are reclaimed. Lease policies are set in Service Broker.
Change Owner	Deployments	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Changes to deployment owner to the selected user. The selected user must be a member of the same project that deployed the request. If you want to assign a service administrator or project administrator as the owner, you must add them as a project member. When a cloud template designer deploys a template, the designer is both the requester and the owner. However, a requester can make another project member the owner. You can use policies to control what an owner can do with a deployment, giving them permissions that are more restrictive or less restrictive.

Table 5-2. List of possible actions (continued)

Action	Applies to these resource types	Deployed resource type	Description
Change Project	Deployments	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	The change project action is only available for deployments with onboarded resources. The onboarded deployments can include only machines and disks. The action is not available for deployed cloud templates nor migrated deployments. If you make any changes to the deployment resources, for example, add a disk, you cannot run the change project action. Change the project of an onboarded deployment. This action allows you to change individual deployments from the onboarding project to a different project. Action constraints: ■ The initiating user must have permission to run the change project action. ■ If you are an administrator moving the deployment, you could move the deployment to a project where the owner is not a member and therefore loses access. You can add the user to the target project or move the deployment to a project where they are a member. ■ The target project cloud zones must be the same as the source project cloud zones. If they are not, any future day 2 actions involving cloud account / region resources that you run might not work.
Change Security Groups	Machines	■ VMware vSphere	You can associate and dissociate security groups with machine networks in a deployment. The change action applies to existing and on-demand security groups for NSX-V and NSX-T. This action is available only for single machines, not machine clusters. To associate a security group with the machine network, the security group must be present in the deployment. Dissociating a security group from all networks of all machines in a deployment does not remove the security group from the deployment. These changes do not affect security groups applied as part of the network profiles. This action changes the machine's security group configuration without recreating the machine. This is a non-destructive change. ■ To change the machine's security group configuration, select the machine in the topology pane, then click the **Action** menu in the right pane and select **Change Security Groups**. You can now add or remove the association on the security groups with the machine networks.

Table 5-2. List of possible actions (continued)

Action	Applies to these resource types	Deployed resource type	Description
Connect to Remote Console	Machines	■ VMware vSphere	Open a remote session on the selected machine. Review the following requirements for a successful connection. ■ As a deployment consumer, verify that the provisioned machine is powered on.
Create Disk Snapshot	Machines and disks	■ Microsoft Azure	Create a snapshot of a virtual machine disk or a storage disk. ■ For machines, you create snapshots for individual machine disks, including boot disk, image disks, and storage disks. ■ For storage disks, you create snapshots of independent managed disks, not unmanaged disks. In addition to providing a snapshot name, you can also provide the following information for the snapshot: ■ Incremental Snapshot. Select the check box to create a snapshot of the changes since the last snapshot rather full snapshot. ■ Resource Group. Enter the name of the target resource group where you want to create the snapshot. By default, the snapshot is created in the same resource group that is used by the parent disk. ■ Encryption Set Id. Select the encryption key for the snapshot. By default, the snapshot is encrypted with the same key that is used by the parent disk. ■ Tags. Enter any tags that will help you manage the snapshots in Microsoft Azure.
Create Snapshot	Machines	■ Google Cloud Platform ■ VMware vSphere	Create a snapshot of the virtual machine. If you are allowed only two snapshots in vSphere and you already have them, this command is not available until you delete a snapshot.
Delete	Deployments	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Destroy a deployment. All the resources are deleted and the reclaimed. If a delete fails, you can run the delete action on a deployment a second time. During the second attempt, you can select **Ignore Delete Failures**. If you select this option, the deployment is deleted, but the resources might not be reclaimed. You should check the systems on which the deployment was provisioned to ensure that all resources are removed. If they are not, you must manually delete the residual resources on those systems.
	NSX Gateway	■ NSX	Delete the NAT port forwarding rules from an NSX-T or NSX-V gateway.

Table 5-2. List of possible actions (continued)

Action	Applies to these resource types	Deployed resource type	Description
	Machines and load balancers	■ Amazon Web Service ■ Microsoft Azure ■ VMware vSphere ■ VMware NSX	Delete a machine or load balancer from a deployment. This action might result in an unusable deployment.
	Security groups	■ NSX-T ■ NSX-V	If the security is not associated with any machine in the deployment, the process removes the security group from the deployment. ■ If the security group is on-demand, then it is destroyed on the endpoint. ■ If the security group is shared, the action fails.
Delete Disk Snapshot	Machines and disks	■ Microsoft Azure	Delete an Azure virtual machine disk or managed disk snapshot. This action is available when there is at least one snapshot.
Delete Snapshot	Machines	■ VMware vSphere ■ Google Cloud Platform	Delete a snapshot of the virtual machine.
Disable Boot Diagnostics	Machines	■ Microsoft Azure	Turn off the Azure virtual machine debugging feature. The Disable option is only available if the feature is turned on.
Edit Tags	Deployments	■ Amazon Web Service ■ Microsoft Azure ■ VMware vSphere	Add or modify resource tags that are applied to individual deployment resources.
Enable Boot Diagnostics	Machines	■ Microsoft Azure	Turn on the Azure virtual machine debugging feature to diagnose virtual machine boot failures. The boot diagnostics information is available in your Azure console. The Enable option is only available if the feature is not currently turned on.

Table 5-2. List of possible actions (continued)

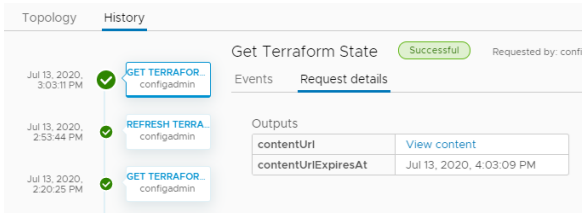
Action	Applies to these resource types	Deployed resource type	Description
Get Terraform State	Terraform Configuration	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Display the Terraform state file. To view any changes that were made to the Terraform machines on the cloud platforms that they were deployed on and update the deployment, you first run the Refresh Terraform State action, and then run this Get Terraform State action. When the file is displayed in a dialog box. The file is available for approximately 1 hour before you need to run a new refresh action. You can copy it if you need it for later. You can also view the file on the deployment History tab. Select the Get Terraform State event on the Events tab, and then click **Request Details**. If the file is not expired, click **View content**. If the file is expired, run the Refresh and Get actions again.  You can run other day 2 action on the Terraform resources that are embedded in the configuration. The available actions depend on the resource type, the cloud platform that they are deployed on, and whether you are entitled to run the actions based on a day 2 policy.
Power Off	Deployments	■ Amazon Web Service ■ Microsoft Azure ■ VMware vSphere	Power off the deployment without shutting down the guest operating systems.
	Machines	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Power off the machine without shutting down the guest operating systems.
Power On	Deployments	■ Amazon Web Service ■ Microsoft Azure ■ VMware vSphere	Power on the deployment. If the resources were suspended, normal operation resumes from the point at which they were suspended.
	Machines	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Power on the machine. If the machine was suspended, normal operation resumes from the point at which the machine was suspended.
Reboot	Machines	■ Amazon Web Service ■ VMware vSphere	Reboot the guest operating system on a virtual machine. For a vSphere machine, VMware Tools must be installed on the machine to use this action.

Table 5-2. List of possible actions (continued)

Action	Applies to these resource types	Deployed resource type	Description
Reconfigure	Load Balancers	■ Amazon Web Service ■ Microsoft Azure ■ VMware NSX	Change the load balancer size and logging level. You can also add or remove routes, and change the protocol, port, health configuration, and member pool settings. For NSX load balancers, you can enable or disable the health check and modify the health options. For NSX-T, you can set the check to active or passive. NSX-V does not support passive health checks.
	NSX Gateway port forwarding	■ NSX-T ■ NSX-V	Add, edit, or delete the NAT port forwarding rules from an NSX-T or NSX-V gateway.
	Security Groups	■ NSX-T ■ NSX-V ■ VMware Cloud ■ VMware vSphere	Add, edit, or remove firewall rules or constraints based on whether the security group is an on-demand or an existing security group. ■ On-demand security group Add, edit, or remove firewall rules for NSX-T and VMware Cloud on-demand security groups. ■ To add or remove a rule, select the security group in the topology pane, click the **Action** menu in the right pane, and select **Reconfigure**. You can now add, edit, or remove the rules. ■ Existing security group Add, edit, or remove constraints for existing NSX-V, NSX-T, and VMware Cloud security groups. ■ To add or remove a constraint, select the security group in the topology pane, click the **Action** menu in the right pane, and select **Reconfigure**. You can now add, edit, or remove the constraints.
Refresh Terraform State	Terraform Configuration	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Retrieve the latest iteration of the Terraform state file. To retrieve any changes that were made to the Terraform machines on the cloud platforms that they were deployed on and update the deployment, you first run this Refresh Terraform State action. To view the file, run the **Get Terraform State** action on the configuration. Use the deployment history tab to monitor the refresh process.
Remove Disk	Machines	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Remove disks from existing virtual machines. If you run the day 2 action on a deployment that is deployed as vSphere machines and disks, the disk count is reclaimed as it applies to project storage limits. The project storage limits do not apply to additional disks that you added after deployment as a day 2 action.

Table 5-2. List of possible actions (continued)

Action	Applies to these resource types	Deployed resource type	Description
Reset	Machines	■ Amazon Web Service ■ Google Cloud Platform ■ VMware vSphere	Force a virtual machine restart without shutting down the guest operating system.
Resize	Machines	■ Amazon Web Service ■ Microsoft Azure ■ Google Cloud Platform ■ VMware vSphere	Increase or decrease the CPU and memory of a virtual machine.
Resize Boot Disk	Machines	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Increase or decrease the size of your boot disk medium. If you run the day 2 action on a deployment that is deployed as vSphere machines and disks, and the action fails with a message similar to “The requested storage is more than the available storage placement,” it is likely due to the defined storage limits on your vSphere VM templates that are defined in the project. The project storage limits do not apply to additional disks that you added after deployment as a day 2 action.
Resize Disk	Storage disk	■ Amazon Web Service ■ Google Cloud Platform	Increase the capacity of a storage disk. If you run the day 2 action on a deployment that is deployed as vSphere machines and disks, and the action fails with a message similar to “The requested storage is more than the available storage placement,” it is likely due to the defined storage limits on your vSphere VM templates that are defined in the project. The project storage limits do not apply to additional disks that you added after deployment as a day 2 action.
	Machines	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	Increase or decrease the size of disks included in the machine image template and any attached disks.
Restart	Machines	■ Microsoft Azure	Shut down and restart a running machine.
Revert to Snapshot	Machines	■ VMware vSphere	Revert to a previous snapshot of the machine. You must have an existing snapshot to use this action.
Run Puppet Task	Managed resources	■ Puppet Enterprise	Run the selected task on machines in your deployment. The tasks are defined in your Puppet instance. You must be able to identify the task and provide the input parameters.
Shutdown	Machines	■ VMware vSphere	Shut down the guest operating system and power off the machine. VMware Tools must be installed on the machine to use this action.
Suspend	Machines	■ Microsoft Azure ■ VMware vSphere	Pause the machine so that it cannot be used and does not consume any system resources other than the storage it is using.

Table 5-2. List of possible actions (continued)

Action	Applies to these resource types	Deployed resource type	Description
Update	Deployments	■ Amazon Web Service ■ Microsoft Azure ■ VMware vSphere	Change the deployment based on the input parameters. For an example, see How to move a deployed machine to another network. If the deployment is based on vSphere resources, and the machine and disks include the count option, storage limits defined in the project might apply when you increase the count. If the action fails with a message similar to "The requested storage is more than the available storage placement," it is likely due to the defined storage limits on your vSphere VM templates that are defined in the project. The project storage limits do not apply to additional disks that you added after deployment as a day 2 action.
Update Tags	Machines and disks	■ Amazon Web Service ■ Microsoft Azure ■ VMware vSphere	Add, modify, or delete a tag that is applied to an individual resource.
Unregister	Machines	■ Amazon Web Service ■ Google Cloud Platform ■ Microsoft Azure ■ VMware vSphere	The unregister action is only available for onboarded deployment machines. Unregistered machines are removed from the deployment, along with any attached disks. By removing the resources, you can then re-run the onboarding workflow for the unregistered machine. You might want to onboard the resource again, this time to a new project. If you make any changes to the machine, for example, add a disk, before unregistering the machine, the unregister action fails.

How to move a deployed machine to another network

While maintaining deployments and networks, you might need the ability to relocate machines that you deployed with Cloud Assembly.

For example, you might deploy to a test network first, then move to a production network. The technique described here lets you design a cloud template in advance to prepare for such day 2 actions. Note that the machine is moved. It isn't deleted and redeployed.

This procedure only applies to **`Cloud.vSphere.Machine`** resources. It won't work for cloud agnostic machines deployed to vSphere.

Prerequisites

- The Cloud Assembly network profile must include all subnets that the machine will connect to. In Cloud Assembly, you can check networks by going to **Infrastructure > Configure > Network Profiles**.

The network profile must be in an account and region that are part of the appropriate Cloud Assembly project for your users.

- Tag the two subnets with different tags. The example that follows assumes that **`test`** and **`prod`** are the tag names.
- The deployed machine must keep the same IP assignment type. It can't change from static to DHCP, or vice versa, while moving to another network.

Procedure

1 In Cloud Assembly, go to **Design**, and create a cloud template for the deployment.

2 In the inputs section of the code, add an entry that lets the user select a network.

```
inputs:
  net-tagging:
    type: string
    enum:
      - test
      - prod
    title: Select a network
```

3 In the resources section of the code, add the **`Cloud.Network`** and connect the vSphere machine to it.

4 Under the **`Cloud.Network`**, create a constraint that references the selection from the inputs.

```
resources:
  ABCServer:
    type: Cloud.vSphere.Machine
    properties:
      name: abc-server
      . . .
      networks:
       - network: '${resource["ABCNet"].id}'
  ABCNet:
    type: Cloud.Network
    properties:
      name: abc-network
      . . .
      constraints:
       - tag: '${input.net-tagging}'
```

5 Continue with your design, and deploy it as you normally would. At deployment, the interface prompts you to select the **`test`** or **`prod`** network.

6 When you need to make a day 2 change, go to **Deployments > Deployments**, and locate the deployment associated with the cloud template.

7 To the right of the deployment, click **Actions > Update**.

8 In the Update panel, the interface prompts you the same way, to select the **`test`** or **`prod`** network.

9 To change networks, make your selection, click **Next**, and click **Submit**.

How do I track my requests that require approval in Service Broker

As a Service Broker or Cloud Assembly user, you received an email notification about a deployment request that you made. You can use this procedure to understand the approval policy workflow related to your request.

This information assumes that you received an email notification about the approval, or that you noticed that your deployment did not progress.

You receive an email with the name of your deployment and the name of the first approver on the list. The message includes a link to the deployment details where you can track the approvals in the deployment details.

If you received an email about the pending request, you can see the name of your deployment and the name of the first approver on the list. The message includes a link to the deployment details where you can track the approvals in the deployment details.

Prerequisites

- To learn more about how approval policies are configured, see How do I configure Service Broker approval policies.

Procedure

1 Click the **Deployments** tab.

2 You requested a deployment or a day 2 action on an existing deployment, but you now see message on your deployment card.

For example, your card displays `Create - Approval Pending` and lists the names of the approvers.

Your request triggered one or more approval policies.

3 For information that helps you track the progress of your request, click the deployment name, and then click the **Details** tab.

When the deployment is first awaiting approval, you only see APPROVAL_IN_PROGRESS. After a few minutes the list of approver names are added in the Details column. If the request requires multiple approvers, the approver list updates as an approver responds. With each update, only the pending approver names remain.

4 When your request is approved or rejected, you receive another email message appropriate to the outcome.

If the request is rejected, the deployment details **History** tab displays REQUEST_FAILED and the details column provides the name of the approver and the reason for rejecting the request.

How do I respond to an approval request in Service Broker

As a designated approver for deployment or day 2 action requests made in Service Broker or Cloud Assembly, you are tasked with approving requests. If you are an assigned approver in the policy, you received an email notification about a deployment request that someone made. If you are user with the Manage Approvals custom role who monitors and responds to approval requests, you do not receive a notification. In either scenario, you can use this procedure to understand how to respond to approval requests.

Some policies might require only your approval, while others require multiple people to approve approvals.

If the policy that you are responding to has multiple approvers but only requires one approver, you might see an already approved request in the Approvals tab. You do not need to take further action.

If you are managing many requests, you can limit the number of approval requests by using the filter option. For example, you might prefer to just see Pending approval requests rather than all the requests.

Prerequisites

- To learn more about how approval policies are configured, see How do I configure Service Broker approval policies.

Procedure

1. If you are an assigned approver, you receive in email that provides the name of the requesting user, the catalog item, and a link to the request in the **Approvals** tab in Service Broker.

   If you are someone who manages approvals, you can open the Approvals tab and continue with the following steps.
2. Locate the approval card for the notification.
3. Review the deployment details and the approval details, and approve or reject the request.

   If you reject the request, you must provide a reason that is included in the email message sent to the requester.
4. The system sends an email to the requester indicating that the request was approved or rejected.