

# Desarrollo de un cliente de servicios web para VMware vRealize Orchestrator

vRealize Orchestrator 7.3

Puede encontrar la documentación técnica más actualizada en el sitio web de VMware:

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

**VMware, Inc.**  
3401 Hillview Ave.  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

**VMware Spain, S.L.**  
Calle Rafael Boti 26  
2.ª planta  
Madrid 28023  
Tel.: +34 914125000  
[www.vmware.com/es](http://www.vmware.com/es)

Copyright © 2008-2017 VMware, Inc. Todos los derechos reservados. [Información sobre el copyright y la marca comercial.](#)

# Contenido

Desarrollo de un cliente de servicios web para VMware vRealize Orchestrator  
5

## 1 Desarrollar un cliente de servicios web 6

## 2 Utilizar la API de REST de vRealize Orchestrator 7

Autenticar respecto a Orchestrator y sistemas de terceros 8

Utilizar la autenticación de vCenter Single Sign-On con la API de REST de Orchestrator 8

Usar autenticación LDAP con la API de REST de Orchestrator 11

Acceder a la documentación de referencia de la API de REST de Orchestrator 11

Utilizar SDK de Java REST 11

Operaciones con flujos de trabajo 12

Buscar un flujo de trabajo y recuperar su definición 12

Ejecutar un flujo de trabajo 15

Ejecutar un flujo de trabajo después de validar sus parámetros de entrada frente a la presentación del flujo de trabajo 17

Interactuar con un flujo de trabajo mientras se ejecuta 21

Recuperar las interacciones de un grupo de trabajo 29

Acceder a un esquema de flujo de trabajo 30

Trabajar con tareas 30

Crear una tarea 30

Modificar una tarea 32

Comprobación del estado de una tarea 33

Búsqueda de objetos en el inventario de Orchestrator 33

Buscar objetos por su tipo y su ID 34

Buscar objetos por relaciones 35

Aplicar filtros 36

Importar y exportar objetos de Orchestrator 37

Importar un flujo de trabajo 37

Exportar un flujo de trabajo 37

Importar una acción 38

Exportar una acción 38

Importar un paquete 39

Exportar un paquete 40

Importar un recurso 41

Exportar un recurso 42

Importar un elemento de configuración 42

Exportar un elemento de configuración 43

Eliminar objetos de Orchestrator	43
Eliminar un flujo de trabajo	43
Eliminar una acción	44
Eliminar un paquete	44
Eliminar un recurso	45
Eliminar un elemento de configuración	45
Configurar permisos en objetos de Orchestrator	46
Permisos de API de REST	46
Recuperar los permisos de un flujo de trabajo	47
Eliminar los permisos de un flujo de trabajo	47
Permisos de un flujo de trabajo	48
Recuperar los permisos de una acción	49
Eliminar los permisos de una acción	49
Permisos de una acción	50
Recuperar los permisos de un paquete	50
Eliminar los permisos de un paquete	51
Permisos de un paquete	51
Recuperar los permisos de un recurso	52
Eliminar los permisos de un recurso	52
Permisos de un recurso	53
Recuperar los permisos de un elemento de configuración	53
Eliminar los permisos de un elemento de configuración	54
Permisos de un elemento de configuración	54
Realizar operaciones con complementos	55
Recuperar información sobre complementos	55
Importar un complemento	56
Exportar un complemento	56
Habilitar o deshabilitar un complemento	57
Operaciones de configuración de servidor	57
Recuperar información sobre la configuración del servidor de Orchestrator	57
Importar la configuración del servidor de Orchestrator	58
Exportar la configuración del servidor de Orchestrator	58
Operaciones de etiquetado	59
Etiquetar un objeto	59
Eliminar etiquetas de un objeto	60
Enumerar etiquetas de objeto	60
Enumerar objetos etiquetados por tipo	61
Enumerar propietarios de etiquetas	61
Enumerar etiquetas por usuarios	61
Enumerar etiquetas por usuarios filtradas por nombre de etiqueta	62
Quitar etiquetas por usuarios	62

# Desarrollo de un cliente de servicios web para VMware vRealize Orchestrator

*Desarrollo de un cliente de servicios web para VMware vRealize Orchestrator* ofrece información sobre el desarrollo de un cliente de servicios web para VMware® vRealize Orchestrator.

Orchestrator ofrece una API de servicios web para desarrollar aplicaciones que permitan acceder a los flujos de trabajo a través de la web y utilizarlos. Orchestrator ofrece una API de REST (transferencia de estado de representación) que puede utilizar para llevar a cabo varias operaciones en los flujos de trabajo.

## Público objetivo

Esta información está destinada a desarrolladores de aplicaciones web que desean acceder a los procesos de Orchestrator a través de una red, mediante el servicio web RESTful.

# Desarrollar un cliente de servicios web

# 1

VMware vRealize Orchestrator ofrece varias API de servicios web para desarrollar aplicaciones que permitan acceder a los flujos de trabajo a través de la web. La finalidad principal de las API de servicios web de Orchestrator es permitir integrar flujos de trabajo de Orchestrator en aplicaciones basadas en Internet personalizadas.

Orchestrator proporciona una API de servicios web que se basa en una API de REST (Representational state transfer, Transferencia de estado de representación). La API de REST de Orchestrator expone los objetos de del inventario de Orchestrator y los inventarios de los complementos instalados como recursos a los que se puede tener acceso en las URL predefinidas. Las solicitudes HTTP en estas URL hacen que se activen operaciones sobre flujos de trabajo. La API de REST de Orchestrator expone objetos de inventario como recursos mediante una serie de servicios web RESTful que se pueden utilizar para recuperar las definiciones de flujos de trabajo, ejecutar flujos de trabajo, comprobar el estado de los flujos de trabajo en ejecución, cancelar ejecuciones de flujos de trabajo, procesar interacciones del usuario en espera, recuperar la presentación de flujos de trabajo, etcétera.

# Utilizar la API de REST de vRealize Orchestrator

## 2

La API de REST de Orchestrator proporciona funcionalidades que permiten comunicarse con el servidor de Orchestrator directamente a través de HTTP y realizar varias operaciones relativas al flujo de trabajo a través de los flujos de trabajo.

La API de REST de Orchestrator expone los objetos de los inventarios del servidor de Orchestrator y los complementos instalados como recursos en las URL predefinidas. Las llamadas HTTP se realizan en estas URL para activar operaciones en Orchestrator. De este modo, es posible llevar a cabo varias tareas a través de los flujos de trabajo:

- Ejecute o planifique un flujo de trabajo, recupere las ejecuciones de un flujo de trabajo, responda a una interacción de usuario y cancele una ejecución de flujo de trabajo.
- Recupere detalles sobre un flujo de trabajo, como sus parámetros de entrada o de salida o su presentación.
- Recupere detalles sobre una ejecución de flujo de trabajo, como su estado, los registros generados, la fecha de inicio y la fecha de finalización.
- Examine los inventarios de Orchestrator y los complementos instalados.
- Importe y exporte flujos de trabajo, acciones y paquetes.

Mediante el uso de la API de REST de Orchestrator, puede integrar fácilmente los flujos de trabajo de Orchestrator en aplicaciones personalizadas que puede incorporar en cualquier lenguaje de programación.

La API de REST de Orchestrator también admite eTag y un mecanismo para almacenar en caché los datos de respuesta.

Este capítulo incluye los siguientes temas:

- [Autenticar respecto a Orchestrator y sistemas de terceros](#)
- [Acceder a la documentación de referencia de la API de REST de Orchestrator](#)
- [Utilizar SDK de Java REST](#)
- [Operaciones con flujos de trabajo](#)
- [Trabajar con tareas](#)
- [Búsqueda de objetos en el inventario de Orchestrator](#)

- [Importar y exportar objetos de Orchestrator](#)
- [Eliminar objetos de Orchestrator](#)
- [Configurar permisos en objetos de Orchestrator](#)
- [Realizar operaciones con complementos](#)
- [Operaciones de configuración de servidor](#)
- [Operaciones de etiquetado](#)

## Autenticar respecto a Orchestrator y sistemas de terceros

Debe autenticar respecto a Orchestrator en las solicitudes HTTP que realiza a través de la API de REST de Orchestrator. Si utiliza la API de REST de Orchestrator para acceder a recursos de un sistema de terceros, como vCenter Server, también debe realizar la autenticación respecto a dicho sistema.

Por ejemplo, para acceder a todos los flujos de trabajo del inventario de Orchestrator, debe autenticarse respecto a Orchestrator. Ahora bien, para ejecutar un flujo de trabajo respecto a vCenter Server, debe realizar la autenticación respecto a Orchestrator y vCenter Server.

En función de si configura Orchestrator con LDAP o vCenter Single Sign-On, el esquema de autenticación para la API de REST de Orchestrator varía. Si Orchestrator utiliza LDAP, debe realizar la autenticación mediante credenciales válidas. Si Orchestrator utiliza vCenter Single Sign-On, en función de la configuración, puede autenticar mediante la autenticación básica o un token poseedor de la clave emitido por el servidor vCenter Single Sign-On.

Si realiza solicitudes HTTP en la URL de nivel superior de la API de REST de Orchestrator, no necesita autenticar respecto a Orchestrator. La URL de nivel superior de la API de REST de Orchestrator es `https://host_orchestrator:puerto/vco/api/`.

---

**Nota** El número de puerto predeterminado es 8281.

---

Una solicitud GET en la URL de nivel superior de la API de REST devuelve direcciones URL a todos los recursos a los que se puede acceder con la API. Para realizar solicitudes HTTP en esas direcciones URL, debe autenticar respecto a Orchestrator.

## Utilizar la autenticación de vCenter Single Sign-On con la API de REST de Orchestrator

Si Orchestrator se configura con el servidor de vCenter Single Sign-On y se desactiva la autenticación básica, necesita un token poseedor de la clave principal para acceder a los objetos del sistema en Orchestrator a través de la API de REST de Orchestrator. Para acceder a vCenter Server o a sistemas de terceros que utilicen el servidor de vCenter Single Sign-On a través del servidor de Orchestrator, necesita un token poseedor de la clave delegado para Orchestrator y su token principal.



Si Orchestrator se configura con el servidor de vCenter Single Sign-On y la autenticación básica está activada, debe autenticar mediante credenciales válidas y Orchestrator administrará el token poseedor de la clave.

## Acceder a objetos del sistema en Orchestrator

Puede acceder a los objetos del sistema en Orchestrator en las direcciones URL de los servicios de catálogo e inventario de la API de REST.

- `https://host_orchestrator:puerto/vco/api/inventory/System/`
- `https://host_orchestrator:puerto/vco/api/catalog/System/`

Cuando accede a objetos del sistema en Orchestrator, transfiere su token poseedor de la clave principal en el encabezado de Authorization de las solicitudes HTTP que realiza al servicio de catálogo o inventario.

Por ejemplo, para recuperar todos los objetos del sistema de tipo Workflow, realiza una solicitud GET en `https://host_orchestrator:puerto/vco/api/catalog/System/Workflow/`. Para autenticarse en Orchestrator, debe transferir su token poseedor de la clave principal en el encabezado Authorization de la solicitud.

## Acceder a objetos en sistemas de terceros

Para realizar operaciones en sistemas de otros proveedores registrados con el servidor de vCenter Single Sign-On a través de la API de REST de Orchestrator, debe autenticarse en Orchestrator y en el sistema de terceros. Es necesario incluir dos encabezados en las llamadas HTTP que realice a través de la API de REST de Orchestrator.

- **Authorization.** Debe transferir el token poseedor de la clave principal en este encabezado.
- **VC0Authorization.** Debe transferir un token poseedor de la clave delegado para Orchestrator en este encabezado. Es necesario obtener el token delegado del servidor de vCenter Single Sign-On Server para Orchestrator. Orchestrator utiliza el token delegado para autenticarse en el sistema de terceros en su nombre.

Por ejemplo, para ejecutar un flujo de trabajo que utiliza una máquina virtual a través de la API de REST de Orchestrator, acceda a los recursos de Orchestrator y de vCenter Server. Para autenticarse en Orchestrator y en vCenter Server, debe transferir su token poseedor de la clave principal en el encabezado de Authorization de la solicitud realizada y el token delegado en el encabezado de VC0Authorization. De este modo, se autentica en Orchestrator con su token principal y Orchestrator se autentica en su nombre en vCenter Server con el token delegado.

El servidor de vCenter Single Sign-On trata a Orchestrator como una solución y cada solución queda registrada con un nombre de usuario exclusivo en el servidor de vCenter Single Sign-On. Para solicitar un token delegado para Orchestrator, transfiera el nombre de usuario de la solución de Orchestrator y un token poseedor de la clave principal al servidor de vCenter Single Sign-On. El token que emite el servidor de vCenter Single Sign-On Server es un token poseedor de la clave delegado para que Orchestrator se autentique en su nombre en los sistemas de terceros.

## Ejemplo: Obtener una sesión en modo vCenter Single Sign-On

Con el código de ejemplo siguiente se obtiene una sesión en modo vCenter Single Sign-On.

```
URI uri = URI.create("https://orchestrator-server:8281/vco/api");
VcoSessionFactory sessionFactory = new DefaultVcoSessionFactory(uri);

//provide the address of the vCenter Single Sign-On server
URI ssoUri = URI.create("https://sso-server:7444/ims/STSService?wsdl");

//set the tokens to be valid for an hour
long lifeTimeSeconds = 60 * 60;

//create a factory for vCenter Single Sign-On tokens
SsoAuthenticator sso = new SsoAuthenticator(ssoUri, sessionFactory, lifeTimeSeconds);

//provide vCenter Single Sign-On credentials
SsoAuthentication authentication = sso.createSsoAuthentication("username", "password");

VcoSession session = sessionFactory.newSession(authentication);
//use session here
```

## Obtener el nombre de usuario de la solución Orchestrator

El servidor vCenter Single Sign-On trata a Orchestrator como una solución y cada solución queda registrada con un nombre de usuario exclusivo en el servidor vCenter Single Sign-On. Para poder solicitar al servidor vCenter Single Sign-On un token poseedor de la clave delegado para Orchestrator, necesita el nombre de usuario de la solución de Orchestrator.

### Requisitos previos

Compruebe que dispone de un token poseedor de la clave principal válido que haya emitido el servidor vCenter Single Sign-On.

### Procedimiento

- 1 Haga una solicitud GET en la dirección URL del nombre de usuario de la solución Orchestrator:

```
GET https://{host_orchestrator}:{puerto}/vco/api/users/
```

- 2 Indique su token poseedor de la clave principal en el encabezado Authorization de la solicitud.

### Resultados

El elemento `<user solution-user="OrchestratorSolutionUserName"/>` de la respuesta contiene el nombre de usuario de la solución Orchestrator. Este es un ejemplo de un nombre de usuario de la solución Orchestrator.

```
<user xmlns="http://www.vmware.com/vco" solution-user="Orchestrator-133acc26ff78e5695b102146326"
admin-rights="true"/>
```

## Pasos siguientes

Utilice el nombre de usuario de la solución Orchestrator y su token poseedor de la clave principal para solicitar al servidor vCenter Single Sign-On un token poseedor de la clave delegado.

## Usar autenticación LDAP con la API de REST de Orchestrator

Se debe aplicar el esquema de autenticación HTTP básica si Orchestrator se configura con LDAP, o si se utiliza el servidor de Orchestrator para acceder a un sistema de terceros configurado con LDAP.

El esquema de autenticación HTTP básica permite realizar la autenticación frente a Orchestrator o un sistema de terceros mediante la inclusión de un `Authorization` encabezado en las llamadas de API que realice. En el encabezado de `Authorization` debe proporcionar credenciales con codificación base64. Orchestrator utiliza las mismas credenciales para autenticar en su nombre frente a sistemas de terceros configurados con LDAP.

Para ver información detallada sobre la autenticación HTTP básica, consulte RFC 2617.

### Ejemplo: Obtener una sesión en modo LDAP

Con el siguiente código de ejemplo se obtiene una sesión en modo LDAP.

```
URI uri = URI.create("https://orchestrator-server:8283/vco/api");
VcoSessionFactory sessionFactory = new DefaultVcoSessionFactory(uri);

//provide LDAP credentials
Authentication auth = new UsernamePasswordAuthentication("username", "password");

VcoSession session = sessionFactory.newSession(auth);
//use session here
```

## Acceder a la documentación de referencia de la API de REST de Orchestrator

La documentación de referencia de la API de REST de Orchestrator contiene información sobre los servicios web RESTful de la API, el modelo de datos aplicable a la API, los códigos de respuesta válidos para la API, ejemplos de códigos, etcétera.

La documentación de referencia de la API de REST de Orchestrator se instala junto con Orchestrator. La documentación de referencia se encuentra en `https://host_orchestrator:puerto/vco/api/docs/`.

## Utilizar SDK de Java REST

Puede utilizar una biblioteca de SDK de Java para llamar a operaciones de la API de REST de Orchestrator en aplicaciones Java y trabajar directamente con objetos.

Cada servicio web RESTful del SDK de Orchestrator REST tiene una clase de ajuste de Java con métodos que corresponden a las operaciones que se pueden ejecutar mediante el servicio.

El SDK de Java REST se instala junto con Orchestrator. Los artefactos del SDK de Java REST están disponibles en las ubicaciones siguientes.

---

**Nota** Solo es posible acceder a los artefactos si se ha implementado Orchestrator Appliance.

---

- [https://host\\_orchestrator:puerto/vco-repo/com/vmware/o11n/o11n-rest-client/](https://host_orchestrator:puerto/vco-repo/com/vmware/o11n/o11n-rest-client/)
- [https://host\\_orchestrator:puerto/vco-repo/com/vmware/o11n/o11n-rest-client-examples/](https://host_orchestrator:puerto/vco-repo/com/vmware/o11n/o11n-rest-client-examples/)
- [https://host\\_orchestrator:puerto/vco-repo/com/vmware/o11n/o11n-rest-client-services/](https://host_orchestrator:puerto/vco-repo/com/vmware/o11n/o11n-rest-client-services/)
- [https://host\\_orchestrator:puerto/vco-repo/com/vmware/o11n/o11n-rest-client-stubs/](https://host_orchestrator:puerto/vco-repo/com/vmware/o11n/o11n-rest-client-stubs/)

## Ejemplo: Ejecutar un flujo de trabajo y esperar a su finalización

El código de ejemplo siguiente ejecuta un flujo de trabajo y espera a que finalice.

```
//start a new session to Orchestrator by using specified credentials
VcoSession session = DefaultVcoSessionFactory.newLdapSession(new URI("https://orchestrator-
server:8281/vco/api/"), "username", "password");

//create the services
WorkflowService workflowService = new WorkflowService(session);
ExecutionService executionService = new ExecutionService(session);

//find a workflow by ID
Workflow workflow = workflowService.getWorkflow("1231235");

//create an ExecutionContext from the user's input
ExecutionContext context = new ExecutionContextBuilder().addParam("name", "Jerry").addParam("age",
18).build();

//run the workflow
WorkflowExecution execution = executionService.execute(workflow, context);

//wait for the workflow to reach the user interaction state, checking every 500 milliseconds
execution = executionService.awaitState(execution, 500, 10, WorkflowExecutionState.CANCELED,
WorkflowExecutionState.FAILED, WorkflowExecutionState.COMPLETED);

String nameParamValue = new ParameterExtractor().fromTheOutputOf(execution).extractString("name");
System.out.println("workflow was executed with 'name' input set to" + nameParamValue);
```

## Operaciones con flujos de trabajo

La API de REST de Orchestrator proporciona servicios web que se pueden utilizar para realizar varias operaciones con flujos de trabajo.

### Buscar un flujo de trabajo y recuperar su definición

Para poder realizar cualquier tipo de operación con un flujo de trabajo, debe buscar ese flujo de trabajo en el inventario de Orchestrator y recuperar su definición. La definición enumera los

parámetros de entrada y salida del flujo de trabajo, y contiene vínculos a las ejecuciones de flujo de trabajo disponibles, a la presentación del flujo de trabajo y a otros objetos.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

### Procedimiento

#### 1 Busque el elemento de inventario del flujo de trabajo.

- Si dispone del nombre completo del flujo de trabajo o de una palabra clave del nombre, haga una solicitud GET en la URL del servicio de flujo de trabajo aplicando el filtro:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows?
conditions=name={nombre_completo_flujo_trabajo}

GET https://{host_orchestrator}:{puerto}/vco/api/workflows?conditions=name~{palabra_clave}
```

- Busque el flujo de trabajo en el servicio de catálogo o de inventario realizando una solicitud GET en la URL que sea un punto de entrada para los elementos del inventario del flujo de trabajo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/catalog/System/Workflow/

GET https://{host_orchestrator}:{puerto}/vco/api/inventory/System/Workflows/
```

#### 2 Recupere el elemento del inventario del flujo de trabajo realizando una solicitud GET en su URL:

```
GET https://{host_orchestrator}:{puerto}/vco/api/catalog/System/Workflow/
{ID_flujo_trabajo}/
```

#### 3 Recupere la definición del flujo de trabajo realizando una solicitud GET en la URL de la definición:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
```

## Ejemplo: Buscar el flujo de trabajo Hola a todos

Puede buscar el flujo de trabajo Hola a todos y recuperar su definición:

- 1 Para buscar el flujo de trabajo Hola a todos, haga una solicitud GET en la URL del servicio de flujo de trabajo aplicando un filtro:

```
GET https://localhost:8281/vco/api/workflows?conditions=name~Hello
```

Recibirá una lista de los flujos de trabajo que contienen "Hello" en sus nombres:

```
<xml version="1.0" encoding="UTF-8" standalone="yes">
<inventory-items xmlns="http://www.vmware.com/vco" total="2">
  <link rel="down"
    href="https://localhost:8281/vco/api/catalog/System/Workflow/
CF808080808080808080808080808080E6808080013086668236014a0614d16e1/">
    <attributes>
      <attribute name="id"
value="CF808080808080808080808080808080E6808080013086668236014a0614d16e1"/>
      <attribute name="canExecute" value="true" />
      <attribute name="description" value="" />
      <attribute name="name" value="Interactive Hello World" />
      <attribute name="type" value="Workflow"/>
      <attribute name="canEdit" value="true"/>
    </attributes>
  </link>
  <link rel="down"
    href="https://localhost:8281/vco/api/catalog/System/Workflow/
CF808080808080808080808080808080DA808080013086668236014a0614d16e1/">
    <attributes>
      <attribute name="id"
value="CF808080808080808080808080808080DA808080013086668236014a0614d16e1"/>
      <attribute name="canExecute" value="true" />
      <attribute name="description" value="" />
      <attribute name="name" value="Send Hello" />
      <attribute name="type" value="Workflow"/>
      <attribute name="canEdit" value="true"/>
    </attributes>
  </link>
</inventory-items>
```

- 2 Haga una solicitud GET en la dirección URL del elemento del inventario del flujo de trabajo  
Hola a todos:

```
GET https://localhost:8281/vco/api/catalog/System/Workflow/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/
```

Recibirá el elemento del inventario del flujo de trabajo Hola a todos en el cuerpo de la respuesta:

```
<xml version="1.0" encoding="UTF-8" standalone="yes">
<inventory-item xmlns="http://www.vmware.com/vco"
    href="https://localhost:8281/vco/api/catalog/System/Workflow/
CF8080808080808080808080DA808080013086668236014a0614d16e1/">
    <relations>
        <link rel="down"
            href="https://localhost:8281/vco/api/workflows/
CF8080808080808080808080DA808080013086668236014a0614d16e1/" />
    </relations>
    <attributes>
        <attribute name="id"
value="CF8080808080808080808080808080808080808080DA808080013086668236014a0614d16e1"/>
        <attribute name="canExecute" value="true" />
```

```
<attribute name="description" value="" />
<attribute name="name" value="Send Hello" />
<attribute name="type" value="Workflow"/>
<attribute name="canEdit" value="true"/>
</attributes>
</inventory-item>
```

- 3 Para recuperar la definición del flujo de trabajo, haga una solicitud GET en su URL:

```
GET https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/
```

Recibirá la definición del flujo de trabajo Hola a todos en el cuerpo de la respuesta:

```
<xml version="1.0" encoding="UTF-8" standalone="yes">
<workflow xmlns="http://www.vmware.com/vco" customized-icon="false"
  href="https://localhost:8281/vco/api/workflows/
CF8080808080808080808080DA808080013086668236014a0614d16e1/">
  <relations>
    <link rel="up"
      href="https://localhost:8281/vco/api/inventory/System/Workflows/Samples/HelloWorld/" />
    <link rel="add"
      href="https://localhost:8281/vco/api/workflows/
CF8080808080808080808080DA808080013086668236014a0614d16e1/executions/" />
    <link rel="down"
      href="https://localhost:8281/vco/api/workflows/
CF8080808080808080808080DA808080013086668236014a0614d16e1/executions/" />
    <link rel="down"
      href="https://localhost:8281/vco/api/workflows/
CF8080808080808080808080DA808080013086668236014a0614d16e1/presentation/" />
    <link rel="down"
      href="https://localhost:8281/vco/api/workflows/
CF8080808080808080808080DA808080013086668236014a0614d16e1/tasks/" />
    <link rel="down"
      href="https://localhost:8281/vco/api/workflows/
CF8080808080808080808080DA808080013086668236014a0614d16e1/icon/" />
  </relations>
  <input-parameters>
    <parameter name="name" type="string" />
  </input-parameters>
  <output-parameters>
    <parameter name="message" type="string" />
  </output-parameters>
  <name>Send Hello</name>
  <description></description>
</workflow>
```

## Ejecutar un flujo de trabajo

Para ejecutar un flujo de trabajo a través de la API de REST de Orchestrator, cree un objeto de ejecución para un determinado flujo de trabajo.





```
<link rel="up"  
    href="https://localhost:8281/vco/api/inventory/System/Workflows/Samples/HelloWorld/" />  
<link rel="add"  
    href="https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/executions/" />  
<link rel="down"  
    href="https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/executions/" />  
<link rel="down"  
    href="https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/presentation/" />  
<link rel="down"  
    href="https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/tasks/" />  
<link rel="down"  
    href="https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/icon/" />  
</relations>  
<input-parameters>  
    <parameter name="name" type="string" />  
</input-parameters>  
<output-parameters>  
    <parameter name="message" type="string" />  
</output-parameters>  
<name>Send Hello</name>  
    <description></description>  
</workflow>
```

- 2 Realice una solicitud POST en la URL que contiene los objetos de ejecución para el flujo de trabajo:

[illegible]

Transfiera los valores para los parámetros de entrada en un elemento `execution-context` del cuerpo de la solicitud:

```
<execution-context xmlns="http://www.vmware.com/vco">
  <parameters>
    <parameter name="name" type="string">
      <string>John Smith</string>
    </parameter>
  </parameters>
</execution-context>
```

Ejecute un flujo de trabajo después de validar sus parámetros de entrada frente a la presentación del flujo de trabajo

La presentación de un flujo de trabajo puede definir restricciones para los valores que puede transferir a los parámetros de entrada del flujo de trabajo, como una lista predefinida de los valores o un determinado rango de valores. Para asegurarse de que el flujo de trabajo se ejecute

correctamente, debe validar los valores que transfiere a los parámetros de entrada del flujo de trabajo respecto a la definición de la presentación del flujo de trabajo.

Cuando integra flujos de trabajo en aplicaciones personalizadas, quizá deba crear un asistente en el que introducir los valores para los parámetros de entrada del flujo de trabajo cuando lo ejecute. Con el uso del servicio Presentación del flujo de trabajo, puede crear una instancia de la presentación de un flujo de trabajo y transferir valores para sus parámetros de entrada en partes que correspondan a las diferentes pantallas del asistente. Puede validar los valores que transfiere a los parámetros de entrada respecto a las restricciones definidas en la presentación del flujo de trabajo.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

### Procedimiento

- 1 Recupere la definición del flujo de trabajo que desee ejecutar realizando una solicitud GET en la URL que contiene la definición del flujo de trabajo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
```

Recibirá la definición del flujo de trabajo en el cuerpo de la respuesta de la solicitud. En la definición del flujo de trabajo, puede ver los parámetros de entrada del flujo de trabajo, la descripción del flujo de trabajo y otra información.

- 2 Recupere la definición de la presentación del flujo de trabajo realizando una solicitud GET en su URL:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/  
presentation/
```

- 3 En el cuerpo de respuesta de la solicitud, examine la definición de la presentación del flujo de trabajo para ver si hay restricciones en los valores que se pueden transferir a los parámetros de entrada.

Por ejemplo, un parámetro de entrada puede tener una lista predefinida de valores en la que elegir.

- 4 Cree una instancia de la presentación del flujo de trabajo realizando una solicitud POST en la URL de las instancias de la presentación:

```
POST https://{orchestrator_host}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/  
presentation/instances/
```

- Proporcione un elemento `execution-context` en el cuerpo de solicitud para crear una instancia de la presentación.

Puede transferir una instancia de `execution-context` vacía o `execution-context` con valores solo para algunos de los parámetros de entrada.

- 6** Para transferir valores a los parámetros de entrada por partes, realice tantas solicitudes POST o PUT como sea necesario en la URL que contiene la instancia de la presentación:

```
PUT https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
presentation/instances/{ID_ejecución}/
```

- 7** Revise el cuerpo de respuesta de la solicitud POST o PUT que ha realizado.

Si los valores que ha transferido a los parámetros de entrada son válidos, encontrará un atributo `valid="true"` en la etiqueta `execution`. Si la presentación es válida, puede tomar los valores que se enumeran en el elemento `out-parameters` de la respuesta y transferirlos como valores a los parámetros de entrada cuando se ejecute el flujo de trabajo.

- Si los valores de los parámetros de entrada son válidos, ejecute el flujo de trabajo realizando una solicitud POST en la URL que contiene las ejecuciones del flujo de trabajo:

```
POST https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
executions/
```

- 9 Proporcione valores válidos para los parámetros de entrada del flujo de trabajo en un elemento `execution-context`.

Ejemplo: Ejecute el flujo de trabajo Hola a todos validando sus parámetros de entrada

Puede ejecutar el flujo de trabajo Hola a todos validando sus parámetros de entrada respecto a las definiciones de su presentación.

- 1 Realice una solicitud GET en la URL que contiene la definición del flujo de trabajo Hola a todos:

```
GET https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/
```

Recibirá la definición del flujo de trabajo en el cuerpo de la respuesta de la solicitud:

```
<xml version="1.0" encoding="UTF-8" standalone="yes">
<workflow xmlns="http://www.vmware.com/vco" customized-icon="false"
    href="https://localhost:8281/vco/api/workflows/
CF808080808080808080808080808080DA808080013086668236014a0614d16e1/">
    <relations>
        <link rel="up"
            href="https://localhost:8281/vco/api/inventory/System/Workflows/Samples>HelloWorld/" />
        <link rel="add"
            href="https://localhost:8281/vco/api/workflows/
CF808080808080808080808080808080DA808080013086668236014a0614d16e1/executions/" />
```

```
<link rel="down"
      href="https://localhost:8281/vco/api/workflows/
CF808080808080808080808080808080DA808080013086668236014a0614d16e1/executions/" />
<link rel="down"
      href="https://localhost:8281/vco/api/workflows/
CF808080808080808080808080808080DA808080013086668236014a0614d16e1/presentation/" />
<link rel="down"
      href="https://localhost:8281/vco/api/workflows/
CF808080808080808080808080808080DA808080013086668236014a0614d16e1/tasks/" />
<link rel="down"
      href="https://localhost:8281/vco/api/workflows/
CF808080808080808080808080808080DA808080013086668236014a0614d16e1/icon/" />
</relations>
<input-parameters>
  <parameter name="name" type="string" />
</input-parameters>
<output-parameters>
  <parameter name="message" type="string" />
</output-parameters>
<name>Send Hello</name>
<description></description>
</workflow>
```

- 2 Realice una solicitud GET en la URL que contiene la definición de la presentación del flujo de trabajo:

GET https://localhost:8281/vco/api/workflows/  
CF808080808080808080808080808080DA808080013086668236014a0614d16e1/presentation/

- Realice una solicitud POST en la URL que contiene las instancias de ejecución de la presentación del flujo de trabajo:

POST https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/presentation/instances/

Proporcione una instancia vacía de `execution-context` solo para crear una instancia de la presentación sin proporcionar ningún valor para los parámetros de entrada:

```
<execution-context xmlns="http://www.vmware.com/vco"/>
```

El cuerpo de respuesta contiene mensajes de error adjuntos a cada campo, que indican que los valores de los parámetros de entrada no son válidos.

```
.....
<fields>
  <field type="string" hidden="false" id="name">
    <display-name>name</display-name>
    <description>name</description>
  <messages>
    <message severity="ERROR" code="VCO-CNS0002">
      <Summary>
        The minimum number of characters allowed for this field is 3.0
      </Summary>
    </message>
  </messages>
</fields>
```

```

    </message>
  </messages>
  <constraints>
    <number-range max="15.0" min="3.0" />
  </constraints>
  .....

```

- 4 Realice una solicitud POST en la URL que contiene la instancia de la presentación particular:

POST https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA80808001308668236014a0614d16e1/presentation/instances/  
888080808080808080808080808080F8080800132145338690643f66a027ec/

En el cuerpo de la solicitud, proporcione valores para los parámetros de entrada:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<execution-context xmlns="http://www.vmware.com/vco">
  <parameters>
    <parameter name="name" type="string">
      <string>John Smith</string>
    </parameter>
  </parameters>
</execution-context>
```

En el cuerpo de respuesta de la solicitud, puede comprobar si los valores de los parámetros de entrada son válidos:

```
<execution started-by="vcoadmin" .... valid="true".....>
```

- 5 Si la presentación es válida, ejecute el flujo de trabajo realizando una solicitud POST en la URL que contiene las ejecuciones del flujo de trabajo:

`POST https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/executions/`

En el cuerpo de la solicitud, transfiera valores a los parámetros de entrada del flujo de trabajo. Utilice los mismos valores que se devuelven como parámetros de salida de la presentación del flujo de trabajo o utilice directamente el cuerpo de la última solicitud POST realizada en la presentación del flujo de trabajo.

## Interactuar con un flujo de trabajo mientras se ejecuta

La API de REST de Orchestrator permite realizar varias operaciones con un flujo de trabajo durante su ejecución. Puede obtener el estado de un flujo de trabajo en ejecución, responder a una interacción de un usuario en espera y cancelar la ejecución de un flujo de trabajo.

## Obtener objetos de ejecución de flujos de trabajo y comprobar el estado de flujos de trabajo

Puede obtener información acerca de las ejecuciones de un flujo de trabajo, como las fechas de inicio y fin, el estado de la ejecución y los valores de los parámetros de entrada. También puede obtener registros generados para la ejecución de un flujo de trabajo.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

### Procedimiento

- 1 Recupere la definición del flujo de trabajo cuyo estado desea comprobar realizando una solicitud GET en la URL del flujo de trabajo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
```

Recibirá la definición del flujo de trabajo en el cuerpo de la respuesta de la solicitud. La definición del flujo de trabajo contiene un vínculo a las instancias de ejecución del flujo de trabajo.

- 2 Recupere las instancias de ejecución disponibles del flujo de trabajo realizando una solicitud GET en su URL:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/  
executions/
```

El cuerpo de respuesta de la solicitud incluye las instancias de ejecución disponibles del flujo de trabajo; allí puede ver las fechas de inicio y fin de cada ejecución del flujo de trabajo así como su estado y el usuario que la inició.

- 3 (opcional) Para obtener más detalles sobre una determinada ejecución del flujo de trabajo, haga una solicitud GET en la dirección URL de esa ejecución:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/  
executions/{ID_ejecución}/
```

En el cuerpo de la respuesta de la solicitud recibirá la representación en XML de esa ejecución del flujo de trabajo en concreto. Puede comprobar los valores de los parámetros de entrada que se han transferido en esta ejecución, el usuario que inició la ejecución, las fechas de inicio y fin, así como el estado de la ejecución.

- 4** (opcional) Para recuperar los registros que se han generado para la ejecución del flujo de trabajo, haga una solicitud GET en la dirección URL que contiene los registros:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
executions/{ID_ejecución}/logs/
```

- 5 (opcional) Para recuperar los registros adicionales acerca del estado de la ejecución, haga una solicitud GET en la dirección URL que contiene el estado del flujo de trabajo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
executions/{ID_ejecución}/state/
```

Ejemplo: obtener las ejecuciones del flujo de trabajo Hola a todos y comprobar el estado de una ejecución en concreto

Si ha ejecutado el flujo de trabajo Hola a todos, puede obtener los objetos de ejecución disponibles y comprobar sus detalles.

- 1 Obtenga la definición del flujo de trabajo Hola a todos realizando una solicitud GET en la dirección URL que contiene la definición:

```
GET https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/
```

- Obtenga las ejecuciones disponibles del flujo de trabajo realizando una solicitud GET en la dirección URL que contiene los objetos de ejecución del flujo de trabajo:

```
GET https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/executions/
```

- 3 En el cuerpo de respuesta de la solicitud, seleccione una ejecución del flujo de trabajo y haga una solicitud GET para recuperarla:

```
GET https://localhost:8281/vco/api/workflows/  
CF808080808080808080808080808080DA808080013086668236014a061d16e1/executions/  
88808080808080808080808080808080A8080800132145338690643f66a027ec/
```

El cuerpo de respuesta contiene la representación en XML de la ejecución del flujo de trabajo con el ID especificado; allí podrá comprobar los detalles de esa ejecución:

```
.....
<input-parameters>
  <parameter name="name" type="string">
    <string>John Smith</string>
  </parameter>
</input-parameters>
<output-parameters>
  <parameter name="message" type="string">
    <string>Hello, John Smith!</string>
  </parameter>
</output-parameters>
```

```

</output-parameters>
<start-date>2012-01-31T14:28:40.223+03:00</start-date>
<end-date>2012-01-31T14:28:40.410+03:00</end-date>
<started-by>vcoadmin</started-by>
<name>Send Hello</name>
.....

```

## Responder a una interacción de usuario en espera

Puede responder a una interacción de usuario en espera de un flujo de trabajo mediante la API de REST de Orchestrator.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

### Procedimiento

- 1 Recupere la lista de todos los objetos de interacción del usuario realizando una solicitud GET en la URL que contiene los objetos de interacción del usuario disponibles, o filtrando solamente las interacciones del usuario que están en espera:

URL	Descripción
<b>https:// host_orchestrator:puerto/vco/api/ catalog/System/UserInteraction</b>	Contiene los objetos de interacción del usuario disponibles en Orchestrator
<b>https:// host_orchestrator:puerto/vco/api/ catalog/System/UserInteraction? status=0</b>	Filtra solo los objetos de interacción del usuario que están en espera.

Recibe una lista de los objetos de interacción del usuario disponibles. Las interacciones del usuario que están en espera tienen un atributo con el nombre `state` y el valor `waiting`.

- 2 Realice una solicitud GET en la URL que contiene el elemento de inventario de la interacción del usuario en espera a la que desea responder:

```
GET https://{host_orchestrator}:{puerto}/vco/api/catalog/System/UserInteraction/
{ID_interacción_usuario}/
```

El elemento de inventario contiene un vínculo a la instancia de interacción del usuario. La instancia de interacción del usuario está asociada con la ejecución de un flujo de trabajo específico.



- Haga una solicitud POST en la dirección URL de la instancia de interacción del usuario para la ejecución del flujo de trabajo específico:

```
POST https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
executions/{ID_ejecución}/interaction/
```

- 4 Proporcione valores para los parámetros de entrada de la interacción del usuario en un elemento `execution-context` del cuerpo de la solicitud.

## Resultados

Cuando usted responde correctamente a una interacción del usuario, la API de REST devuelve el estado 204.

Ejemplo: Responder a la interacción del usuario del flujo de trabajo interactivo Hola a todos

Puede ejecutar el flujo de trabajo interactivo de ejemplo Hola a todos y responder a su interacción del usuario.

- 1 Busque la interacción del usuario en espera del flujo de trabajo realizando una solicitud GET en el endpoint de los objetos de interacción de usuario del servicio del catálogo:

```
GET https://localhost:8281/vco/api/catalog/System/UserInteraction?status=0
```

- 2 Localice el objeto de inventario de interacción del usuario para el flujo de trabajo interactivo "Hola a todos" y realice una solicitud GET como URL:

[illegible]

- 3 Haga una solicitud POST en la dirección URL de los objetos de interacción del usuario para la ejecución del flujo de trabajo en curso actualmente:

`POST https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080E6808080013086668236014a061d16e1/executions/  
888080808080808080808080808080578080800132145338690643f66a027ec/interaction/`

Proporcione un valor del parámetro de entrada en el cuerpo de la solicitud:

```
<execution-context xmlns="http://www.vmware.com/vco">
  <parameters>
    <parameter name="name" type="string">
      <string>John Smith</string>
    </parameter>
  </parameters>
</execution-context>
```

### Responder a una interacción del usuario tras validar los parámetros de entrada

La presentación de una interacción del usuario podría definir restricciones para los valores que se pueden transferir a los parámetros de entrada del flujo de trabajo. Cuando responde a una

interacción del usuario, puede validar los valores que se transfieren a los parámetros de entrada con las restricciones definidas en la presentación de la interacción del usuario.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

### Procedimiento

- 1 Recupere la lista de todos los objetos de interacción del usuario; para ello, realice una solicitud GET en la URL que contiene los objetos de interacción del usuario disponibles o bien filtre únicamente las interacciones del usuario que están en espera:

URL	Descripción
<b>https:// host_orchestrator:puerto/vco/api/ catalog/System/UserInteraction</b>	Contiene los objetos de interacción del usuario disponibles en Orchestrator.
<b>https:// host_orchestrator:puerto/vco/api/ catalog/System/UserInteraction? status=0</b>	Filtra solo los objetos de interacción del usuario que están en espera.

Recibe una lista de los objetos de interacción del usuario disponibles. Las interacciones del usuario que están en espera tienen un atributo con el nombre `state` y el valor `waiting`.

- 2 Realice una solicitud GET en la URL que contiene el elemento de inventario de la interacción del usuario en espera a la que se desea responder:

```
GET https://{host_orchestrator}:{puerto}/vco/api/catalog/System/UserInteraction/  
{ID_interacción_usuario}/
```

El cuerpo de la respuesta contiene un vínculo a la instancia de interacción del usuario. La instancia de interacción del usuario está asociada con la ejecución de un determinado flujo de trabajo.

- 3 Realice una solicitud GET en la dirección URL de la instancia de interacción del usuario:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/  
executions/{ID_ejecución}/interaction/
```

En el cuerpo de la respuesta, se incluye un vínculo a la presentación de la interacción del usuario.

- 4 Haga una solicitud GET en la dirección URL de la presentación de la interacción del usuario:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/executions/  
{ID_ejecución}/interaction/presentation/
```

Recibe la definición de la presentación de la interacción del usuario en el cuerpo de la respuesta.

- 5 En la definición de la presentación, compruebe las restricciones de los valores que puede transferir a los parámetros de entrada.
- 6 Ejecute la presentación de la interacción del usuario realizando una solicitud POST en la URL en la que se encuentran las instancias de la presentación:

```
POST https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
executions/{ID_ejecución}/interaction/presentation/instances/
```

- 7 En el cuerpo de la solicitud, proporcione valores para los parámetros de entrada en un elemento execution-context.

En el cuerpo de la respuesta, recibe la instancia de la presentación de la interacción del usuario. Si los valores que ha transferido a los parámetros de entrada son válidos, encuentra un atributo valid="true" en el elemento execution. En el elemento output-parameters, se encuentran los valores válidos para los parámetros de entrada que puede utilizar para responder a la interacción del usuario.

- 8 Responda a la interacción del usuario realizando una solicitud POST en la URL en la que se encuentra la instancia de interacción del usuario:

```
POST https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
executions/{ID_ejecución}/interaction/
```

- 9 En el cuerpo de la solicitud, transfiera un contexto execution-context con los valores para los parámetros de entrada.

Puede utilizar el mismo cuerpo de solicitud que el de la solicitud POST que realizó en la URL para la presentación de la interacción del usuario.

## Resultados

Si la última solicitud fue correcta, recibe un código de estado 204 y un cuerpo de respuesta vacío.

## Ejemplo: Responder a la interacción del usuario del flujo de trabajo interactivo "Hola a todos" validando los parámetros de entrada

Puede responder a la interacción del usuario del flujo de trabajo interactivo "Hola a todos" validando los valores de los parámetros de entrada con las restricciones definidas en la presentación de la interacción del usuario.

- 1 Busque las interacciones del usuario en espera del flujo de trabajo; para ello, realice una solicitud GET en el endpoint de los objetos de interacción de usuario del servicio del catálogo:

```
GET https://localhost:8281/vco/api/catalog/System/UserInteraction?status=0
```

- 2 Localice el objeto de inventario de interacción del usuario para el flujo de trabajo interactivo "Hola a todos" y realice una solicitud GET como URL:

```
GET https://localhost:8281/vco/api/catalog/System/UserInteraction/  
888080808080808080808080808080805A8080800132145338690643f66a027ec/
```

- 3 Realice una solicitud GET en la dirección URL de la instancia de interacción del usuario:

```
GET https://localhost:8281/vco/api/catalog/System/UserInteraction/  
888080808080808080808080808080805A8080800132145338690643f66a027ec/interaction/
```

- 4 Realice una solicitud GET en la URL de la presentación de interacción del usuario:

```
GET https://localhost:8281/vco/api/catalog/System/UserInteraction/  
88808080808080808080808080808080A8080800132145338690643f66a027ec/interaction/presentation/
```

La presentación define el parámetro de entrada como obligatorio y contiene una restricción para la longitud de la cadena que se puede transferir.

- 5 Realice una solicitud POST en la URL que contiene las instancias de la presentación de interacción del usuario:

[illegible]

Proporcione un valor del parámetro de entrada en el cuerpo de la solicitud:

```
<execution-context xmlns="http://www.vmware.com/vco">
  <parameters>
    <parameter name="name" type="string">
      <string>John Smith</string>
    </parameter>
  </parameters>
</execution-context>
```

El elemento `execution` del cuerpo de respuesta contiene un atributo `valid="true"`, que indica que el valor del parámetro de entrada es válido respecto a las restricciones en la presentación de la interacción del usuario. El valor válido se enumera en el elemento `output-parameters`:

```
.....
<output-parameters>
  <parameter name="name" type="string">
    <string>John Smith</string>
  </parameter>
</output-parameters>
.....
```

- 6 Realice una solicitud POST en la URL de la instancia de interacción del usuario transfiriendo el mismo cuerpo de solicitud que en la solicitud POST del paso 5.

[illegible]

## Cancelar la ejecución de un flujo de trabajo

Puede cancelar la ejecución de un flujo de trabajo con la API de REST de Orchestrator.

## Procedimiento

- 1 Recupere la definición del flujo de trabajo; para ello, realice una solicitud GET en la URL de la definición del flujo de trabajo:

GET `https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/`

La definición del flujo de trabajo contiene un vínculo a los objetos de ejecución disponibles del flujo de trabajo.

- 2 Consiga las ejecuciones de flujo de trabajo disponibles realizando una solicitud GET a la URL que contiene los objetos de ejecución disponibles para el flujo de trabajo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
executions/
```

- 3** En la lista de ejecuciones de flujo de trabajo disponibles, seleccione la que desee cancelar y realice una solicitud DELETE como URL:

```
DELETE https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
executions/{ID_ejecución}/state
```

## Recuperar las interacciones de un grupo de trabajo

Puede recuperar la lista de todas las interacciones del usuario de un flujo de trabajo mediante la API de REST de Orchestrator.

## Procedimiento

- 1 Recupere la definición del flujo de trabajo realizando una solicitud GET en la URL de la definición del flujo de trabajo:

GET https://{host\_orchestrator}:{puerto}/vco/api/workflows/{ID\_flujo\_trabajo}/

- 2 Obtenga la lista de las interacciones del flujo de trabajo creando una solicitud GET en la URL de las interacciones del flujo de trabajo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/interactions/
```

### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200 y una lista de todas las interacciones del usuario disponibles para el flujo de trabajo.

## Acceder a un esquema de flujo de trabajo

Puede acceder a la imagen de esquema de un flujo de trabajo mediante la API de REST de Orchestrator.

### Procedimiento

- 1 Recupere la definición del flujo de trabajo; para ello, realice una solicitud GET en la URL de la definición del flujo de trabajo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
```

- 2 Obtenga la imagen de esquema del flujo de trabajo; para ello, cree una solicitud GET en la URL del esquema de flujo de trabajo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/schema/
```

### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200 y los datos binarios de la imagen, que representan el esquema del flujo de trabajo. El tipo de contenido de respuesta está establecido en un tipo de medios correcto, por ejemplo Content-Type: image/png.

## Trabajar con tareas

Gracias al servicio de tareas de la API de REST de Orchestrator, puede llevar a cabo cualquier operación relacionada con la administración de tareas en Orchestrator. Puede crear una tarea para planificar un flujo de trabajo, modificar las propiedades de una tarea que ya existe, eliminar una tarea, etcétera.

Orchestrator admite un máximo de 50 tareas programadas.

### Crear una tarea

Puede crear una tarea para programar un flujo de trabajo con la API de REST de Orchestrator.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

## Procedimiento

- 1 Recupere la definición del flujo de trabajo para el que desea crear una tarea. Para ello, efectúe una solicitud GET en la URL del flujo de trabajo:

GET `https://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/`

En la definición del flujo de trabajo, puede ver el nombre, el ID del flujo de trabajo y sus parámetros de entrada.

- 2 Para crear una tarea para el flujo de trabajo, haga una solicitud POST en la URL del servicio de tarea:

POST https://{*host\_orchestrator*}:{*puerto*}/vco/api/tasks/

- 3** En el cuerpo de la solicitud, proporcione los parámetros para la tarea nueva en un elemento task.

## Resultados

Si la última solicitud fue correcta, la API responde con el código de estado 202 y un cuerpo de respuesta vacío.

## Ejemplo: Crear una tarea para el flujo de trabajo Hola a todos

Puede crear una tarea que programe la ejecución del flujo de trabajo Hola a todos en el minuto 15 de cada hora a partir de una fecha determinada.

- 1 Haga una solicitud GET en la URL del flujo de trabajo Hola a todos para recuperar su definición:

```
GET https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/
```

- Haga una solicitud POST en la URL del servicio de tarea proporcionando los parámetros de la tarea nueva en el cuerpo de la solicitud:

```
POST https://localhost:8281/vco/api/tasks/
```

```
<task xmlns="http://www.vmware.com/vco">
  <name>Send Hello Task</name>
  <recurrence-cycle>every-hours</recurrence-cycle>
  <recurrence-start-date>2012-01-31T11:00:00+00:00</recurrence-start-date>
  <recurrence-end-date>2012-02-05T11:00:00+00:00</recurrence-end-date>
  <recurrence-pattern>15:15</recurrence-pattern>
  <input-parameters>
```

```
<parameter name="name" type="string">  
    <string>John Smith</string>  
</parameter>  
</input-parameters>  
<workflow href="https://localhost:8281/vco/api/workflows/  
CF8080808080808080808080808080DA808080013086668236014a0614d16e1/">  
    <name>Send Hello</name>  
</workflow>  
<start-mode>normal</start-mode>  
</task>
```

## Modificar una tarea

Puede cambiar las propiedades de una tarea existente mediante la API de REST de Orchestrator.

Solo se pueden añadir propiedades de programación nuevas a una tarea o cambiar los valores de las propiedades ya existentes. Si desea sustituir las propiedades de programación de una tarea, debe eliminar la tarea y crear una nueva.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

## Procedimiento

- 1** Haga una solicitud GET en la dirección URL de la tarea que desea modificar:

GET https://{*host\_orchestrator*}:{*puerto*}/vco/api/tasks/{*ID\_tarea*}/

- 2 Compruebe las propiedades de la tarea en el cuerpo de respuesta de la solicitud.
- 3 Para modificar la tarea, haga una solicitud POST a la dirección URL de la tarea proporcionando las nuevas propiedades de la tarea en un elemento `task-data` en el cuerpo de la solicitud.

## Resultados

Si la solicitud POST es correcta, la API devuelve un código de estado 200 y la tarea actualizada en el cuerpo de respuesta.

### Ejemplo: Actualizar la tarea de ejemplo Hola a todos

Es posible actualizar las fechas inicial y final de una tarea. Es posible modificar la tarea de ejemplo que se introdujo en [Crear una tarea](#). Debe hacer una solicitud POST en la dirección URL de la tarea proporcionando las nuevas fechas inicial y final en el cuerpo de la solicitud:

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<task-data xmlns="http://www.vmware.com/vco">
  <recurrence-start-date>2012-02-01T14:00:00+02:00</recurrence-start-date>
  <recurrence-end-date>2012-02-05T14:00:00+02:00</recurrence-end-date>
</task-data>
```



## Comprobación del estado de una tarea

Puede comprobar el estado de las tareas disponibles actualmente o de todas las instancias de ejecución de una tarea en concreto.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

### Procedimiento

- ◆ Para comprobar el estado de todas las tareas disponibles, realice una solicitud GET en la URL del servicio de tarea:

```
GET https://{host_orchestrator}:{puerto}/vco/api/tasks/
```

El cuerpo de respuesta contiene las definiciones de las tareas disponibles en Orchestrator. El estado de cualquier tarea está disponible en un elemento `attribute`, cuyo nombre es `state`. Respectivamente, el valor del elemento puede ser `finished`, `pending`, `running`, etcétera.

- ◆ Para comprobar el estado de todas las ejecuciones de una determinada tarea, realice una solicitud GET en la URL donde se encuentran las ejecuciones de tareas:

```
GET https://{host_orchestrator}:{puerto}/vco/api/tasks/{ID_tarea}/executions/
```

Recibe una lista de las ejecuciones disponibles para la tarea en el cuerpo de la respuesta. El estado de cualquier ejecución está disponible en el elemento `state` del objeto de ejecución de tareas.

## Búsqueda de objetos en el inventario de Orchestrator

Puede encontrar cualquier objeto en el inventario de Orchestrator mediante el catálogo o los servicios de inventario. Puede acceder solo a un determinado subconjunto de objetos aplicando parámetros de filtro al final de las direcciones URL en las que realice solicitudes HTTP.

Puede utilizar el servicio de catálogo para buscar objetos en el inventario de Orchestrator que sean de un determinado tipo, o bien para recuperar un objeto específico por tipo o por ID. Por ejemplo, puede recuperar todos los objetos que sean del tipo `workflow` o `action`, o bien una acción o un flujo de trabajo específicos.

El servicio de inventario le permite navegar por el inventario de Orchestrator por relaciones principal-secundaria. Con el servicio de inventario, puede acceder a los objetos disponibles en una ubicación específica en el inventario de Orchestrator. Por ejemplo, puede recuperar todos los flujos de trabajo para la gestión del centro de datos navegando hasta su ubicación en el inventario de Orchestrator, es decir, `Library/vCenter/Datacenter`.

Cada servicio de la API de REST de Orchestrator admite parámetros de filtro que puede añadir al final de las direcciones URL cuando realice solicitudes HTTP. Con los parámetros de filtro, puede limitar los resultados que se reciben en el cuerpo de respuesta de una solicitud en una dirección URL determinada.

## Buscar objetos por su tipo y su ID

Puede usar el servicio de catálogo de la API de REST para buscar objetos en Orchestrator por su tipo y su ID.

### Requisitos previos

Compruebe que haya importado el paquete de ejemplo de flujos de trabajo a Orchestrator. El paquete se incluye en el archivo ZIP de aplicaciones de ejemplo de Orchestrator que puede descargarse desde la página de documentación de Orchestrator.

### Procedimiento

- 1 Haga una solicitud GET en la dirección URL del servicio de catálogo:

```
GET https://{host_orchestrator}:{puerto}/vco/api/catalog/
```

El cuerpo de respuesta de la solicitud contiene vínculos descendentes tanto a los puntos de entrada de los complementos que exponen inventarios en Orchestrator como a los objetos de sistema de Orchestrator:

- `https://{host_orchestrator}:{puerto}/vco/api/catalog/{espacio_nombres_complemento}/`
- `https://{host_orchestrator}:{puerto}/vco/api/catalog/System/`

- 2 Para acceder a objetos que expone un complemento o a los objetos de sistema de Orchestrator, haga una solicitud GET en la dirección URL del punto de entrada del catálogo del complemento o en la URL en la que residen los objetos de sistema de Orchestrator.

El cuerpo de respuesta de la solicitud contiene vínculos a los tipos de objetos que están expuestos.

- 3 Haga una solicitud GET en la dirección URL del tipo de objeto al que desee acceder:

```
GET https://{host_orchestrator}:{puerto}/vco/api/catalog/{espacio_nombres}/
{tipo_objeto}/
```

- 4 Haga una solicitud GET en la dirección URL del objeto específico que desee buscar:

```
GET https://{host_orchestrator}:{puerto}/vco/api/catalog/{espacio_nombres}/
{tipo_objeto}/{ID_objeto}/
```



## Ejemplo: Buscar el flujo de trabajo Hola a todos

Puede explorar el inventario de Orchestrator para buscar el flujo de trabajo Hola a todos.

- 1 Haga una solicitud GET en la dirección URL del servicio de inventario:

```
GET https://localhost:8281/vco/api/inventory/
```

- 2 Haga una solicitud GET en la dirección URL en la que residan los objetos de sistema de Orchestrator:

```
GET https://localhost:8281/vco/api/inventory/System/
```

- 3 Haga una solicitud GET en la dirección URL en la que residan todos los flujos de trabajo de Orchestrator:

```
GET https://localhost:8281/vco/api/inventory/System/Workflows/
```

- 4 Haga una solicitud GET en la dirección URL de la categoría de flujos de trabajo de muestra:

```
GET https://localhost:8281/vco/api/inventory/System/Workflows/Samples/
```

- 5 Utilice el vínculo descendente de la categoría de flujos de trabajo Hola a todos en los que buscar el flujo de trabajo Hola a todos.

## Aplicar filtros

Los servicios de la API de REST de Orchestrator son compatibles con los parámetros de URL adicionales que le permiten restringir los objetos que devuelven las solicitudes HTTP a la API.

Cada URL a un recurso al que puede acceder mediante la API de REST es compatible con distintos parámetros de consulta. Para saber qué parámetros de consulta son aplicables a una URL, consulte la documentación de referencia de la *vRealize Orchestrator API de REST*.

### Procedimiento

- ◆ Para restringir los resultados de una solicitud a una URL determinada, aplique filtros al final de la URL:

*URL?filtro\_1& filtro\_2&filtro\_3&....&filtro\_N*. Cada filtro contiene parámetros de consulta válidos para la URL correspondiente. Para ver más información sobre los parámetros de consulta válidos para cada URL, consulte la documentación de referencia de la API de REST de Orchestrator.

## Ejemplo: filtrar flujos de trabajo

Si busca flujos de trabajo cuyo nombre contiene una palabra específica, como por ejemplo "almacén de datos", puede aplicar el siguiente filtro en una solicitud al servicio de catálogo:

```
GET https://localhost:8281/vco/api/catalog/System/Workflow?conditions=name~datastore
```

Para limitar la cantidad de flujos de trabajo que se devuelven hasta un número determinado, por ejemplo cinco, aplique un filtro adicional a la solicitud:

```
GET https://localhost:8281/vco/api/catalog/System/Workflow?
conditions=name~datastore&maxResult=5
```

## Importar y exportar objetos de Orchestrator

La API de REST de Orchestrator proporciona servicios web que se pueden utilizar para importar y exportar flujos de trabajo, acciones, paquetes, recursos y elementos de configuración.

### Importar un flujo de trabajo

Puede importar un flujo de trabajo mediante la API de REST de Orchestrator.

Según la biblioteca de su aplicación cliente de REST, puede utilizar código personalizado que defina las propiedades del flujo de trabajo.

#### Requisitos previos

El contenido binario del flujo de trabajo debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

#### Procedimiento

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del flujo de trabajo que desee importar.
- 2 Haga una solicitud POST en la dirección URL de los objetos del flujo de trabajo:

```
POST http://{host_orchestrator}:{puerto}/vco/api/workflows/
```

#### Resultados

Si la solicitud POST es correcta, recibirá el código de estado 202.

### Exportar un flujo de trabajo

Puede exportar un flujo de trabajo mediante la API de REST de Orchestrator y descargarlo como un archivo.

#### Procedimiento

- 1 En una aplicación cliente de REST, añada un encabezado de solicitud con los siguientes valores.
  - **Nombre:** `accept`
  - **Valor:** `application/zip`

- 2 Haga una solicitud GET en la dirección URL del flujo de trabajo que desee exportar:

```
GET http://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
```

### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. El contenido binario del flujo de trabajo está disponible como archivo adjunto con un nombre de archivo predeterminado *nombre\_flujo\_trabajo.workflow*. Puede guardar el archivo con una aplicación cliente de REST.

## Importar una acción

Puede importar una acción mediante la API de REST de Orchestrator.

Según la biblioteca de su aplicación cliente de REST, puede utilizar código personalizado que defina las propiedades de la acción.

### Requisitos previos

El contenido binario de la acción debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

### Procedimiento

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades de la acción que desee importar.
- 2 Haga una solicitud POST en la dirección URL de los objetos de la acción:

```
POST http://{host_orchestrator}:{puerto}/vco/api/actions/
```

### Resultados

Si la solicitud POST es correcta, recibirá el código de estado 202.

## Exportar una acción

Puede exportar una acción mediante la API de REST de Orchestrator y descargarla como un archivo.

### Procedimiento

- 1 En una aplicación cliente de REST, añada un encabezado de solicitud con los siguientes valores.
  - **Nombre:** `accept`
  - **Valor:** `application/zip`

- 2 Haga una solicitud GET en la dirección URL de la acción que desee exportar:

```
GET http://{host_orchestrator}:{puerto}/vco/api/actions/{ID_acción}/
```

### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. El contenido binario de la acción está disponible como archivo adjunto con un nombre de archivo predeterminado *nombre\_acción.action*. Puede guardar el archivo con una aplicación cliente de REST.

## Importar un paquete

Puede importar un paquete mediante la API de REST de Orchestrator.

Según la biblioteca de su aplicación cliente de REST, puede utilizar código personalizado que defina las propiedades del paquete.

De manera predeterminada, si importa un paquete de Orchestrator con nombre duplicado, el paquete existente no se sobrescribe. Puede especificar si desea sobrescribir los paquetes existentes utilizando un parámetro en la solicitud.

De manera predeterminada, los paquetes de Orchestrator se importan con los valores de atributos de los elementos de configuración. Puede importar un paquete sin valores de atributos utilizando un parámetro en la solicitud.

De manera predeterminada, las etiquetas incluidas en los paquetes de Orchestrator se importan pero si ya existen las mismas etiquetas en el servidor de Orchestrator, se conservan los valores de las etiquetas existentes. Puede especificar si desea que se conserven los valores de etiqueta existentes utilizando los parámetros de la solicitud.

### Requisitos previos

El contenido binario del paquete debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

### Procedimiento

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del paquete que desee importar.
- 2 Haga una solicitud POST en la dirección URL de los objetos del paquete:

```
POST http://{host_orchestrator}:{puerto}/vco/api/packages/
```

- 3 (opcional) Para importar un paquete y sobrescribir un paquete existente con el mismo nombre, utilice el parámetro `overwrite` de la solicitud POST:

```
POST http://{host_orchestrator}:{puerto}/vco/api/packages/?overwrite=true
```

- 4 (opcional) Para importar un paquete sin los valores de atributos de los elementos de configuración del paquete, utilice el parámetro `importConfigurationAttributeValues` de la solicitud POST:

```
POST http://{host_orchestrator}:{puerto}/vco/api/packages/?
importConfigurationAttributeValues=false
```

- 5 (opcional) Para importar un paquete sin las etiquetas que contiene, utilice el parámetro `tagImportMode` de la solicitud POST:

```
POST http://{host_orchestrator}:{puerto}/vco/api/packages/?tagImportMode=DoNotImport
```

- 6 (opcional) Para importar un paquete con las etiquetas que contiene y sobrescribir los valores de etiqueta existentes, utilice el parámetro `tagImportMode` de la solicitud POST:

```
POST http://{host_orchestrator}:{puerto}/vco/api/packages/?
tagImportMode=ImportAndOverwriteExistingValue
```

## Resultados

Si la solicitud POST es correcta, recibirá el código de estado 202.

## Exportar un paquete

Puede exportar un paquete mediante la API de REST de Orchestrator y descargar el paquete como un archivo.

De manera predeterminada, los paquetes de Orchestrator se exportan con los valores de atributos de los elementos de configuración y las etiquetas globales. Puede exportar un paquete sin valores de atributos ni etiquetas globales utilizando los parámetros de la solicitud. También puede especificar un nombre personalizado para el archivo del paquete que descargue.

### Procedimiento

- 1 En una aplicación cliente de REST, añada un encabezado de solicitud con los siguientes valores.
  - **Nombre:** `accept`
  - **Valor:** `application/zip`
- 2 Haga una solicitud GET en la dirección URL del paquete que desee exportar:

```
GET http://{host_orchestrator}:{puerto}/vco/api/packages/{nombre_paquete}/
```



- 3 (opcional) Para definir un nombre personalizado para el paquete exportado, utilice el parámetro `packageName` de la solicitud GET:

```
GET http://{host_orchestrator}:{puerto}/vco/api/packages/{nombre_paquete}/?
packageName={nombre_personalizado}
```

- 4 (opcional) Para exportar un paquete sin los valores de atributos de los elementos de configuración del paquete, utilice el parámetro `exportConfigurationAttributeValues` de la solicitud GET:

```
GET http://{host_orchestrator}:{puerto}/vco/api/packages/{nombre_paquete}/?
exportConfigurationAttributeValues=false
```

- 5 (opcional) Para exportar un paquete sin las etiquetas globales, utilice el parámetro `exportGlobalTags` de la solicitud GET:

```
GET http://{host_orchestrator}:{puerto}/vco/api/packages/{nombre_paquete}/?
exportGlobalTags=false
```

## Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. El contenido binario del paquete está disponible como archivo adjunto con un nombre de archivo predeterminado `nombre_paquete.package`. Puede guardar el archivo con una aplicación cliente de REST.

## Importar un recurso

Puede importar un recurso mediante la API de REST de Orchestrator.

Según la biblioteca de su aplicación cliente de REST, puede utilizar código personalizado que defina las propiedades del recurso.

### Requisitos previos

El contenido binario del recurso debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

### Procedimiento

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del recurso que desee importar.
- 2 Haga una solicitud POST en la dirección URL de los objetos del recurso:

```
POST http://{host_orchestrator}:{puerto}/vco/api/resources/
```

## Resultados

Si la solicitud POST es correcta, recibirá el código de estado 202.

## Exportar un recurso

Puede exportar un recurso mediante la API de REST de Orchestrator.

### Procedimiento

- 1 En una aplicación cliente de REST, añada un encabezado de solicitud con los siguientes valores.
  - **Nombre:** `accept`
  - **Valor:** `application/octet-stream`
- 2 Haga una solicitud GET en la dirección URL del recurso que desee exportar:

```
GET http://{host_orchestrator}:{puerto}/vco/api/resources/{ID_recurso}/
```

## Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. El contenido del recurso está disponible en el cuerpo de la respuesta.

## Importar un elemento de configuración

Puede importar un elemento de configuración mediante la API de REST de Orchestrator.

Según la biblioteca de su aplicación cliente de REST, puede utilizar código personalizado que defina las propiedades del elemento de configuración.

### Requisitos previos

El contenido binario del elemento de configuración debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

### Procedimiento

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del elemento de configuración que desee importar.
- 2 Haga una solicitud POST en la dirección URL de los objetos del elemento de configuración:

```
POST http://{host_orchestrator}:{puerto}/vco/api/configurations/
```

## Resultados

Si la solicitud POST es correcta, recibirá el código de estado 202.

## Exportar un elemento de configuración

Puede exportar un elemento de configuración mediante la API de REST de Orchestrator.

### Procedimiento

- 1 En una aplicación cliente de REST, añada un encabezado de solicitud con los siguientes valores.
  - **Nombre:** `accept`
  - **Valor:** `application/vcoobject+xml`
- 2 Haga una solicitud GET en la dirección URL del elemento de configuración que desee exportar:

```
GET http://{host_orchestrator}:{puerto}/vco/api/configurations/  
{ID_elemento_configuración}/
```

### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. El contenido del elemento de configuración está disponible en el cuerpo de la respuesta.

## Eliminar objetos de Orchestrator

La API de REST de Orchestrator proporciona servicios web que se pueden utilizar para eliminar flujos de trabajo, acciones, paquetes, recursos y elementos de configuración.

### Eliminar un flujo de trabajo

Puede eliminar un flujo de trabajo mediante la API de REST de Orchestrator.

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID del flujo de trabajo de la lista de flujos de trabajo devueltos:

```
puerto}/vco/api/workflows/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud DELETE en la dirección URL del flujo de trabajo:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/workflows/{id_flujo_trabajo}/
```

### Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 200 y el cuerpo de la respuesta está vacío.

## Eliminar una acción

Puede eliminar una acción mediante la API de REST de Orchestrator.

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID de la acción de la lista de acciones devueltas:

```
puerto}/vco/api/actions/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud DELETE en la dirección URL de la acción:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/actions/{ID_acción}/
```

### Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 200 y el cuerpo de la respuesta está vacío.

## Eliminar un paquete

Puede eliminar un paquete mediante la API de REST de Orchestrator.

Cuando se elimina un paquete, no se eliminan los elementos del paquete. Si desea eliminar el contenido de un paquete, debe proporcionar un parámetro de opción.

### Procedimiento

- 1 Haga una solicitud GET y recupere el nombre del paquete de la lista de paquetes devueltos:

```
puerto}/vco/api/packages/
```

```
GET http://{orchestrator_host}:{
```

- Haga una solicitud DELETE en la URL del paquete. Si desea eliminar elementos del paquete, proporcione un parámetro de opción al final de la solicitud:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/packages/{nombre_paquete}/?
option={parámetro}
```

Parámetro	Descripción
<b>deletePackage</b>	El paquete se elimina, pero su contenido se conserva.
<b>deletePackageWithContent</b>	Se eliminan el paquete y todo su contenido. Si otros paquetes comparten elementos con el paquete eliminado, los elementos compartidos se eliminan de los otros paquetes.
<b>deletePackageKeepingShared</b>	Se eliminan el paquete y el contenido no compartido. No se eliminan los elementos que se comparten con otros paquetes.

Si no se proporciona un parámetro de opción, se utiliza el parámetro **deletePackage** predeterminado.

#### Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 200 y el cuerpo de la respuesta está vacío.

## Eliminar un recurso

Puede eliminar un recurso mediante la API de REST de Orchestrator.

#### Procedimiento

- Haga una solicitud GET y recupere el ID del flujo de recurso de la lista de recursos devueltos:

```
{puerto}/vco/api/resources/
```

```
GET http://{orchestrator_host}:{
```

- Haga una solicitud DELETE en la dirección URL del recurso:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/resources/{ID_recurso}/
```

#### Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 200 y el cuerpo de la respuesta está vacío.

## Eliminar un elemento de configuración

Puede eliminar un elemento de configuración mediante la API de REST de Orchestrator.

**Procedimiento**

- 1 Haga una solicitud GET y recupere el ID del elemento de configuración de recurso de la lista de elementos de configuración devueltos:

```
puerto}/vco/api/configurations/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud DELETE en la URL del elemento de configuración:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/configurations/  
{ID_elemento_configuración}/
```

**Resultados**

Si la solicitud DELETE es correcta, recibirá el código de estado 200 y el cuerpo de la respuesta está vacío.

## Configurar permisos en objetos de Orchestrator

Puede establecer permisos personalizados para un objeto de Orchestrator mediante la API de REST. Para establecer los permisos, debe realizar una solicitud POST en la URL de los permisos del objeto y definir los permisos en el cuerpo de la solicitud.

También puede utilizar la API de REST de Orchestrator para recuperar información sobre permisos de un objeto o eliminar permisos.

### Permisos de API de REST

Al configurar permisos mediante la API de REST de Orchestrator, debe usar un conjunto de caracteres para definir los permisos.

Puede definir los permisos para un elemento incluyendo una secuencia de caracteres en la etiqueta <rights> del cuerpo de una solicitud POST.

Los caracteres que puede usar para configurar permisos mediante la API de REST de Orchestrator tienen significados específicos.

**Tabla 2-1. Conjunto de caracteres de permisos de API de REST de Orchestrator**

Carácter	Descripción
r	Otorga permisos de vista.
x	Otorga permisos de ejecución.
i	Otorga permisos de inspección.
c	Otorga permisos de edición.
a	Otorga permisos administrativos.

## Ejemplo: Sintaxis para configurar permisos

Puede seguir la sintaxis del ejemplo siguiente en el cuerpo de una solicitud POST en la dirección URL de permisos de un elemento de Orchestrator.

```
<permissions xmlns="http://www.vmware.com/vco">
  <permission>
    <principal>cn=vcousers,ou=vco,dc=appliance</principal>
    <rights>ric</rights>
  </permission>
</permissions>
```

Al configurar permisos ric en la etiqueta <rights> del cuerpo de la solicitud, permite que los miembros del grupo de usuarios vcousers vean, inspeccionen y editen el elemento de Orchestrator.

## Recuperar los permisos de un flujo de trabajo

Puede recuperar información sobre los permisos de un flujo de trabajo mediante la API de REST de Orchestrator.

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID del flujo de trabajo de la lista de flujos de trabajo devueltos:

```
puerto}/vco/api/workflows/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud GET en la dirección URL de los permisos del flujo de trabajo:

```
GET http://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
permissions/
```

### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. En el cuerpo de la respuesta hay disponible información sobre los permisos del flujo de trabajo.

## Eliminar los permisos de un flujo de trabajo

Puede eliminar los permisos de un flujo de trabajo mediante la API de REST de Orchestrator. Puede eliminar los permisos existentes de un flujo de trabajo antes de definir nuevos permisos.

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID del flujo de trabajo de la lista de flujos de trabajo devueltos:

```
puerto}/vco/api/workflows/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud DELETE en la dirección URL de los permisos del flujo de trabajo:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
permissions/
```

### Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 204 y el cuerpo de la respuesta estará vacío.

## Permisos de un flujo de trabajo

Es posible establecer los permisos de un flujo de trabajo mediante la API de REST de Orchestrator.

### Requisitos previos

Revise los tipos de permisos que puede establecer y la sintaxis que puede usar en el cuerpo de la solicitud. Consulte [Permisos de API de REST](#).

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID del flujo de trabajo de la lista de flujos de trabajo devueltos:

```
puerto}/vco/api/workflows/
```

```
GET http://{orchestrator_host}:{
```

- 2 En la aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del flujo de trabajo para el que desee establecer permisos.
- 3 En el cuerpo de la solicitud, especifique los permisos que desee establecer.
- 4 Haga una solicitud POST en la dirección URL de los permisos del flujo de trabajo:

```
POST http://{host_orchestrator}:{puerto}/vco/api/workflows/{ID_flujo_trabajo}/
permissions/
```



## Resultados

Si la solicitud POST es correcta, recibirá el código de estado 201. En el cuerpo de la respuesta hay disponible información sobre los permisos del flujo de trabajo.

## Recuperar los permisos de una acción

Puede recuperar información sobre los permisos de una acción mediante la API de REST de Orchestrator.

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID de la acción de la lista de acciones devueltas:

```
puerto}/vco/api/actions/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud GET en la dirección URL de los permisos de la acción:

```
GET http://{host_orchestrator}:{puerto}/vco/api/actions/{ID_acción}/permissions/
```

## Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. En el cuerpo de la respuesta hay disponible información sobre los permisos de la acción.

## Eliminar los permisos de una acción

Puede eliminar los permisos de una acción mediante la API de REST de Orchestrator. Puede eliminar los permisos existentes de una acción antes de definir nuevos permisos.

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID de la acción de la lista de acciones devueltas:

```
puerto}/vco/api/actions/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud DELETE en la dirección URL de los permisos de la acción:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/actions/{ID_acción}/permissions/
```

## Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 204 y el cuerpo de la respuesta estará vacío.

## Permisos de una acción

Puede establecer los permisos de una acción mediante la API de REST de Orchestrator.

### Requisitos previos

Revise los tipos de permisos que puede establecer y la sintaxis que puede usar en el cuerpo de la solicitud. Consulte [Permisos de API de REST](#).

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID de la acción de la lista de acciones devueltas:

```
puerto}/vco/api/actions/
```

```
GET http://{orchestrator_host}:{
```

- 2 En la aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades de la acción para la que desee establecer permisos.
- 3 En el cuerpo de la solicitud, especifique los permisos que desee establecer.
- 4 Haga una solicitud POST en la dirección URL de los permisos de la acción:

```
POST http://{host_orchestrator}:{puerto}/vco/api/actions/{ID_acción}/permissions/
```

### Resultados

Si la solicitud POST es correcta, recibirá el código de estado 201. En el cuerpo de la respuesta hay disponible información sobre los permisos de la acción.

## Recuperar los permisos de un paquete

Puede recuperar información sobre los permisos de un paquete mediante la API de REST de Orchestrator.

### Procedimiento

- 1 Haga una solicitud GET y recupere el nombre del paquete de la lista de paquetes devueltos:

```
puerto}/vco/api/packages/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud GET en la dirección URL de los permisos del paquete:

```
GET http://{host_orchestrator}:{puerto}/vco/api/packages/{nombre_paquete}/permissions/
```

## Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. En el cuerpo de la respuesta hay disponible información sobre los permisos del paquete.

## Eliminar los permisos de un paquete

Puede eliminar los permisos de un paquete mediante la API de REST de Orchestrator. Puede eliminar los permisos existentes de un paquete antes de definir nuevos permisos.

### Procedimiento

- 1 Haga una solicitud GET y recupere el nombre del paquete de la lista de paquetes devueltos:

```
puerto}/vco/api/packages/
```

```
GET http://{orchestrator_host}:{
```

- 2 Haga una solicitud DELETE en la dirección URL de los permisos del paquete:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/packages/{nombre_paquete}/  
permissions/
```

## Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 204 y el cuerpo de la respuesta estará vacío.

## Permisos de un paquete

Es posible establecer los permisos de un paquete mediante la API de REST de Orchestrator.

### Requisitos previos

Revise los tipos de permisos que puede establecer y la sintaxis que puede usar en el cuerpo de la solicitud. Consulte [Permisos de API de REST](#).

### Procedimiento

- 1 Haga una solicitud GET y recupere el nombre del paquete de la lista de paquetes devueltos:

```
puerto}/vco/api/packages/
```

```
GET http://{orchestrator_host}:{
```

- 2 En la aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del paquete para el que desee establecer permisos.
- 3 En el cuerpo de la solicitud, especifique los permisos que desee establecer.

- Haga una solicitud POST en la dirección URL de los permisos del paquete:

```
POST http://{host_orchestrator}:{puerto}/vco/api/packages/{nombre_paquete}/permissions/
```

#### Resultados

Si la solicitud POST es correcta, recibirá el código de estado 201. En el cuerpo de la respuesta hay disponible información sobre los permisos del paquete.

## Recuperar los permisos de un recurso

Puede recuperar información sobre los permisos de un recurso mediante la API de REST de Orchestrator.

#### Procedimiento

- Haga una solicitud GET y recupere el ID del flujo de recurso de la lista de recursos devueltos:

```
{puerto}/vco/api/resources/
```

```
GET http://{orchestrator_host}:{
```

- Haga una solicitud GET en la dirección URL de los permisos del recurso:

```
GET http://{host_orchestrator}:{puerto}/vco/api/resources/{ID_recurso}/permissions/
```

#### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. En el cuerpo de la respuesta hay disponible información sobre los permisos del recurso.

## Eliminar los permisos de un recurso

Puede eliminar los permisos de un recurso mediante la API de REST de Orchestrator. Puede eliminar los permisos existentes de un recurso antes de definir nuevos permisos.

#### Procedimiento

- Haga una solicitud GET y recupere el ID del flujo de recurso de la lista de recursos devueltos:

```
{puerto}/vco/api/resources/
```

```
GET http://{orchestrator_host}:{
```

- Haga una solicitud DELETE en la dirección URL de los permisos del recurso:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/resources/{ID_recurso}/permissions/
```

## Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 204 y el cuerpo de la respuesta estará vacío.

## Permisos de un recurso

Es posible establecer los permisos de un recurso mediante la API de REST de Orchestrator.

### Requisitos previos

Revise los tipos de permisos que puede establecer y la sintaxis que puede usar en el cuerpo de la solicitud. Consulte [Permisos de API de REST](#).

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID del flujo de recurso de la lista de recursos devueltos:

```
puerto}/vco/api/resources/
```

```
GET http://{orchestrator_host}:{
```

- 2 En la aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del recurso para el que desee establecer permisos.
- 3 En el cuerpo de la solicitud, especifique los permisos que desee establecer.
- 4 Haga una solicitud POST en la dirección URL de los permisos del recurso:

```
POST http://{host_orchestrator}:{puerto}/vco/api/resources/{ID_recurso}/permissions/
```

## Resultados

Si la solicitud POST es correcta, recibirá el código de estado 201. En el cuerpo de la respuesta hay disponible información sobre los permisos del recurso.

## Recuperar los permisos de un elemento de configuración

Puede recuperar información sobre los permisos de un elemento de configuración mediante la API de REST de Orchestrator.

### Procedimiento

- 1 Haga una solicitud GET y recupere el ID del elemento de configuración de recurso de la lista de elementos de configuración devueltos:

```
puerto}/vco/api/configurations/
```

```
GET http://{orchestrator_host}:{
```

- Haga una solicitud GET en la dirección URL de los permisos del elemento de configuración:

```
GET http://{host_orchestrator}:{puerto}/vco/api/configurations/
{ID_elemento_configuración}/permissions/
```

### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. En el cuerpo de la respuesta hay disponible información sobre los permisos del elemento de configuración.

## Eliminar los permisos de un elemento de configuración

Puede eliminar los permisos de un elemento de configuración mediante la API de REST de Orchestrator. Puede eliminar los permisos existentes de un elemento de configuración antes de definir nuevos permisos.

### Procedimiento

- Haga una solicitud GET y recupere el ID del elemento de configuración de recurso de la lista de elementos de configuración devueltos:

```
puerto}/vco/api/configurations/
```

```
GET http://{orchestrator_host}:{
```

- Haga una solicitud DELETE en la dirección URL de los permisos del elemento de configuración:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/configurations/
{ID_elemento_configuración}/permissions/
```

### Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 204 y el cuerpo de la respuesta estará vacío.

## Permisos de un elemento de configuración

Es posible establecer los permisos de un elemento de configuración mediante la API de REST de Orchestrator.

### Requisitos previos

Revise los tipos de permisos que puede establecer y la sintaxis que puede usar en el cuerpo de la solicitud. Consulte [Permisos de API de REST](#).

**Procedimiento**

- 1 Haga una solicitud GET y recupere el ID del elemento de configuración de recurso de la lista de elementos de configuración devueltos:

```
puerto}/vco/api/configurations/
```

```
GET http://{orchestrator_host}:{
```

- 2 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del elemento de configuración para el que desee establecer permisos.
- 3 En el cuerpo de la solicitud, especifique los permisos que desee establecer.
- 4 Haga una solicitud POST en la dirección URL de los permisos del elemento de configuración:

```
POST http://{host_orchestrator}:{puerto}/vco/api/configurations/  
{ID_elemento_configuración}/permissions/
```

**Resultados**

Si la solicitud POST es correcta, recibirá el código de estado 201. En el cuerpo de la respuesta hay disponible información sobre los permisos del elemento de configuración.

## Realizar operaciones con complementos

La API de REST de Orchestrator proporciona servicios web que se pueden utilizar para realizar varias operaciones con complementos.

### Recuperar información sobre complementos

Puede recuperar información de metadatos sobre todos los complementos instalados mediante la API de REST de Orchestrator.

**Procedimiento**

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades de los complementos.
- 2 Haga una solicitud GET en la dirección URL de los objetos del complemento:

```
GET http://{host_orchestrator}:{puerto}/vco/api/plugins/
```

**Resultados**

Si la solicitud GET es correcta, recibirá el código de estado 200.

## Importar un complemento

Puede importar un complemento mediante la API de REST de Orchestrator.

Según la biblioteca de su aplicación cliente de REST, puede utilizar código personalizado que defina las propiedades del complemento.

---

**Nota** No puede importar un complemento si ya hay instalado un complemento con el mismo nombre.

---

### Requisitos previos

El contenido binario del complemento debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

### Procedimiento

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del complemento que desee importar.
- 2 Haga una solicitud POST en la dirección URL de los objetos del complemento:

```
POST http://{host_orchestrator}:{puerto}/vco/api/plugins/
```

### Resultados

Si la solicitud POST es correcta, recibirá el código de estado 200.

## Exportar un complemento

Puede exportar un complemento mediante la API de REST de Orchestrator.

### Procedimiento

- 1 En una aplicación cliente de REST, añada un encabezado de solicitud con los siguientes valores.
  - **Nombre:** `accept`
  - **Valor:** `application/dar`
- 2 Haga una solicitud GET en la dirección URL del complemento que desee exportar:

```
GET http://{host_orchestrator}:{puerto}/vco/api/plugins/{nombre_complemento}/
```

### Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. El contenido del complemento está disponible en el cuerpo de la respuesta.



## Habilitar o deshabilitar un complemento

Puede habilitar o deshabilitar un complemento mediante la API de REST de Orchestrator.

Puede cambiar el estado de un complemento de habilitado a deshabilitado o de deshabilitado a habilitado realizando una solicitud PUT en la URL del complemento. Puede comprobar el estado actual de un complemento recuperando información acerca de los complementos de Orchestrator. Consulte [Recuperar información sobre complementos](#).

### Requisitos previos

El contenido binario del complemento debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

### Procedimiento

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del complemento que desee habilitar o deshabilitar.
- 2 Haga una solicitud PUT en la dirección URL del complemento que desee habilitar o deshabilitar:

```
PUT http://{host_orchestrator}:{puerto}/vco/api/plugins/{nombre_complemento}/state/
```

### Resultados

Si la solicitud PUT es correcta, recibirá el código de estado 200.

## Operaciones de configuración de servidor

La API de REST de Orchestrator proporciona servicios web que se pueden utilizar para realizar varias operaciones de configuración del servidor de Orchestrator.

## Recuperar información sobre la configuración del servidor de Orchestrator

Puede recuperar información sobre la configuración del servidor de Orchestrator mediante la API de REST de Orchestrator.

### Procedimiento

- 1 En una aplicación cliente de REST, añada encabezados de solicitud para definir las propiedades del servidor para el que desea recuperar información.
- 2 Haga una solicitud GET en la dirección URL de los objetos del complemento:

```
GET http://{host_orchestrator}:{puerto}/vco/api/server-configuration/
```

## Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200.

## Importar la configuración del servidor de Orchestrator

Mediante la API de REST de Orchestrator, puede importar una configuración guardada.

### Requisitos previos

El contenido binario de la configuración debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

### Procedimiento

- 1 En una aplicación cliente de REST, añada un encabezado de solicitud con los siguientes valores.
  - **Nombre:** `content-type`
  - **Valor:** `multipart/form-data`
- 2 Haga una solicitud POST en la dirección URL de la configuración del servidor:

```
POST http://{host_orchestrator}:{puerto}/vco/api/server-configuration/
```

## Resultados

Si la solicitud POST es correcta, recibirá el código de estado 200.

## Exportar la configuración del servidor de Orchestrator

Puede exportar la configuración del servidor mediante la API de REST de Orchestrator.

### Requisitos previos

El contenido binario de la configuración debe estar disponible como contenido de varias partes. Para ver más detalles, consulte RFC 2387.

### Procedimiento

- 1 En una aplicación cliente de REST, añada un encabezado de solicitud con los siguientes valores.
  - **Nombre:** `content-type`
  - **Valor:** `multipart/form-data`
- 2 Añada otro encabezado de solicitud con los siguientes valores.
  - **Nombre:** `accept`
  - **Valor:** `*/*`

- 3 Haga una solicitud POST en la dirección URL de la configuración del servidor:

```
POST http://{host_orchestrator}:{puerto}/vco/api/server-configuration/
```

#### Resultados

Si la solicitud POST es correcta, recibirá el código de estado 200.

## Operaciones de etiquetado

La API de REST de Orchestrator proporciona servicios web que se pueden utilizar para realizar varias operaciones a fin de que los objetos sean más fáciles de buscar mediante etiquetas en Orchestrator.

Puede hacer que los objetos sean más fáciles de buscar si les adjunta etiquetas. Las etiquetas son cadenas con una longitud entre 3 y 64 caracteres; no deben contener espacios en blanco.

Puede añadir etiquetas globales y privadas. Las etiquetas globales están visibles para todos los usuarios de Orchestrator; las privadas, solo para los usuarios que las han creado. Los usuarios con privilegios administrativos son los únicos que pueden crear y eliminar etiquetas globales.

### Etiquetar un objeto

Puede asignar etiquetas a un objeto mediante la API de REST de Orchestrator.

Puede crear etiquetas privadas y globales. Especifique si la etiqueta es privada o global en el cuerpo de la solicitud.

---

**Nota** Para crear etiquetas globales, debe iniciar la sesión como usuario con privilegios administrativos.

---

También puede asignar un valor a la etiqueta que cree. El valor es un parámetro opcional que puede utilizar para filtrar etiquetas.

#### Procedimiento

- 1 Defina el cuerpo de la solicitud mediante la sintaxis siguiente.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<tag-instance xmlns="http://www.vmware.com/vco" global="false">
  <name>nombre_etiqueta</name>
  <value>valor_etiqueta</value>
</tag-instance>
```

---

**Nota** Puede crear una etiqueta global estableciendo la variable **global** en **"true"**.

---

## 2 Haga una solicitud POST en la dirección URL del objeto:

```
POST http://{host_orchestrator}:{puerto}/vco/api/catalog/{namespace}/{objectType}/
{objectId}/tags
```

### Resultados

Si la solicitud POST es correcta, recibirá el código de estado 200.

## Eliminar etiquetas de un objeto

Puede eliminar las etiqueta asignadas a un objeto mediante la API de REST de Orchestrator.

Es posible eliminar las etiquetas privadas y las globales.

---

**Nota** Para quitar etiquetas globales, debe iniciar la sesión como usuario con privilegios administrativos.

---

### Procedimiento

- ◆ Realice una solicitud DELETE para eliminar etiquetas privadas o globales.
  - Para eliminar una etiqueta privada, realice una solicitud DELETE en la URL del objeto mediante esta sintaxis:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/catalog/{namespace}/{objectType}/
{objectId}/tag/{tag_name}
```

- Para eliminar una etiqueta global, haga una solicitud DELETE en la URL del objeto mediante esta sintaxis:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/catalog/{namespace}/{objectType}/{objectId}/
tag/{tag_name}
```

### Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 200.

## Enumerar etiquetas de objeto

Se puede recuperar una lista de etiquetas asignadas a un objeto mediante el uso de la API de REST de Orchestrator.

### Procedimiento

- ◆ Haga una solicitud GET en la dirección URL del objeto:

```
GET http://{host_orchestrator}:{puerto}/vco/api/catalog/{espacio_nombres}/
{tipo_objeto}/{ID_objeto}/tags
```

## Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200.

## Enumerar objetos etiquetados por tipo

La API de REST de Orchestrator se puede utilizar para recuperar una lista de objetos etiquetados con una etiqueta específica y para filtrarlos por tipo de objeto.

### Procedimiento

- ◆ Haga una solicitud GET en la dirección URL del tipo de objeto:

```
GET http://{host_orchestrator}:{puerto}/vco/api/catalog/{espacio_nombres}/  
{tipo_objeto}?tags=etiqueta1&tags=:etiqueta2=value
```

## Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200.

## Enumerar propietarios de etiquetas

Se puede recuperar una lista de propietarios de etiquetas mediante el uso de la API de REST de Orchestrator. Los propietarios de etiquetas son usuarios que han creado al menos una etiqueta.

### Procedimiento

- ◆ Haga una solicitud GET en la siguiente dirección URL:

```
GET http://{host_orchestrator}:{puerto}/vco/api/tags
```

## Resultados

Si la solicitud GET es correcta, recibirá el código de estado 200. La lista que se recupera contiene usuarios que crearon al menos una etiqueta. Las etiquetas globales se enumeran bajo el nombre de usuario del sistema \_\_GLOBAL\_\_.

## Enumerar etiquetas por usuarios

La API de REST de Orchestrator se puede utilizar para recuperar una lista de etiquetas creadas por un usuario específico.

También se pueden recuperar etiquetas globales. Las etiquetas globales se enumeran bajo el nombre de usuario del sistema \_\_GLOBAL\_\_.

**Procedimiento**

- ◆ Haga una solicitud GET en la dirección URL del usuario.
  - Para recuperar una lista de las etiquetas creadas por un usuario específico, haga una solicitud GET con la siguiente sintaxis:

```
GET http://{host_orchestrator}:{puerto}/vco/api/tags/{nombre_usuario}
```

- Para recuperar una lista de etiquetas globales, haga una solicitud GET con la siguiente sintaxis:

```
GET http://{host_orchestrator}:{puerto}/vco/api/tags/__GLOBAL__
```

**Resultados**

Si la solicitud GET es correcta, recibirá el código de estado 200.

**Enumerar etiquetas por usuarios filtradas por nombre de etiqueta**

La API de REST de Orchestrator se puede utilizar para recuperar una lista de instancias de etiquetas creadas por un usuario específico y para filtrar las etiquetas por nombre de etiqueta.

También se pueden recuperar instancias de etiquetas globales. Las etiquetas globales se enumeran bajo el nombre de usuario del sistema \_\_GLOBAL\_\_.

**Procedimiento**

- ◆ Haga una solicitud GET en la dirección URL del usuario.
  - Para recuperar una lista filtrada de las instancias de etiquetas creadas por un usuario específico, haga una solicitud GET con la siguiente sintaxis:

```
GET http://{host_orchestrator}:{puerto}/vco/api/tags/{nombre_usuario}/{nombre_etiqueta}
```

- Para recuperar una lista filtrada de instancias de etiquetas globales, haga una solicitud GET con la siguiente sintaxis:

```
GET http://{host_orchestrator}:{puerto}/vco/api/tags/__GLOBAL__/{nombre_etiqueta}
```

**Resultados**

Si la solicitud GET es correcta, recibirá el código de estado 200. La información que se recupera contiene una referencia al objeto etiquetado, nombre de etiqueta, valor de etiqueta y una indicación de si la instancia de etiqueta es global o privada.

**Quitar etiquetas por usuarios**

La API de REST de Orchestrator se puede utilizar para quitar todas las etiquetas creadas por un usuario específico.

También se pueden quitar etiquetas globales. Las etiquetas globales se enumeran bajo el nombre de usuario del sistema \_\_GLOBAL\_\_.

---

**Nota** Para quitar etiquetas globales, debe iniciar la sesión como usuario con privilegios administrativos.

---

#### Procedimiento

- ◆ Haga una solicitud DELETE en la dirección URL del usuario.
  - Para quitar las etiquetas creadas por un usuario específico, haga una solicitud DELETE con la siguiente sintaxis:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/tags/{nombre_usuario}
```

- Para quitar las etiquetas globales, haga una solicitud DELETE con la siguiente sintaxis:

```
DELETE http://{host_orchestrator}:{puerto}/vco/api/tags/__GLOBAL__
```

#### Resultados

Si la solicitud DELETE es correcta, recibirá el código de estado 204.