

# Développement d'un client de services Web pour VMware vRealize Orchestrator

vRealize Orchestrator 7.4



vmware®

Vous trouverez la documentation technique la plus récente sur le site Web de VMware à l'adresse :

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

N'hésitez pas à nous transmettre tous vos commentaires concernant cette documentation à l'adresse suivante :

[docfeedback@vmware.com](mailto:docfeedback@vmware.com)

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

**VMware, Inc.**  
100-101 Quartier Boieldieu  
92042 Paris La Défense  
France  
[www.vmware.com/fr](http://www.vmware.com/fr)

Copyright © 2008–2018 VMware, Inc. Tous droits réservés. [Copyright et informations sur les marques.](#)

# Table des matières

Développement d'un client de services Web pour VMware vRealize Orchestrator 5

## 1 Développement d'un client de services Web 6

## 2 Utilisation de l'API REST vRealize Orchestrator 7

Authentification auprès d'Orchestrator et de systèmes tiers 8

Utiliser l'authentification vCenter Single Sign-On avec l'API REST d'Orchestrator 8

Accès à la documentation de référence de l'API REST d'Orchestrator 14

Utilisation du SDK REST Java 14

Opérations avec des workflows 15

Rechercher un workflow et récupérer sa définition 15

Exécuter un workflow 18

Exécuter un workflow après avoir validé ses paramètres d'entrée en fonction de sa présentation 20

Interagir avec un workflow au cours de son exécution 24

Récupérer les interactions d'un workflow 32

Accéder à un schéma de workflow 32

Utilisation des tâches 33

Créer une tâche 33

Modifier une tâche 34

Vérifier l'état d'une tâche 35

Rechercher des objets dans l'inventaire Orchestrator 36

Rechercher des objets par type et par ID 36

Rechercher des objets par relation 37

Appliquer des filtres 38

Importer et exporter des objets Orchestrator 39

Importer un workflow 39

Exporter un workflow 40

Importer une action 40

Exporter une action 40

Importer un module 41

Exporter un module 42

Importer une ressource 43

Exporter une ressource 43

Importer un élément de configuration 44

Exporter un élément de configuration 44

Supprimer des objets Orchestrator 45

Supprimer un workflow 45

Supprimer une action	45
Supprimer un module	45
Supprimer une ressource	46
Supprimer un élément de configuration	46
Définir des autorisations sur les objets Orchestrator	47
Autorisations de l'API REST	47
Récupérer les autorisations d'un workflow	48
Supprimer les autorisations d'un workflow	48
Définir les autorisations d'un workflow	48
Récupérer les autorisations d'une action	49
Supprimer les autorisations d'une action	49
Définir les autorisations d'une action	50
Récupérer les autorisations d'un module	50
Supprimer les autorisations d'un module	51
Définir les autorisations d'un module	51
Récupérer les autorisations d'une ressource	52
Supprimer les autorisations d'une ressource	52
Définir les autorisations d'une ressource	52
Récupérer les autorisations d'un élément de configuration	53
Supprimer les autorisations d'un élément de configuration	53
Définir les autorisations d'un élément de configuration	54
Effectuer des opérations avec des plug-ins	54
Récupérer des informations à propos des plug-ins	54
Importer un plug-in	55
Exporter un plug-in	55
Activer ou désactiver un plug-in	56
Opérations de configuration du serveur	56
Récupérer des informations sur la configuration du serveur Orchestrator	56
Importer la configuration du serveur Orchestrator	57
Exporter la configuration du serveur Orchestrator	57
Réaliser des opérations de balisage	58
Baliser un objet	58
Supprimer les balises d'un objet	58
Répertorier les balises d'un objet	59
Répertorier les objets balisés par type	59
Répertorier les propriétaires des balises	60
Répertorier les balises en fonction des utilisateurs	60
Répertorier les balises en fonction des utilisateurs filtrés par nom de balise	60
Supprimer les balises en fonction des utilisateurs	61

# Développement d'un client de services Web pour VMware vRealize Orchestrator

La rubrique *Développement d'un client de services Web pour VMware vRealize Orchestrator* fournit des informations sur le développement d'un client de services Web pour VMware<sup>®</sup> vRealize Orchestrator.

Orchestrator fournit une API de services Web afin que vous puissiez développer des applications pour ouvrir et utiliser les workflows via Internet. Orchestrator fournit une API REST (Representational State Transfer) que vous pouvez utiliser pour exécuter différentes opérations relatives aux workflows.

## Public visé

Ces informations s'adressent aux développeurs d'applications Web qui souhaitent accéder aux processus Orchestrator d'un réseau via le service Web RESTful.

# Développement d'un client de services Web

1

VMware vRealize Orchestrator fournit des API de services Web afin que vous puissiez développer des applications pour accéder aux workflows via Internet. L'objectif principal des API de services Web Orchestrator est de vous permettre d'intégrer les workflows Orchestrator dans des applications Web personnalisées.

Orchestrator propose une API de services Web basée sur une API REST (Representational State Transfer). L'API REST d'Orchestrator expose les objets de l'inventaire Orchestrator et les inventaires des plug-ins installés en tant que ressources auxquelles vous pouvez accéder dans des URL prédéfinies. Les demandes HTTP auprès de ces URL déclenchent des opérations sur les workflows. L'API REST d'Orchestrator expose les objets d'inventaire en tant que ressources via un ensemble de services Web RESTful. Vous pouvez utiliser ces services pour récupérer les définitions des workflows, exécuter des workflows, vérifier l'état des workflows en cours d'exécution, annuler des exécutions de workflows, traiter des interactions utilisateurs en attente, récupérer la présentation des workflows, etc.

# Utilisation de l'API REST vRealize Orchestrator

# 2

L'API REST d'Orchestrator confère des fonctionnalités qui vous permettent de communiquer directement avec le serveur Orchestrator via HTTP et d'exécuter différentes opérations relatives aux workflows.

L'API REST d'Orchestrator expose les objets des inventaires du serveur Orchestrator et les plug-ins installés en tant que ressources dans des URL prédéfinies. Vous envoyez des appels HTTP au niveau de ces URL pour déclencher des opérations dans Orchestrator. De cette manière, vous pouvez effectuer différentes tâches de workflows :

- Exécuter un workflow, planifier un workflow, récupérer les exécutions d'un workflow, répondre à une interaction utilisateur et annuler l'exécution d'un workflow.
- Récupérer les détails d'un workflow comme, par exemple, ses paramètres d'entrée et de sortie et sa présentation.
- Récupérer les détails d'une exécution de workflow comme, par exemple, son état, les journaux générés, sa date de début et sa date de fin.
- Parcourir les inventaires d'Orchestrator et des plug-ins installés.
- Importer et exporter des workflows, des actions et des modules.

L'utilisation de l'API REST d'Orchestrator vous permet d'intégrer facilement les workflows Orchestrator dans des applications personnalisées que vous pouvez créer avec n'importe quel langage de programmation.

L'API REST d'Orchestrator fournit également un support e-Tag et un mécanisme de mise en cache des données de réponse.

Ce chapitre aborde les rubriques suivantes :

- [Authentification auprès d'Orchestrator et de systèmes tiers](#)
- [Accès à la documentation de référence de l'API REST d'Orchestrator](#)
- [Utilisation du SDK REST Java](#)
- [Opérations avec des workflows](#)
- [Utilisation des tâches](#)
- [Rechercher des objets dans l'inventaire Orchestrator](#)
- [Importer et exporter des objets Orchestrator](#)

- [Supprimer des objets Orchestrator](#)
- [Définir des autorisations sur les objets Orchestrator](#)
- [Effectuer des opérations avec des plug-ins](#)
- [Opérations de configuration du serveur](#)
- [Réaliser des opérations de balisage](#)

## Authentification auprès d'Orchestrator et de systèmes tiers

Vous devez fournir votre authentification Orchestrator dans les demandes HTTP que vous soumettez via l'API REST d'Orchestrator. Si vous utilisez l'API REST d'Orchestrator pour accéder aux ressources d'un système tiers tel que vCenter Server ou vRealize Automation, vous devez également vous authentifier auprès de ce système.

Par exemple, vous devez vous authentifier auprès d'Orchestrator pour accéder à tous les workflows de l'inventaire Orchestrator. Cependant, l'exécution d'un workflow par rapport à vCenter Server requiert une authentification auprès d'Orchestrator et de vCenter Server.

Selon que vous configurez Orchestrator avec vRealize Automation ou vSphere en tant que fournisseur d'authentification, le schéma d'authentification pour l'API REST d'Orchestrator est différent. Si Orchestrator utilise vCenter Single Sign-On, vous pouvez, selon votre configuration, vous authentifier à l'aide d'un jeton de détenteur de clé émis par un serveur vCenter Single Sign-On. Si Orchestrator est configuré avec vRealize Automation, vous pouvez utiliser l'authentification via un jeton d'accès du porteur OAuth.

Si vous soumettez des demandes HTTP au niveau de l'URL supérieure de l'API REST d'Orchestrator, il n'est pas nécessaire de vous authentifier auprès d'Orchestrator. L'URL supérieure de l'API REST d'Orchestrator est `https://orchestrator_host:port/vco/api/`.

---

**Remarque** Le numéro de port par défaut pour le serveur Orchestrator externe est 8281. Le numéro de port par défaut pour l'instance d'Orchestrator qui est intégrée dans vRealize Automation est 443.

---

Une demande GET au niveau de l'URL supérieure de l'API REST renvoie les URL dirigeant vers l'ensemble des ressources accessibles via l'API. Vous devez vous authentifier auprès d'Orchestrator pour soumettre des demandes HTTP au niveau de ces URL.

## Utiliser l'authentification vCenter Single Sign-On avec l'API REST d'Orchestrator

Si Orchestrator est configuré avec le serveur vCenter Single Sign-On à l'aide du mode d'authentification vSphere, vous avez besoin d'un jeton de détenteur de clé principal pour accéder aux objets du système Orchestrator via l'API REST d'Orchestrator. Si vous souhaitez accéder à vCenter Server ou à des systèmes tiers utilisant le serveur vCenter Single Sign-On via le serveur Orchestrator, vous avez besoin d'un jeton de détenteur de clé délégué pour Orchestrator et de votre jeton principal.

Si Orchestrator est configuré avec le serveur vCenter Single Sign-On, vous devez vous authentifier à l'aide d'informations d'identification valides. Orchestrator gère le jeton de détenteur de clé.

## Accéder à des objets système dans Orchestrator

Vous pouvez accéder à des objets système d'Orchestrator au niveau des URL des services Inventaire et Catalogue de l'API REST.

- `https://orchestrator_host:port/vco/api/inventory/System/`
- `https://orchestrator_host:port/vco/api/catalog/System/`

Pour accéder à des objets système d'Orchestrator, vous devez transposer votre jeton de détenteur de clé principal dans l'en-tête `Authorization` des demandes HTTP que vous envoyez au service Inventaire ou Catalogue.

Par exemple, pour récupérer tous les objets système de type `Workflow`, vous soumettez une demande GET au niveau de `https://orchestrator_host:port/vco/api/catalog/System/Workflow/`. Pour vous authentifier auprès d'Orchestrator, vous devez transposer votre jeton de détenteur de clé principal dans l'en-tête `Authorization` de la demande.

## Accéder à des objets dans des systèmes tiers

Pour réaliser des opérations dans des systèmes tiers enregistrés auprès du serveur vCenter Single Sign-On via l'API REST d'Orchestrator, vous devez vous authentifier auprès d'Orchestrator et du système tiers. Vous devez insérer deux en-têtes dans les appels HTTP que vous envoyez via l'API REST d'Orchestrator.

- `Authorization`. Vous devez transposer votre jeton de détenteur de clé principal dans cet en-tête.
- `VCOAuthorization`. Vous devez transposer un jeton de détenteur de clé délégué pour Orchestrator dans cet en-tête. Vous devez obtenir le jeton délégué pour Orchestrator auprès du serveur vCenter Single Sign-On. Orchestrator utilise le jeton délégué pour s'authentifier en votre nom auprès du système tiers.

Par exemple, pour exécuter un workflow qui utilise une machine virtuelle via l'API REST d'Orchestrator, vous accédez aux ressources dans Orchestrator et dans vCenter Server. Pour vous authentifier auprès d'Orchestrator et de vCenter Server, vous devez transposer votre jeton de détenteur de clé principal dans l'en-tête `Authorization` de la demande que vous soumettez, et le jeton délégué dans l'en-tête `VCOAuthorization`. De cette façon, vous vous authifiez auprès d'Orchestrator avec votre jeton principal et Orchestrator s'authentifie en votre nom auprès de vCenter Server avec le jeton délégué.

Le serveur vCenter Single Sign-On considère Orchestrator comme une solution, et chaque solution est enregistrée sous un nom d'utilisateur unique auprès du serveur vCenter Single Sign-On. Vous pouvez demander un jeton délégué pour Orchestrator en transmettant le nom d'utilisateur de la solution Orchestrator et un jeton de détenteur de clé principal au serveur vCenter Single Sign-On. Le jeton émis par le serveur vCenter Single Sign-On est un jeton de détenteur de clé délégué pour qu'Orchestrator puisse s'authentifier en votre nom auprès des systèmes tiers.

## Exemple : Obtenir une session en mode vCenter Single Sign-On

L'exemple de code suivant permet d'obtenir une session en mode 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(URI ssoUri, URI adminUri, VcoSessionFactory
vcoSessionFactory, long lifeTimeSeconds);

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

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

## Obtenir le nom d'utilisateur de la solution Orchestrator

Le serveur vCenter Single Sign-On considère Orchestrator comme une solution, et chaque solution est enregistrée sous un nom d'utilisateur unique auprès du serveur vCenter Single Sign-On. Pour pouvoir demander un jeton de détenteur délégué pour Orchestrator auprès du serveur vCenter Single Sign-On, vous devez connaître le nom d'utilisateur de la solution Orchestrator.

### Prérequis

Vérifiez que vous disposez d'un jeton de détenteur principal valide émis par le serveur vCenter Single Sign-On.

### Procédure

- 1 Soumettez une demande GET au niveau de l'URL du nom d'utilisateur de la solution Orchestrator :

```
GET https://{orchestrator_host}:{port}/vco/api/users/
```

- 2 Indiquez votre jeton de détenteur principal dans l'en-tête Authorization de la demande.

L'élément `<user solution-user="OrchestratorSolutionUserName"/>` de la réponse contient le nom d'utilisateur de la solution Orchestrator. L'exemple suivant est un nom d'utilisateur de la solution Orchestrator.

```
<user xmlns="http://www.vmware.com/vco" admin-rights="true" solution-
user="vCO-15d98795afa5b0d6f47ee3aeab3">
```

## Suivant

Utilisez le nom d'utilisateur de la solution Orchestrator et votre jeton de détenteur principal pour demander un jeton de détenteur délégué auprès du serveur vCenter Single Sign-On.

## Utilisation du SDK de l'API REST vRealize Orchestrator avec l'authentification vRealize Automation configurée

Vous pouvez utiliser le SDK de l'API REST avec une authentification vRA configurée dans un environnement à locataire unique ou à locataires multiples.

Pour obtenir un jeton d'authentification (OAuth2.0) requis pour le code ci-dessous, consultez l'article KB [Autorisation de l'API REST de vRO avec l'authentification OAuth2.0 \(2148518\)](#).

---

**Remarque** Obtenir une session dans le mode d'authentification de vRealize Automation

---

L'exemple de code suivant permet d'obtenir une session dans le mode d'authentification de vRealize Automation (dans un environnement à locataire unique et à locataires multiples).

- Si la mutualisation n'est pas activée :

```
URI uri = URI.create("https://orchestrator-server:8283/vco/api");
VcoSessionFactory sessionFactory = new DefaultVcoSessionFactory(uri);
String token =
"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJqdGkiOiJjNzg4NWNiYS1hZTFmLTRiM2UtYmYyYi04ZmRmNzY3N"
+
"GZiZWElLCJwcm4iOiJhZG1pbmZldHJhdG9yQFZTUeHFUKUUE9DQUwILCJkb21haW4iOiJ2c3BoZXJlLmxvY2FsIiwidXNlc19pZCI6Ij"
+
"MiLCJhdXRoX3RpbWUiOiJlMDIyMDIxMTAsImZcyI6Imh0dHBzOi8vc2YtMjktMTAtMjkuc29mLW1idS5lbmcudm13YXJlLnN"
+
"vbS9TUFTL3QvdnNwaGVyZS5sb2Nhbnh0dXRoIiwiaXVkiOiJoaHR0cHM6Ly9zZi0yOS0xMC0yOS5zb2YtbWJlLmVuZy52bXdhc"
+
"mUuY29tL"
+
"1NBQVMvdC92c3BoZXJlLmxvY2FsL2F1dGgvd2F1dGh0b2t1biIsImN0eCI6Ilt7XCJtdGRcIjpcInVybjpvYXNpczpuYW1lc"
+
"z0YzptQU1M0jIuMDphYzpjbgGFzc2VzO1Bhc3N3b3JkUHJvdGVjdGVkVHJhbnNwb3J0XCIsXCJpYXRcIjoxNTAyMjAyMTEwLW1lc"
+
"n1dIiwic2NwIjoidXNlc1IsImk6I6IjAilCJlbnw0iJhZG1pbmZldHJhdG9yQHNmLTl5LTUwLTI5LnVnZi1tYnUuZW5nLnZ"
+
"td2FyZS5jb20iLCJjaWQiOiJjYWZlX2NsaS1yRlJlNEExbURnIiwiaXVkiOiJoaHR0cHM6Ly9zZi0yOS0xMC0yOS5zb2YtbWJlLmVuZy52bXdhc"
+
"mUuY29tL"
+
"E1MDIyMDIxMTAsInN1YiI6IjQ1ZjQwNWUzLTNlNTgtNGJmZC1hNzMwLTQ1MjU4OWIxOGUxNyIsInByb190eXB1IjoiVVFuIj9"
+
".G9gEQPtMEH5jYab-ILTK8NFYcwc3JZCEEjsmpUSH6oxLmZKEf-1JbsysBVH4ufqmGah3GMvmy6PUiTTamLRLFKLwa500QCQ-agKutZ121R"
+
"m740qBKLHmBB0NQg19ysMAVJNSxapFzirmWurF_5CKpv4WM7Y8H_bY9iNmDKQTXI";

//provide OAuth2 token obtained in step 1 here
```

```
Authentication auth = new OAuthTokenAuthentication(token);

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

- Si la mutualisation est activée :
  - Pour les utilisateurs de locataires réguliers :

[illegible]

- Pour les utilisateurs de solution :

Les utilisateurs de solution peuvent travailler au niveau de leur propre locataire et au niveau des locataires réguliers. Ils peuvent remplacer le niveau des opérations qu'ils réalisent.

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

exapmle

String token =
"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJqdGkiOiJNzg4NWNiYS1hZTFmLTRiM2UtYmYyYi04ZmRmNzY3N"
+
"GZiZWEiLCJwcm4iOiJhZG1pbmlzdHJhdG9yQFZTUehFUKUuTE9DQUwiLCJkb21haW4iOiJ2c3BoZXJlLmxvY2FsIiwidXNlcl9pZCI6Ij"
+
"MiLCJhdXRoX3RpbWUiOiJlMDIyMDIxMTAsImZcyI6Imh0dHBzOi8vc2YtMjktMTAtMjkuc29mLWlidS5lbmcudm13YXJlLmN"
+
"vbS9TQUFTL3QvdnNwaGVyZS5sb2Nhbc9hdXRoIiwiaXVkiOiJoaHR0cHM6Ly9zZi0yOS0xMC0yOS5zb2YtbWJlLmVuZy52bXdhcmUuY29tL"
+
"1NBQVMvdC92c3BoZXJlLmxvY2FsL2F1dGgvd2F1dGh0b2t1biIsImN0eCI6Ilt7XCJtdGRcIjpcInVybjpvYXNpczpuYW1lc2p"
+
"0YzptQU1MOjIuMDphYzpjbgFzc2VzO1Bhc3N3b3JkUHJvdGVjdGVkVHJhbnNwb3J0XCIsXCJpYXRcIjoxNTAyMjAyMTEwLWlrc2p"
+
"n1dIiwic2NwIjoidXNlciIsIm1kcCI6IjAiLCJlbWwiOiJhZG1pbmlzdHJhdG9yQHNmLTl5LTUwLTI5LnNvZi1tYnUuZW5nLnZ"
+
"td2FyZS5jb20iLCJjaWQiOiJjYWZlX2NsaS1yRlJlNE56bURnIiwiaXVkiOiJoaHR0cHM6Ly9zZi0yOS0xMC0yOS5zb2YtbWJlLmVuZy52bXdhcmUuY29tL"
+
"E1MDIyMDIxMTAsInN1YiI6IjQ1ZjQwNWUzLTNlNTgtNGJmZC1hNzMwLTQ1MjU4OWIxOGUxNyIsInByb190eXB1IjoiVVNFUiJ9."
+ "G9gEQPtmeH5jYab-
ILT8NFYcwc3JZCEEjsmpUSH6oxLmZKEf-1JbsysBVH4ufqmGah3GMvmy6PUiTTamLRLfkCLwa500QCQ-qgKutZl21R"
+ "m740qBKLhmBB0NQg19ysMAVJNSxapFzirmWurF_5CKpv4WM7Y8H_bY9iNmDKQTXI";

//provide OAuth2 token obtained in step 1 here
Authentication auth = new OAuthTokenAuthentication(token);

// By default each tenant works in its tenant scope. However, solution users can overrde the
tenant in which they perform a given operation:
// Here, users of SDK should provide a value that is meaningful to their context.
String overrideWithTenant = "nonSolutionUserTenant";

VcoSession session = sessionFactory.newSession(auth, overrideWithTenant);

//Use session below
```

## Accès à la documentation de référence de l'API REST d'Orchestrator

La documentation de référence de l'API REST d'Orchestrator comporte des informations sur les services Web RESTful de l'API, le modèle de données applicable à l'API, les codes de réponse valides pour l'API, des exemples de code, etc.

La documentation de référence de l'API REST Orchestrator est installée en même temps qu'Orchestrator. La documentation de référence est disponible à l'adresse [https://orchestrator\\_host:port/vco/api/docs/](https://orchestrator_host:port/vco/api/docs/).

La spécification Swagger officielle est disponible à l'adresse : <https://swagger.io/specification/>.

## Utilisation du SDK REST Java

Vous pouvez utiliser une bibliothèque SDK Java pour appeler des opérations sur l'API REST d'Orchestrator dans les applications Java et travailler directement avec les objets.

Tous les services Web RESTful du SDK REST d'Orchestrator possèdent une catégorie d'encapsulation Java avec des méthodes qui correspondent aux opérations qui peuvent être exécutées par le biais de ce service.

Le SDK REST Java est installé en même temps qu'Orchestrator. Les artefacts du SDK REST Java sont disponibles aux emplacements suivants.

---

**Remarque** Vous devez avoir déployé le dispositif Orchestrator pour accéder aux artefacts.

---

- [https://orchestrator\\_host:port/vco-repo/com/vmware/o11n/o11n-rest-client/](https://orchestrator_host:port/vco-repo/com/vmware/o11n/o11n-rest-client/)
- [https://orchestrator\\_host:port/vco-repo/com/vmware/o11n/o11n-rest-client-examples/](https://orchestrator_host:port/vco-repo/com/vmware/o11n/o11n-rest-client-examples/)
- [https://orchestrator\\_host:port/vco-repo/com/vmware/o11n/o11n-rest-client-services/](https://orchestrator_host:port/vco-repo/com/vmware/o11n/o11n-rest-client-services/)
- [https://orchestrator\\_host:port/vco-repo/com/vmware/o11n/o11n-rest-client-stubs/](https://orchestrator_host:port/vco-repo/com/vmware/o11n/o11n-rest-client-stubs/)

## Exemple : Exécuter un workflow et attendre son achèvement

L'exemple de code suivant exécute un workflow et attend la fin de son exécution.

```
//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);

```

## Opérations avec des workflows

L'API REST d'Orchestrator propose des services Web que vous pouvez utiliser pour effectuer différentes opérations avec des workflows.

### Rechercher un workflow et récupérer sa définition

Si vous souhaitez pouvoir effectuer toutes les opérations avec un workflow, vous devez trouver ce workflow dans l'inventaire Orchestrator et récupérer sa définition. La définition répertorie les paramètres d'entrée et de sortie du workflow, et comporte des liens vers les exécutions de workflow disponibles, la présentation du workflow et les autres objets.

#### Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

#### Procédure

##### 1 Recherchez l'élément d'inventaire du workflow.

- Si vous disposez du nom complet du workflow ou d'un mot-clé provenant de ce nom, soumettez une demande GET au niveau de l'URL du service de workflow en appliquant un filtre :

```

GET https://{orchestrator_host}:{port}/vco/api/workflows?conditions=name={workflowFullName}

GET https://{orchestrator_host}:{port}/vco/api/workflows?conditions=name~{keyWord}

```

- Recherchez le workflow via le service Catalogue ou Inventaire en soumettant une demande GET au niveau de l'URL servant de point d'entrée aux éléments d'inventaire du workflow :

```

GET https://{orchestrator_host}:{port}/vco/api/catalog/System/Workflow/

GET https://{orchestrator_host}:{port}/vco/api/inventory/System/Workflows/

```

- 2 Récupérez l'élément d'inventaire du workflow en soumettant une demande GET au niveau de son URL :

```
GET https://{orchestrator_host}:{port}/vco/api/catalog/System/Workflow/{workflowID}/
```

- 3 Récupérez la définition du workflow en soumettant une demande GET au niveau de l'URL de la définition :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

## Exemple : Rechercher le workflow Send Hello

Vous pouvez rechercher le workflow Send Hello et récupérer sa définition :

- 1 Pour rechercher le workflow Send Hello, soumettez une demande GET au niveau de l'URL du service Workflow en appliquant un filtre :

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

Vous obtenez une liste des workflows dont le nom contient le terme Hello :

```
<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/CF808080808080808080808080808080E6808080013086668236014a0614d1
6e1/">
    <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/CF808080808080808080808080808080DA808080013086668236014a0614d1
6e1/">
    <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"/>
    </attributes>
  </link>
</inventory-items>
</xml>
```

```

        <attribute name="canEdit" value="true"/>
    </attributes>
</link>
</inventory-items>

```

- 2 Soumettez une demande GET au niveau de l'URL de l'élément d'inventaire du workflow Send Hello :

```
GET https://localhost:  
8281/vco/api/catalog/System/Workflow/CF808080808080808080808080DA808080013086668236014a0614d1  
6e1/
```

Vous obtenez l'élément d'inventaire du workflow Send Hello dans la réponse :

```
<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/CF8080808080808080808080DA808080013086668236014a0614d1
6e1/">
    <relations>
        <link rel="down"
            href="https://localhost:
8281/vco/api/workflows/CF80808080808080808080808080DA808080013086668236014a0614d16e1/" />
    </relations>
    <attributes>
        <attribute name="id"
value="CF8080808080808080808080808080DA808080013086668236014a0614d16e1"/>
        <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 Pour récupérer la définition du workflow, soumettez une demande GET au niveau de son URL :

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

Vous obtenez la définition du workflow Send Hello dans la réponse :

```
<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/CF8080808080808080808080808080DA808080013086668236014a0614d16e1/">
    <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/CF8080808080808080808080808080DA808080013086668236014a0614d16e1/execution
s/" />
        <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>

```

## Exécuter un workflow

Vous exécutez un workflow via l'API REST d'Orchestrator en créant un nouvel objet d'exécution pour un workflow spécifique.

## Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

## Procédure

- 1 Récupérez la définition du workflow que vous souhaitez exécuter en soumettant une demande GET au niveau de l'URL de la définition :

```
GET http://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

Vous obtenez la définition du workflow dans la réponse à la demande. La définition du workflow vous permet de consulter les paramètres d'entrée et la description du workflow, ainsi que d'autres informations.

- 2 Soumettez une demande POST au niveau de l'URL qui comporte les objets d'exécution du workflow :

POST [https://{orchestrator\\_host}:{port}/vco/api/workflows/{workflowID}/executions/](https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/)

- 3 Veillez à indiquer les valeurs des paramètres d'entrée du workflow dans un élément execution-context de la demande.

Si vous fournissez un `execution-context` vide dans la demande, le workflow s'exécute avec les valeurs par défaut de ses paramètres d'entrée, le cas échéant.

Si la demande POST aboutit, vous obtenez le code d'état 202 avec une réponse vide et un lien vers le nouvel objet d'exécution dans l'en-tête Location.

### Exemple : Exécuter le workflow Send Hello

Vous pouvez récupérer la définition du workflow Send Hello et l'exécuter.

- 1 Soumettez une demande GET au niveau de l'URL qui comporte la définition du workflow Send Hello :

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

Vous obtenez la définition du workflow dans la réponse à la demande :

```
<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/CF8080808080808080808080808080DA808080013086668236014a0614d16e1/">
    <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/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/presentations/" />
        <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 Soumettez une demande POST au niveau de l'URL qui comporte les objets d'exécution du workflow :

```
POST https://localhost:8281/vco/api/workflows/CF80808080808080808080808080DA808080013086668236014a0614d16e1/executions/
```

Transposez les valeurs des paramètres d'entrée dans un élément execution-context dans la demande :

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

## Exécuter un workflow après avoir validé ses paramètres d'entrée en fonction de sa présentation

La présentation d'un workflow peut définir des contraintes concernant les valeurs que vous pouvez transmettre aux paramètres d'entrée. Il peut par exemple s'agir d'une liste ou d'une gamme de valeurs. Pour assurer la réussite de l'exécution d'un workflow, vous devez valider les valeurs que vous transmettez aux paramètres d'entrée de ce workflow en vous référant à la définition de sa présentation.

Lorsque vous intégrez des workflows dans des applications personnalisées, il est parfois nécessaire de créer un assistant dans lequel vous saisissez les valeurs des paramètres d'entrée du workflow pendant son exécution. Grâce au service Présentation du workflow, vous pouvez instancier la présentation d'un workflow et transmettre les valeurs de ses paramètres d'entrée dans les parties qui correspondent aux différents écrans de l'assistant. Vous pouvez valider les valeurs transmises aux paramètres d'entrée en fonction des contraintes définies dans la présentation du workflow.

## Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

## Procédure

- 1 Récupérez la définition du workflow que vous souhaitez exécuter en soumettant une demande GET au niveau de l'URL de la définition de ce workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

Vous obtenez la définition du workflow dans la réponse à la demande. La définition du workflow vous permet de consulter les paramètres d'entrée et la définition du workflow, ainsi que d'autres informations.

- 2 Récupérez la définition de la présentation du workflow en soumettant une demande GET au niveau de l'URL :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/presentation/
```

- 3 Dans la réponse à la demande, consultez la définition de la présentation du workflow pour vérifier les contraintes concernant les valeurs que vous pouvez transmettre aux paramètres d'entrée.

Par exemple, un paramètre d'entrée peut présenter une liste prédéfinie de valeurs à sélectionner.

- 4 Instanciez la présentation du workflow en soumettant une demande POST au niveau de l'URL des instances de présentation :

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/presentation/instances/
```

- 5 Indiquez un élément `execution-context` dans le corps de la demande pour instancier la présentation.

Vous pouvez transposer un `execution-context` vide ou un `execution-context` avec les valeurs de certains paramètres d'entrée uniquement.

- 6 Pour transposer des valeurs aux paramètres d'entrée en plusieurs parties, soumettez autant de demandes POST ou PUT que vous le souhaitez au niveau de l'URL qui comporte l'instance de présentation :

```
PUT https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/presentation/instances/{executionID}/
```

- 7 Vérifiez la réponse à la demande POST ou PUT que vous avez soumise.

Si les valeurs que vous avez transmises dans les paramètres d'entrée sont valides, vous obtenez un attribut `valid="true"` dans la balise `execution`. Si la présentation est valide, vous pouvez prendre les valeurs qui sont répertoriées dans l'élément `out-parameters` de la réponse et les transmettre en tant que valeurs aux paramètres d'entrée lorsque vous exécutez le workflow.

- 8 Si les valeurs des paramètres d'entrée sont valides, exécutez le workflow en soumettant une demande POST au niveau de l'URL qui comporte les exécutions du workflow :

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/
```

- 9 Fournissez les valeurs valides aux paramètres d'entrée du workflow dans un élément execution-context.

### Exemple : Exécuter le workflow Send Hello en validant ses paramètres d'entrée

Vous pouvez exécuter le workflow Send Hello en validant ses paramètres d'entrée en fonction des définitions de sa présentation.

- 1 Soumettez une demande GET au niveau de l'URL qui comporte la définition du workflow Send Hello :

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

Vous obtenez la définition du workflow dans la réponse à la demande :

```
<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/CF8080808080808080808080808080DA808080013086668236014a0614d16e1/">
  <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/CF8080808080808080808080808080DA808080013086668236014a0614d16e1/execution
s/" />
    <link rel="down"
      href="https://localhost:
8281/vco/api/workflows/CF8080808080808080808080808080DA808080013086668236014a0614d16e1/execution
s/" />
    <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 Soumettez une demande GET au niveau de l'URL qui comporte la définition de la présentation du workflow :

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

- 3 Soumettez une demande POST au niveau de l'URL qui comporte les instances d'exécution de la présentation du workflow :

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

Indiquez un execution-context vide afin d'instancier la présentation sans fournir aucune valeur pour les paramètres d'entrée :

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

La réponse contient des messages d'erreur associés à chaque champ indiquant que les valeurs des paramètres d'entrée ne sont pas valides.

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

- 4 Soumettez une demande POST au niveau de l'URL qui comporte l'instance spécifique de la présentation :

POST https://localhost:  
8281/vco/api/workflows/CF808080808080808080808080DA808080013086668236014a0614d16e1/presentation/instances/8880808080808080808080808080803F8080800132145338690643f66a027ec/

Dans le corps de la demande, indiquez les valeurs des paramètres d'entrée :

```
<?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>
```

Dans le corps de la réponse à la demande, vous pouvez vérifier la validité des valeurs des paramètres d'entrée :

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

- Si la présentation est valide, exécutez le workflow en effectuant une demande POST au niveau de l'URL qui comporte les exécutions du workflow :

```
POST https://localhost:8281/vco/api/workflows/CF808080808080808080808080DA808080013086668236014a0614d16e1/executions/
```

Dans la demande, transposez les valeurs aux paramètres d'entrée du workflow. Utilisez les mêmes valeurs que celles retournées pour les paramètres de sortie de la présentation de workflow ou utilisez directement le corps de la dernière demande POST que vous avez effectuée dans la présentation de workflow.

## Interagir avec un workflow au cours de son exécution

L'API REST d'Orchestrator vous permet d'effectuer différentes opérations avec un workflow en cours d'exécution. Vous pouvez obtenir le statut du workflow en cours d'exécution, répondre à une interaction utilisateur en attente et annuler une exécution de workflow.

## Obtenir des objets d'exécution de workflow et vérifier l'état du workflow

Vous pouvez obtenir des informations à propos des exécutions d'un workflow comme, par exemple, les dates de début et de fin, l'état de l'exécution et les valeurs des paramètres d'entrée. Vous pouvez également obtenir les journaux générés pour une exécution de workflow.

## Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

## Procédure

- 1 Récupérez la définition du workflow dont vous souhaitez vérifier l'état en soumettant une demande GET au niveau de l'URL de ce workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

Vous obtenez la définition du workflow dans la réponse à la demande. La définition du workflow contient un lien redirigeant vers les instances d'exécution de ce workflow.

- 2 Récupérez les instances d'exécution disponibles du workflow en soumettant une demande GET au niveau de leurs URL :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/
```

Le corps de la réponse à la demande contient une liste des instances d'exécution disponibles pour le workflow. Cette liste reprend les dates de début et de fin de chaque exécution de workflow, ainsi que l'état et l'initiateur des instances.

- 3 (Facultatif) Pour obtenir des informations plus détaillées sur une exécution précise du workflow, soumettez une demande GET au niveau de l'URL de cette exécution :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/{executionID}/
```

La réponse à cette demande contient la représentation XML de cette exécution de workflow en particulier. Vous pouvez vérifier les valeurs des paramètres d'entrée transmis au cours de cette exécution, l'identité de l'utilisateur qui en est à l'origine, les dates de début et de fin, et l'état de l'exécution.

- 4 (Facultatif) Pour récupérer les journaux générés pour l'exécution de workflow, soumettez une demande GET au niveau de l'URL qui héberge les journaux :

```
GET https://{orchestrator_host}:  
{port}/vco/api/workflows/{workflowID}/executions/{executionID}/logs/
```

- 5 (Facultatif) Pour récupérer des informations supplémentaires sur l'état de l'exécution, soumettez une demande GET au niveau de l'URL qui dirige vers l'état du workflow :

```
GET https://{orchestrator_host}:  
{port}/vco/api/workflows/{workflowID}/executions/{executionID}/state/
```

### Exemple : Obtenir les exécutions du workflow Send Hello et vérifier l'état d'une exécution en particulier

Si vous avez exécuté le workflow Send Hello, vous pouvez obtenir les objets d'exécution disponibles et vérifier les détails les concernant.

- 1 Obtenez la définition du workflow Send Hello en soumettant une demande GET au niveau de l'URL de la définition :

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

- 2 Accédez aux exécutions de workflow disponibles en soumettant une demande GET au niveau de l'URL qui détient les objets d'exécution pour ce workflow :

```
GET https://localhost:  
8281/vco/api/workflows/CF80808080808080808080808080DA808080013086668236014a0614d16e1/executions/  
s/
```

- 3 Dans la réponse à la demande, sélectionnez une exécution de workflow et soumettez une demande GET pour la récupérer :

```
GET https://localhost:  
8281/vco/api/workflows/CF808080808080808080808080808080A808080013086668236014a0614d16e1/execution  
s/888080808080808080808080808080803A8080800132145338690643f66a027ec/
```

Le corps de la réponse contient la représentation XML de l'exécution de workflow avec l'ID de l'emplacement où vous pouvez consulter les détails de cette exécution :

```
.....
<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>
<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>vcodadmin</started-by>
<name>Send Hello</name>
.....
```

## Répondre à une interaction utilisateur en attente

Vous pouvez répondre à une interaction utilisateur en attente dans un workflow à l'aide de l'API REST d'Orchestrator.

## Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

## Procédure

- 1 Récupérez la liste de l'ensemble des objets d'interaction en soumettant une demande GET au niveau de l'URL comportant les objets d'interaction utilisateur disponibles ou en filtrant uniquement les interactions utilisateur en attente :

URL	Description
<code>https://orchestrator_host:port/vco/api/catalog/System/UserInteraction</code>	Détient les objets d'interaction utilisateur disponibles dans Orchestrator
<code>https://orchestrator_host:port/vco/api/catalog/System/UserInteraction?status=0</code>	Filtre uniquement les objets d'interaction utilisateur en attente.

Vous recevez une liste des objets d'interaction utilisateur disponibles. Les interactions utilisateur en attente présentent un attribut avec un nom `state` et une valeur `waiting`.

- 2 Soumettez une demande GET au niveau de l'URL qui détient l'élément d'inventaire de l'interaction utilisateur en attente à laquelle vous souhaitez répondre :

```
GET https://{orchestrator_host}:{port}/vco/api/catalog/System/UserInteraction/{userInteractionID}/
```

L'élément d'inventaire contient un lien redirigeant vers l'instance de l'interaction utilisateur. L'instance de l'interaction utilisateur est associée à une exécution de workflow spécifique.

- 3 Soumettez une demande POST au niveau de l'URL de l'instance de l'interaction utilisateur pour cette exécution spécifique du workflow :

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/{executionID}/interaction/
```

- 4 Veillez à indiquer les valeurs des paramètres d'entrée de l'interaction utilisateur dans un élément `execution-context` de la demande.

Si votre réponse à l'interaction utilisateur a réussi, l'API REST renvoie un code d'état 204.

### Exemple : Répondre à l'interaction utilisateur du workflow Hello World interactif

Vous pouvez exécuter l'exemple de workflow Hello World interactif et répondre à son interaction utilisateur.

- 1 Recherchez l'interaction utilisateur du workflow en attente en soumettant une demande GET au point de terminaison pour obtenir les objets d'interaction utilisateur du service Catalogue :

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

- 2 Localisez l'objet d'inventaire de l'interaction utilisateur pour le workflow interactif Hello World et soumettez une demande GET au niveau de son URL :

```
GET https://localhost:  
8281/vco/api/catalog/System/UserInteraction/8880808080808080808080805A808080013214533869064  
3f66a027ec/
```

- 3 Soumettez une demande POST au niveau de l'URL des objets de l'interaction utilisateur pour l'exécution de workflow en cours :

```
POST https://localhost:
8281/vco/api/workflows/CF808080808080808080808080E6808080013086668236014a0614d16e1/execution
s/88808080808080808080808080578080800132145338690643f66a027ec/interaction/
```

Indiquez une valeur pour le paramètre d'entrée de la demande :

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

## Répondre à une interaction utilisateur après avoir validé les paramètres d'entrée

La présentation d'une interaction utilisateur peut définir les contraintes concernant les valeurs que vous pouvez transmettre aux paramètres d'entrée du workflow. Lorsque vous répondez à une interaction utilisateur, vous devez tenir compte des contraintes définies dans la présentation de l'interaction utilisateur pour valider les valeurs que vous transmettez aux paramètres d'entrée.

## Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

## Procédure

- 1 Récupérez la liste de l'ensemble des objets d'interaction utilisateur en soumettant une demande GET au niveau de l'URL comportant les objets d'interaction utilisateur disponibles ou en filtrant uniquement les interactions utilisateur en attente :

URL	Description
<code>https://orchestrator_host:port/vco/api/catalog/System/UserInteraction</code>	Comporte les objets d'interaction utilisateur disponibles dans Orchestrator.
<code>https://orchestrator_host:port/vco/api/catalog/System/UserInteraction?status=0</code>	Filtre uniquement les objets d'interaction utilisateur en attente.

Vous recevez une liste des objets d'interaction utilisateur disponibles. Les interactions utilisateur en attente présentent un attribut avec un nom `state` et une valeur `waiting`.

- 2 Soumettez une demande GET au niveau de l'URL qui comporte l'élément d'inventaire de l'interaction utilisateur en attente à laquelle vous souhaitez répondre :

```
GET https://{orchestrator_host}:{port}/vco/api/catalog/System/UserInteraction/{userInteractionID}/
```

La réponse contient un lien vers l'instance de l'interaction utilisateur. L'instance de l'interaction utilisateur est associée à une exécution de workflow spécifique.

- 3 Soumettez une demande GET au niveau de l'URL de l'instance de l'interaction utilisateur :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/{executionID}/interaction/
```

Le corps de la réponse comporte un lien redirigeant vers la présentation de l'interaction utilisateur.

- 4 Soumettez une demande GET au niveau de l'URL de la présentation de l'interaction utilisateur :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/{executionID}/interaction/presentation/
```

Vous recevez la définition de la présentation de l'interaction utilisateur dans la réponse.

- 5 Consultez la définition de la présentation pour vérifier les contraintes concernant les valeurs que vous pouvez transmettre aux paramètres d'entrée.
- 6 Exécutez la présentation de l'interaction utilisateur en soumettant une demande POST au niveau de l'URL hébergeant les instances de la présentation :

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/{executionID}/interaction/presentation/instances/
```

- 7 Dans le corps de la demande, veillez à indiquer les valeurs des paramètres d'entrée dans un élément `execution-context`.

Vous recevez l'instance de la présentation de l'interaction utilisateur dans le corps de la réponse. Si les valeurs que vous avez transmises dans les paramètres d'entrée sont valides, vous obtenez un attribut `valid="true"` dans l'élément `execution`. L'élément `output-parameters` comporte les valeurs valides des paramètres d'entrée à utiliser pour répondre à l'interaction utilisateur.

- 8 Répondez à l'interaction utilisateur en soumettant une demande POST au niveau de l'URL hébergeant l'instance de l'interaction :

```
POST https://{orchestrator_host}:
{port}/vco/api/workflows/{workflowID}/executions/{executionID}/interaction/
```

- 9 Dans la demande, indiquez un contexte `execution-context` avec les valeurs des paramètres d'entrée.

Vous pouvez utiliser le même corps de demande que celui de la demande POST effectuée au niveau de l'URL pour la présentation de l'interaction utilisateur.

Si la dernière demande aboutit, vous obtenez un code d'état 204 dans une réponse vide.

### Exemple : Répondre à l'interaction utilisateur du workflow Hello World interactif en validant les paramètres d'entrée

Vous pouvez répondre à l'interaction utilisateur du workflow interactif Hello World en validant les valeurs transmises aux paramètres d'entrée en fonction des contraintes définies dans la présentation de l'interaction utilisateur.

- 1 Recherchez les interactions utilisateur du workflow en attente en soumettant une demande GET au point de terminaison pour obtenir les objets d'interaction utilisateur du service Catalogue :

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

- 2 Localisez l'objet d'inventaire de l'interaction utilisateur pour le workflow interactif Hello World et soumettez une demande GET au niveau de son URL :

```
GET https://localhost:
8281/vco/api/catalog/System/UserInteraction/8880808080808080808080808080805A808080013214533869064
3f66a027ec/
```

- 3 Soumettez une demande GET au niveau de l'URL de l'instance de l'interaction utilisateur :

```
GET https://localhost:
8281/vco/api/catalog/System/UserInteraction/8880808080808080808080808080805A808080013214533869064
3f66a027ec/interaction/
```

- 4 Soumettez une demande GET au niveau de l'URL de la présentation de l'interaction utilisateur :

```
GET https://localhost:  
8281/vco/api/catalog/System/UserInteraction/88808080808080808080808080805A808080013214533869064  
3f66a027ec/interaction/presentation/
```

La présentation définit le paramètre d'entrée comme obligatoire et comporte une contrainte de longueur sur la chaîne que vous pouvez transmettre.

- 5 Soumettez une demande POST au niveau de l'URL qui comporte les instances de la présentation de l'interaction utilisateur :

POST https://localhost:  
8281/vco/api/catalog/System/UserInteraction/88808080808080808080808080808080805A808080013214533869064  
3f66a027ec/interaction/presentation/instances/

Indiquez une valeur pour le paramètre d'entrée dans le corps de la demande :

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

L'élément `execution` du corps de la réponse comporte un attribut `valid="true"` indiquant que la valeur du paramètre d'entrée est valide, conformément aux contraintes de la présentation de l'interaction utilisateur. La valeur valide est répertoriée dans l'élément `output-parameters` :

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

- 6 Soumettez une demande POST au niveau de l'URL de l'instance de l'interaction utilisateur en transmettant la même demande POST qu'à l'étape 5.

```
POST https://localhost:  
8281/vco/api/catalog/System/UserInteraction/88808080808080808080808080805A808080013214533869064  
3f66a027ec/interaction/
```

## Annuler l'exécution d'un workflow

L'API REST d'Orchestrator vous permet d'annuler l'exécution d'un workflow.

**Procédure**

- 1 Récupérez la définition du workflow en soumettant une demande GET au niveau de l'URL de la définition du workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

La définition du workflow contient un lien redirigeant vers les objets d'exécution disponibles de ce workflow.

- 2 Accédez aux exécutions de workflow disponibles en soumettant une demande GET au niveau de l'URL qui comporte les objets d'exécution disponibles pour ce workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/
```

- 3 Dans la liste des exécutions disponibles pour le workflow, sélectionnez celle que vous souhaitez annuler et soumettez une demande DELETE au niveau de son URL :

```
DELETE https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/executions/{executionID}/state
```

**Récupérer les interactions d'un workflow**

Vous pouvez récupérer la liste de toutes les interactions utilisateurs d'un workflow à l'aide de l'API REST d'Orchestrator.

**Procédure**

- 1 Récupérez la définition du workflow en soumettant une demande GET au niveau de l'URL de la définition de ce workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

- 2 Obtenez la liste des interactions du workflow en soumettant une demande GET au niveau de l'URL du schéma du workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/interactions/
```

Si la demande GET aboutit, vous obtenez le code d'état 200 et une liste de l'ensemble des interactions utilisateurs disponibles pour le workflow.

**Accéder à un schéma de workflow**

Vous pouvez accéder à l'image du schéma d'un workflow en utilisant l'API REST d'Orchestrator.

**Procédure**

- 1 Récupérez la définition du workflow en soumettant une demande GET au niveau de l'URL de la définition du workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

- 2 Obtenez l'image du schéma du workflow en soumettant une demande GET dans l'URL du schéma du workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/schema/
```

Si la demande GET aboutit, vous obtenez le code d'état 200 et les données binaires de l'image représentant le schéma de workflow. Le type de contenu de la réponse est défini sur un type de support adéquat, par exemple Content-Type: image/png.

## Utilisation des tâches

L'utilisation du service Tâches de l'API REST d'Orchestrator vous permet de réaliser toutes les opérations relatives à la gestion des tâches dans Orchestrator. Vous pouvez créer une tâche pour planifier un workflow, modifier les propriétés d'une tâche existante, supprimer une tâche, etc.

Le nombre maximal de tâches planifiées pris en charge par Orchestrator est 50.

## Créer une tâche

L'API REST d'Orchestrator vous permet de créer une tâche de planification d'un workflow.

**Prérequis**

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

**Procédure**

- 1 Récupérez la définition du workflow pour lequel vous souhaitez créer une tâche en soumettant une demande GET au niveau de l'URL de ce workflow :

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

La définition du workflow vous permet de consulter le nom et l'identifiant du workflow, ainsi que ses paramètres d'entrée.

- 2 Si vous souhaitez créer une nouvelle tâche pour le workflow, soumettez une demande POST au niveau de l'URL du service Tâche :

```
POST https://{orchestrator_host}:{port}/vco/api/tasks/
```

- 3** Dans la demande, indiquez les paramètres de la nouvelle tâche dans un élément `task`.

Si la demande aboutit, l'API répond avec un code d'état 202 et une réponse vide.

## Exemple : Créer une tâche pour le workflow Send Hello

Vous pouvez créer une tâche qui planifie le workflow Send Hello afin qu'il s'exécute à la quinzième minute de chaque heure à partir d'une date spécifique.

- 1 Soumettez une demande GET au niveau de l'URL du workflow Send Hello afin de récupérer sa définition :

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

- 2 Soumettez une demande POST au niveau de l'URL du service Tâche en indiquant les paramètres de la nouvelle tâche dans la demande :

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/CF808080808080808080808080DA808080013086668236014a0614d16e1/">
        <name>Send Hello</name>
    </workflow>
    <start-mode>normal</start-mode>
</task>
```

## Modifier une tâche

Vous pouvez modifier les propriétés d'une tâche existante à l'aide de l'API REST d'Orchestrator.

Vous pouvez uniquement ajouter de nouvelles propriétés de planification à une tâche ou modifier les valeurs des propriétés déjà existantes. Si vous souhaitez remplacer les propriétés de planification d'une tâche, vous devez supprimer cette dernière et en créer une nouvelle.

## Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

## Procédure

- 1 Soumettez une demande GET au niveau de l'URL de la tâche que vous souhaitez modifier :

```
GET https://{orchestrator_host}:{port}/vco/api/tasks/{task ID}/
```

- 2 Vérifiez les propriétés de la tâche dans le corps de la réponse à la demande.
- 3 Pour modifier la tâche, soumettez une demande POST au niveau de l'URL de la tâche en indiquant les nouvelles propriétés de la tâche dans un élément `task-data` de cette demande.

Si la demande POST aboutit, l'API renvoie le code d'état 200 et la tâche mise à jour dans le corps de la réponse.

## Exemple : Mettre à jour l'exemple de tâche Send Hello

Vous pouvez mettre à jour les dates de début et de fin d'une tâche. Vous pouvez modifier l'exemple de tâche présenté dans [Créer une tâche](#). Vous devez soumettre une demande POST au niveau de l'URL de la tâche en indiquant les nouvelles dates de début et de fin :

```
<?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>
```

## Vérifier l'état d'une tâche

Vous pouvez vérifier l'état des tâches disponibles ou des instances d'exécution d'une tâche donnée.

### Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

### Procédure

- Pour vérifier l'état de toutes les tâches disponibles, soumettez une demande GET au niveau de l'URL du service Tâche :

```
GET https://{orchestrator_host}:{port}/vco/api/tasks/
```

La réponse contient les définitions des tâches disponibles dans Orchestrator. L'état de chaque tâche est disponible dans un élément attribut nommé `state`. La valeur de l'élément peut être respectivement `finished`, `pending`, `running`, etc.

- Pour vérifier l'état de toutes les exécutions d'une tâche donnée, soumettez une demande GET au niveau de l'URL comportant les exécutions de cette tâche :

```
GET https://{orchestrator_host}:{port}/vco/api/tasks/{taskID}/executions/
```

La réponse contient une liste des exécutions disponibles pour cette tâche. L'état de chaque exécution s'affiche dans l'élément `state` de l'objet d'exécution de la tâche.

## Rechercher des objets dans l'inventaire Orchestrator

Le catalogue de services Inventaire vous permet de rechercher n'importe quel objet de l'inventaire Orchestrator. Vous pouvez limiter votre recherche à un sous-ensemble d'objets en appliquant des filtres à la fin des URL au niveau desquelles vous soumettez les demandes HTTP.

Vous pouvez utiliser le service Catalogue pour rechercher des objets d'un certain type dans l'inventaire Orchestrator ou récupérer un objet spécifique grâce à son type et à son ID. Par exemple, vous pouvez récupérer tous les objets du type `workflow` ou `action`, ou un `workflow` ou une `action` spécifique.

Le service Inventaire vous permet de parcourir l'inventaire Orchestrator en suivant les relations parent-enfant. Grâce au service Inventaire, vous pouvez accéder à des objets situés à des emplacements spécifiques de l'inventaire Orchestrator. Par exemple, vous pouvez récupérer l'ensemble des workflows de la gestion du Centre informatique en naviguant jusqu'à leur emplacement de l'inventaire Orchestrator, soit dans `Library/vCenter/Datacenter`

Tous les services de l'API REST d'Orchestrator prennent en charge les paramètres de filtre que vous ajoutez à la fin des URL lorsque vous soumettez des demandes HTTP. Les filtres vous permettent d'affiner les résultats que vous recevez dans les réponses aux demandes formulées au niveau d'URL spécifiques.

## Rechercher des objets par type et par ID

Vous pouvez utiliser le service Catalogue de l'API REST pour rechercher des objets dans Orchestrator par leur type et leur ID.

### Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

### Procédure

- 1 Soumettez une demande GET au niveau de l'URL du service Catalogue :

```
GET https://{orchestrator_host}:{port}/vco/api/catalog/
```

Le corps de la réponse à la demande contient des liens descendants redirigeant vers les points d'entrée de catalogue des plug-ins qui exposent les inventaires dans Orchestrator, ainsi que vers les objets système d'Orchestrator :

- `https://{orchestrator_host}:{port}/vco/api/catalog/{plug-in namespace}/`
- `https://{orchestrator_host}:{port}/vco/api/catalog/System/`

- 2 Pour accéder aux objets exposés par un plug-in ou aux objets système d'Orchestrator, soumettez une demande GET au niveau de l'URL du point d'entrée du catalogue pour le plug-in, ou au niveau de l'URL hébergeant les objets système dans Orchestrator.

Le corps de la réponse à la demande contient des liens redirigeant vers les types d'objet exposés.

- 3 Soumettez une demande GET au niveau de l'URL du type d'objet auquel vous souhaitez accéder.

GET https://{orchestrator\_host}:{port}/vco/api/catalog/{namespace}/{objectType}/

- 4 Soumettez une demande GET au niveau de l'URL de l'objet précis que vous recherchez :

GET https://{orchestrator\_host}:{port}/vco/api/catalog/{namespace}/{objectType}/{objectID}/

### Exemple : Rechercher le workflow Send Hello

Vous pouvez rechercher l'exemple de workflow Send Hello à l'aide du service Catalogue.

- 1 Soumettez une demande GET au niveau de l'URL du service Catalogue :

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

- 2 Soumettez une demande GET au niveau de l'URL hébergeant tous les objets système d'Orchestrator :

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

- 3 Soumettez une demande GET au niveau de l'URL hébergeant tous les workflows :

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

- 4 Soumettez une demande GET au niveau de l'URL du workflow Send Hello :

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

## Rechercher des objets par relation

Vous pouvez utiliser le service Inventaire du REST d'Orchestrator pour parcourir Orchestrator et la hiérarchie des inventaires de plug-ins.

## Prérequis

Vérifiez que vous avez importé le modèle de paquet de workflows dans Orchestrator. Ce paquet est inclus dans le modèle de fichier ZIP Orchestrator que vous pouvez télécharger depuis la page de la documentation Orchestrator.

## Procédure

- 1 Soumettez une demande GET au niveau de l'URL du service Inventaire :

```
GET https://{orchestrator_host}:{port}/vco/api/inventory/
```

La réponse contient des liens descendants redirigeant vers les inventaires enregistrés des plug-ins installés, ainsi que vers les objets système d'Orchestrator sous System.

- 2 Soumettez une demande GET au niveau du lien descendant de l'inventaire auquel vous souhaitez accéder.
- 3 Soumettez des demandes GET au niveau des liens descendants et ascendants de l'inventaire jusqu'à ce que vous atteigniez l'objet que vous recherchez.

## Exemple : Rechercher le workflow Send Hello

Vous pouvez parcourir l'inventaire Orchestrator pour rechercher le workflow Send Hello.

- 1 Soumettez une demande GET au niveau de l'URL du service Inventaire :

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

- 2 Soumettez une demande GET au niveau de l'URL hébergeant les objets système d'Orchestrator :

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

- 3 Soumettez une demande GET au niveau de l'URL hébergeant tous les workflows d'Orchestrator :

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

- 4 Soumettez une demande GET au niveau de l'URL de la catégorie de workflows Exemples :

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

- 5 Utilisez le lien descendant pour atteindre la catégorie de workflow Hello World où trouver le workflow Send Hello.

## Appliquer des filtres

Les services de l'API REST d'Orchestrator prennent en charge des paramètres d'URL supplémentaires qui vous permettent d'affiner la liste d'objets demandés à l'API par HTTP.

Plusieurs paramètres de requête sont pris en charge pour chaque URL menant à une ressource accessible par le biais de l'API REST. Pour connaître les paramètres de requête applicables à une URL, reportez-vous à la documentation de référence de l'*API REST vRealize Orchestrator*.

#### Procédure

- ◆ Pour affiner les résultats d'une requête au niveau d'une URL précise, appliquez des filtres à la fin de l'URL :

*URL?filter\_1&filter\_2&filter\_3&....&filter\_N*. Chaque filtre contient des paramètres de requête valides pour l'URL concernée. Pour en savoir plus sur les paramètres de requête valides de chaque URL, reportez-vous à la documentation de référence de l'API REST d'Orchestrator.

#### Exemple : Filtrer les workflows

Si vous recherchez des workflows contenant un mot précis dans leur nom (par exemple, banque de données), vous pouvez appliquer le filtre suivant dans une demande adressée au service Catalogue :

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

Pour définir un nombre maximal de workflows retournés (par exemple, cinq), appliquez un filtre supplémentaire à la demande :

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

## Importer et exporter des objets Orchestrator

L'API REST d'Orchestrator propose des services Web que vous pouvez utiliser pour importer et exporter des workflows, des actions, des modules, des ressources et des éléments de configuration.

### Importer un workflow

Vous pouvez importer un workflow à l'aide de l'API REST d'Orchestrator.

Selon la bibliothèque de votre application client REST, vous pouvez utiliser un code personnalisé qui définit les propriétés du workflow.

#### Prérequis

Le contenu binaire du workflow doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

#### Procédure

- 1 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés du workflow que vous souhaitez importer.
- 2 Soumettez une demande POST au niveau de l'URL des objets du workflow :

```
POST http://{orchestrator_host}:{port}/vco/api/workflows/
```

Si la demande POST aboutit, vous obtenez le code d'état 202.

## Exporter un workflow

Vous pouvez exporter un workflow à l'aide de l'API REST d'Orchestrator, puis télécharger le workflow en tant que fichier.

### Procédure

- 1 Ouvrez une application client REST et ajoutez un en-tête de demande avec les valeurs suivantes.
  - **Nom** : **accepter**
  - **Valeur** : **application/zip**
- 2 Soumettez une demande GET au niveau de l'URL du workflow que vous souhaitez exporter :

```
GET http://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Le contenu binaire du workflow est disponible en tant que pièce jointe avec *workflow\_name.workflow* comme nom de fichier par défaut. Vous pouvez enregistrer le fichier à l'aide d'une application client REST.

## Importer une action

Vous pouvez importer une action à l'aide de l'API REST d'Orchestrator.

Selon la bibliothèque de votre application client REST, vous pouvez utiliser un code personnalisé qui définit les propriétés de l'action.

### Prérequis

Le contenu binaire de l'action doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

### Procédure

- 1 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés de l'action que vous souhaitez importer.
- 2 Soumettez une demande POST au niveau de l'URL des objets de l'action :

```
POST http://{orchestrator_host}:{port}/vco/api/actions/
```

Si la demande POST aboutit, vous obtenez le code d'état 202.

## Exporter une action

Vous pouvez exporter une action à l'aide de l'API REST d'Orchestrator, puis télécharger l'action en tant que fichier.

## Procédure

- 1 Ouvrez une application client REST et ajoutez un en-tête de demande avec les valeurs suivantes.
  - **Nom** : **accepter**
  - **Valeur** : **application/zip**
- 2 Soumettez une demande GET au niveau de l'URL de l'action que vous souhaitez exporter :

```
GET http://{orchestrator_host}:{port}/vco/api/actions/{actionID}/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Le contenu binaire de l'action est disponible en tant que pièce jointe avec *action\_name.action* comme nom de fichier par défaut. Vous pouvez enregistrer le fichier à l'aide d'une application client REST.

## Importer un module

Vous pouvez importer un module à l'aide de l'API REST d'Orchestrator.

Selon la bibliothèque de votre application client REST, vous pouvez utiliser un code personnalisé qui définit les propriétés du module.

Par défaut, si vous importez un module Orchestrator avec un nom dupliqué, le module existant n'est pas remplacé. Vous pouvez indiquer s'il est nécessaire de remplacer les modules existants à l'aide d'un paramètre dans la demande.

Par défaut, les modules Orchestrator sont importés avec les valeurs d'attributs des éléments de configuration. Vous pouvez importer un module sans valeur d'attribut en utilisant un paramètre dans la demande.

Par défaut, les balises des modules Orchestrator sont importées. Toutefois, si les mêmes balises existent déjà sur le serveur Orchestrator, les valeurs des balises existantes sont préservées. Vous pouvez indiquer si les valeurs des balises existantes doivent être préservées en utilisant des paramètres dans la demande.

## Prérequis

Le contenu binaire du module doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

## Procédure

- 1 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés du module que vous souhaitez importer.
- 2 Soumettez une demande POST au niveau de l'URL des objets du module :

```
POST http://{orchestrator_host}:{port}/vco/api/packages/
```

- 3 (Facultatif) Pour importer un module et remplacer un module existant qui possède le même nom, utilisez le paramètre `overwrite` dans la demande POST :

```
POST http://{orchestrator_host}:{port}/vco/api/packages/?overwrite=true
```

- 4 (Facultatif) Pour importer un module sans les valeurs d'attributs de ses éléments de configuration, utilisez le paramètre `importConfigurationAttributeValues` de la demande POST :

```
POST http://{orchestrator_host}:{port}/vco/api/packages/?importConfigurationAttributeValues=false
```

- 5 (Facultatif) Pour importer un module sans ses balises, utilisez le paramètre `tagImportMode` de la demande POST :

```
POST http://{orchestrator_host}:{port}/vco/api/packages/?tagImportMode=DoNotImport
```

- 6 (Facultatif) Pour importer un module avec ses balises et remplacer les valeurs existantes, utilisez le paramètre `tagImportMode` de la demande POST :

```
POST http://{orchestrator_host}:{port}/vco/api/packages/?tagImportMode=ImportAndOverwriteExistingValue
```

Si la demande POST aboutit, vous obtenez le code d'état 202.

## Exporter un module

Vous pouvez exporter un module à l'aide de l'API REST d'Orchestrator, puis télécharger le module en tant que fichier.

Par défaut, les modules Orchestrator sont exportés avec les valeurs d'attributs des éléments de configuration et des balises globales. Vous pouvez exporter un module sans valeur d'attribut ou balise globale à l'aide des paramètres de la demande. Vous pouvez également indiquer un nom personnalisé pour le fichier de module que vous téléchargez.

### Procédure

- Ouvrez une application client REST et ajoutez un en-tête de demande avec les valeurs suivantes.
  - Nom : `accepter`**
  - Valeur : `application/zip`**
- Soumettez une demande GET au niveau de l'URL du module que vous souhaitez exporter :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/{package_name}/
```

- 3 (Facultatif) Si vous souhaitez attribuer un nom personnalisé au module exporté, utilisez le paramètre `packageName` de la demande GET :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/{package_name}/?packageName={custom_name}
```

- 4 (Facultatif) Pour exporter un module sans les valeurs d'attributs des éléments de configuration, utilisez les paramètres `exportConfigurationAttributeValues` de la demande GET :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/{package_name}/?
exportConfigurationAttributeValues=false
```

- 5 (Facultatif) Pour exporter un module sans les balises globales, utilisez les paramètres `exportGlobalTags` de la demande GET :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/{package_name}/?exportGlobalTags=false
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Le contenu binaire du module est disponible en tant que pièce jointe avec `package_name.package` comme nom de fichier par défaut. Vous pouvez enregistrer le fichier à l'aide d'une application client REST.

## Importer une ressource

Vous pouvez importer une ressource à l'aide de l'API REST d'Orchestrator.

Selon la bibliothèque de votre application client REST, vous pouvez utiliser un code personnalisé qui définit les propriétés de la ressource.

### Prérequis

Le contenu binaire de la ressource doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

### Procédure

- 1 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés de la ressource que vous souhaitez importer.
- 2 Soumettez une demande POST au niveau de l'URL des objets de la ressource :

```
POST http://{orchestrator_host}:{port}/vco/api/resources/
```

Si la demande POST aboutit, vous obtenez le code d'état 202.

## Exporter une ressource

Vous pouvez exporter une ressource à l'aide de l'API REST d'Orchestrator.

### Procédure

- 1 Ouvrez une application client REST et ajoutez un en-tête de demande avec les valeurs suivantes.
  - **Nom** : `accepter`
  - **Valeur** : `application/octet-stream`

- 2 Soumettez une demande GET au niveau de l'URL de la ressource que vous souhaitez exporter :

```
GET http://{orchestrator_host}:{port}/vco/api/resources/{resourceID}/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Le contenu de la ressource s'affiche dans la réponse.

## Importer un élément de configuration

Vous pouvez importer un élément de configuration à l'aide de l'API REST d'Orchestrator.

Selon la bibliothèque de votre application client REST, vous pouvez utiliser un code personnalisé qui définit les propriétés de l'élément de configuration.

### Prérequis

Le contenu binaire de l'élément de configuration doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

### Procédure

- 1 Dans l'application client REST, ajoutez les en-têtes de demande afin de définir les propriétés de l'élément de configuration que vous souhaitez importer.
- 2 Soumettez une demande POST au niveau de l'URL des objets de l'élément de configuration :

```
POST http://{orchestrator_host}:{port}/vco/api/configurations/
```

Si la demande POST aboutit, vous obtenez le code d'état 202.

## Exporter un élément de configuration

Vous pouvez exporter un élément de configuration en utilisant l'API REST d'Orchestrator.

### Procédure

- 1 Ouvrez une application client REST et ajoutez un en-tête de demande avec les valeurs suivantes.
  - **Nom** : **accepter**
  - **Valeur** : **application/vcoobject+xml**
- 2 Soumettez une demande GET au niveau de l'URL de l'élément de configuration que vous souhaitez exporter :

```
GET http://{orchestrator_host}:{port}/vco/api/configurations/{configuration_elementID}/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Le contenu de l'élément de configuration s'affiche dans la réponse.

## Supprimer des objets Orchestrator

L'API REST d'Orchestrator propose des services Web que vous pouvez utiliser pour supprimer des workflows, des actions, des modules, des ressources et des éléments de configuration.

### Supprimer un workflow

Vous pouvez supprimer un workflow à l'aide de l'API REST d'Orchestrator.

#### Procédure

- 1 Faites une demande GET et récupérez l'ID du workflow dans la liste des workflows renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/workflows/
```

- 2 Soumettez une demande DELETE au niveau de l'URL du workflow :

```
DELETE http://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 200 avec une réponse vide.

### Supprimer une action

Vous pouvez supprimer une action à l'aide de l'API REST d'Orchestrator.

#### Procédure

- 1 Faites une demande GET et récupérez l'ID de l'action dans la liste des actions renvoyées :

```
GET http://{orchestrator_host}:{port}/vco/api/actions/
```

- 2 Soumettez une demande DELETE au niveau de l'URL de l'action :

```
DELETE http://{orchestrator_host}:{port}/vco/api/actions/{actionID}/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 200 avec une réponse vide.

### Supprimer un module

Vous pouvez supprimer un module à l'aide de l'API REST d'Orchestrator.

Lorsque vous supprimez un module, les éléments du module ne sont pas supprimés. Si vous souhaitez supprimer le contenu d'un module, vous devez indiquer un paramètre d'option.

## Procédure

- 1 Faites une demande GET et récupérez le nom du paquet dans la liste des paquets renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/
```

- 2 Soumettez une demande DELETE au niveau de l'URL du module et, si vous souhaitez supprimer les éléments du module, indiquez un paramètre d'option à la fin de la demande :

```
DELETE http://{orchestrator_host}:{port}/vco/api/packages/{package_name}/?option={parameter}
```

Paramètre	Description
<b>deletePackage</b>	Seul le module est supprimé, tandis que son contenu est conservé.
<b>deletePackageWithContent</b>	Le module est supprimé avec tout son contenu. Si d'autres modules partagent des éléments avec le module supprimé, les éléments partagés sont supprimés des autres modules.
<b>deletePackageKeepingShared</b>	Le module et le contenu non partagé sont supprimés. Les éléments partagés avec d'autres modules ne sont pas supprimés.

Si vous n'indiquez aucun paramètre d'option, le paramètre `deletePackage` par défaut est utilisé.

Si la demande DELETE aboutit, vous obtenez le code d'état 200 avec une réponse vide.

## Supprimer une ressource

Vous pouvez supprimer une ressource à l'aide de l'API REST d'Orchestrator.

### Procédure

- 1 Faites une demande GET et récupérez l'ID de la ressource dans la liste des ressources renvoyées :

```
GET http://{orchestrator_host}:{port}/vco/api/resources/
```

- 2 Soumettez une demande DELETE au niveau de l'URL de la ressource :

```
DELETE http://{orchestrator_host}:{port}/vco/api/resources/{resourceID}/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 200 avec une réponse vide.

## Supprimer un élément de configuration

Vous pouvez supprimer un élément de configuration en utilisant l'API REST d'Orchestrator.

### Procédure

- 1 Faites une demande GET et récupérez l'ID de l'élément de configuration dans la liste des éléments de configuration renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/configurations/
```

## 2 Soumettez une demande DELETE au niveau de l'URL de l'élément de configuration :

```
DELETE http://{orchestrator_host}:{port}/vco/api/configurations/{configuration_elementID}/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 200 avec une réponse vide.

## Définir des autorisations sur les objets Orchestrator

L'API REST d'Orchestrator vous permet de définir des autorisations personnalisées pour un objet Orchestrator. Pour définir les autorisations, vous devez soumettre une demande POST au niveau de l'URL des autorisations de l'objet et indiquer les autorisations dans la demande.

Vous pouvez également utiliser l'API REST d'Orchestrator pour récupérer les informations sur les autorisations d'un objet ou supprimer les autorisations existantes.

## Autorisations de l'API REST

L'utilisation de l'API REST d'Orchestrator pour définir des autorisations nécessite la définition d'un ensemble de caractères.

Vous pouvez définir les autorisations pour un élément en insérant une séquence de caractères dans la balise `<rights>` de la demande POST.

Les caractères que vous pouvez utiliser pour définir les autorisations dans l'API REST d'Orchestrator revêtent des significations particulières.

**Tableau 2-1. Caractères pour les autorisations de l'API REST d'Orchestrator**

Caractère	Description
r	Autorisations d'affichage.
x	Autorisations d'exécution.
i	Autorisations d'inspection.
c	Autorisations de modification.
a	Autorisations d'administration.

## Exemple : Syntaxe de définition des autorisations

Vous pouvez utiliser les exemples de syntaxe suivants dans la demande POST, au niveau de l'URL des autorisations de l'élément d'Orchestrator.

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

Lorsque vous définissez des autorisations `ric` dans la balise `<rights>` de la demande, vous permettez aux membres du groupe d'utilisateurs `vcousers` d'afficher, de contrôler et de modifier l'élément Orchestrator.

## Récupérer les autorisations d'un workflow

L'API REST d'Orchestrator vous permet de récupérer les informations relatives aux autorisations d'un workflow.

### Procédure

- 1 Faites une demande GET et récupérez l'ID du workflow dans la liste des workflows renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/workflows/
```

- 2 Soumettez une demande GET au niveau de l'URL des autorisations du workflow :

```
GET http://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/permissions/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Les informations relatives aux autorisations du workflow s'affichent dans la réponse.

## Supprimer les autorisations d'un workflow

Vous pouvez supprimer les autorisations d'un workflow à l'aide de l'API REST d'Orchestrator. Vous pouvez supprimer les autorisations existantes d'un workflow avant de définir les nouvelles autorisations.

### Procédure

- 1 Faites une demande GET et récupérez l'ID du workflow dans la liste des workflows renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/workflows/
```

- 2 Soumettez une demande DELETE au niveau de l'URL des autorisations du workflow :

```
DELETE http://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/permissions/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 204 avec une réponse vide.

## Définir les autorisations d'un workflow

L'API REST d'Orchestrator vous permet de définir les autorisations pour un workflow.

### Prérequis

Consultez les types d'autorisations que vous pouvez définir et la syntaxe à utiliser dans la demande. Reportez-vous à [Autorisations de l'API REST](#).

**Procédure**

- 1 Faites une demande GET et récupérez l'ID du workflow dans la liste des workflows renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/workflows/
```

- 2 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés du workflow pour lequel vous souhaitez configurer des autorisations.
- 3 Dans la demande, indiquez les autorisations que vous souhaitez définir.
- 4 Soumettez une demande POST au niveau de l'URL des autorisations du workflow :

```
POST http://{orchestrator_host}:{port}/vco/api/workflows/{workflowID}/permissions/
```

Si la demande POST aboutit, vous obtenez le code d'état 201. Les informations relatives aux autorisations du workflow s'affichent dans la réponse.

**Récupérer les autorisations d'une action**

L'API REST d'Orchestrator vous permet de récupérer les informations relatives aux autorisations d'une action.

**Procédure**

- 1 Faites une demande GET et récupérez l'ID de l'action dans la liste des actions renvoyées :

```
GET http://{orchestrator_host}:{port}/vco/api/actions/
```

- 2 Soumettez une demande GET au niveau de l'URL des autorisations de l'action :

```
GET http://{orchestrator_host}:{port}/vco/api/actions/{actionID}/permissions/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Les informations relatives aux autorisations de l'action s'affichent dans la réponse.

**Supprimer les autorisations d'une action**

Vous pouvez supprimer les autorisations d'une action à l'aide de l'API REST d'Orchestrator. Vous pouvez supprimer les autorisations existantes d'une action avant de définir les nouvelles autorisations.

**Procédure**

- 1 Faites une demande GET et récupérez l'ID de l'action dans la liste des actions renvoyées :

```
GET http://{orchestrator_host}:{port}/vco/api/actions/
```

- 2 Soumettez une demande DELETE au niveau de l'URL des autorisations de l'action :

```
DELETE http://{orchestrator_host}:{port}/vco/api/actions/{actionID}/permissions/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 204 avec une réponse vide.

## Définir les autorisations d'une action

L'API REST d'Orchestrator vous permet de définir les autorisations pour une action.

### Prérequis

Consultez les types d'autorisations que vous pouvez définir et la syntaxe à utiliser dans la demande. Reportez-vous à [Autorisations de l'API REST](#).

### Procédure

- 1 Faites une demande GET et récupérez l'ID de l'action dans la liste des actions renvoyées :

```
GET http://{orchestrator_host}:{port}/vco/api/actions/
```

- 2 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés de l'action pour laquelle vous souhaitez configurer des autorisations.
- 3 Dans la demande, indiquez les autorisations que vous souhaitez définir.
- 4 Soumettez une demande POST au niveau de l'URL des autorisations de l'action :

```
POST http://{orchestrator_host}:{port}/vco/api/actions/{actionID}/permissions/
```

Si la demande POST aboutit, vous obtenez le code d'état 201. Les informations relatives aux autorisations de l'action s'affichent dans la réponse.

## Récupérer les autorisations d'un module

L'API REST d'Orchestrator vous permet de récupérer les informations relatives aux autorisations d'un module.

### Procédure

- 1 Faites une demande GET et récupérez le nom du paquet dans la liste des paquets renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/
```

- 2 Soumettez une demande GET au niveau de l'URL des autorisations du module :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/{package_name}/permissions/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Les informations relatives aux autorisations du module s'affichent dans la réponse.

## Supprimer les autorisations d'un module

Vous pouvez supprimer les autorisations d'un module à l'aide de l'API REST d'Orchestrator. Vous pouvez supprimer les autorisations existantes d'un module avant de définir les nouvelles autorisations.

### Procédure

- 1 Faites une demande GET et récupérez le nom du paquet dans la liste des paquets renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/
```

- 2 Soumettez une demande DELETE au niveau de l'URL des autorisations du module :

```
DELETE http://{orchestrator_host}:{port}/vco/api/packages/{package_name}/permissions/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 204 avec une réponse vide.

## Définir les autorisations d'un module

L'API REST d'Orchestrator vous permet de définir les autorisations pour un module.

### Prérequis

Consultez les types d'autorisations que vous pouvez définir et la syntaxe à utiliser dans la demande. Reportez-vous à [Autorisations de l'API REST](#).

### Procédure

- 1 Faites une demande GET et récupérez le nom du paquet dans la liste des paquets renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/packages/
```

- 2 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés du module pour lequel vous souhaitez configurer des autorisations.
- 3 Dans la demande, indiquez les autorisations que vous souhaitez définir.
- 4 Soumettez une demande POST au niveau de l'URL des autorisations du module :

```
POST http://{orchestrator_host}:{port}/vco/api/packages/{package_name}/permissions/
```

Si la demande POST aboutit, vous obtenez le code d'état 201. Les informations relatives aux autorisations du module s'affichent dans la réponse.

## Récupérer les autorisations d'une ressource

L'API REST d'Orchestrator vous permet de récupérer les informations relatives aux autorisations d'une ressource.

### Procédure

- 1 Faites une demande GET et récupérez l'ID de la ressource dans la liste des ressources renvoyées :

```
GET http://{orchestrator_host}:{port}/vco/api/resources/
```

- 2 Soumettez une demande GET au niveau de l'URL des autorisations de la ressource :

```
GET http://{orchestrator_host}:{port}/vco/api/resources/{resourceID}/permissions/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Les informations relatives aux autorisations de la ressource s'affichent dans la réponse.

## Supprimer les autorisations d'une ressource

Vous pouvez supprimer les autorisations d'une ressource à l'aide de l'API REST d'Orchestrator. Vous pouvez supprimer les autorisations existantes d'une ressource avant de définir les nouvelles autorisations.

### Procédure

- 1 Faites une demande GET et récupérez l'ID de la ressource dans la liste des ressources renvoyées :

```
GET http://{orchestrator_host}:{port}/vco/api/resources/
```

- 2 Soumettez une demande DELETE au niveau de l'URL des autorisations de la ressource :

```
DELETE http://{orchestrator_host}:{port}/vco/api/resources/{resourceID}/permissions/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 204 avec une réponse vide.

## Définir les autorisations d'une ressource

L'API REST d'Orchestrator vous permet de définir les autorisations pour une ressource.

### Prérequis

Consultez les types d'autorisations que vous pouvez définir et la syntaxe à utiliser dans la demande. Reportez-vous à [Autorisations de l'API REST](#).

**Procédure**

- 1 Faites une demande GET et récupérez l'ID de la ressource dans la liste des ressources renvoyées :

```
GET http://{orchestrator_host}:{port}/vco/api/resources/
```

- 2 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés de la ressource pour laquelle vous souhaitez configurer des autorisations.
- 3 Dans la demande, indiquez les autorisations que vous souhaitez définir.
- 4 Soumettez une demande POST au niveau de l'URL des autorisations de la ressource :

```
POST http://{orchestrator_host}:{port}/vco/api/resources/{resourceID}/permissions/
```

Si la demande POST aboutit, vous obtenez le code d'état 201. Les informations relatives aux autorisations de la ressource s'affichent dans la réponse.

**Récupérer les autorisations d'un élément de configuration**

L'API REST d'Orchestrator vous permet de récupérer les informations relatives aux autorisations d'un élément de configuration.

**Procédure**

- 1 Faites une demande GET et récupérez l'ID de l'élément de configuration dans la liste des éléments de configuration renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/configurations/
```

- 2 Soumettez une demande GET au niveau de l'URL des autorisations de l'élément de configuration :

```
GET http://{orchestrator_host}:{port}/vco/api/configurations/{configuration_elementID}/permissions/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Les informations relatives aux autorisations de l'élément de configuration s'affichent dans la réponse.

**Supprimer les autorisations d'un élément de configuration**

Vous pouvez supprimer les autorisations d'un élément de configuration à l'aide de l'API REST d'Orchestrator. Vous pouvez supprimer les autorisations existantes d'un élément de configuration avant de définir les nouvelles autorisations.

**Procédure**

- 1 Faites une demande GET et récupérez l'ID de l'élément de configuration dans la liste des éléments de configuration renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/configurations/
```

- 2 Soumettez une demande DELETE au niveau de l'URL des autorisations de l'élément de configuration :

```
DELETE http://{orchestrator_host}:
{port}/vco/api/configurations/{configuration_elementID}/permissions/
```

Si la demande DELETE aboutit, vous obtenez le code d'état 204 avec une réponse vide.

## Définir les autorisations d'un élément de configuration

L'API REST d'Orchestrator vous permet de définir les autorisations pour un élément de configuration.

### Prérequis

Consultez les types d'autorisations que vous pouvez définir et la syntaxe à utiliser dans la demande. Reportez-vous à [Autorisations de l'API REST](#).

### Procédure

- 1 Faites une demande GET et récupérez l'ID de l'élément de configuration dans la liste des éléments de configuration renvoyés :

```
GET http://{orchestrator_host}:{port}/vco/api/configurations/
```

- 2 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés de l'élément de configuration pour lequel vous souhaitez configurer des autorisations.
- 3 Dans la demande, indiquez les autorisations que vous souhaitez définir.
- 4 Soumettez une demande POST au niveau de l'URL des autorisations de l'élément de configuration :

```
POST http://{orchestrator_host}:
{port}/vco/api/configurations/{configuration_elementID}/permissions/
```

Si la demande POST aboutit, vous obtenez le code d'état 201. Les informations relatives aux autorisations de l'élément de configuration s'affichent dans la réponse.

## Effectuer des opérations avec des plug-ins

L'API REST d'Orchestrator propose des services Web que vous pouvez utiliser pour effectuer différentes opérations avec des plug-ins.

### Récupérer des informations à propos des plug-ins

Vous pouvez récupérer les informations relatives aux métadonnées de plug-ins installés à l'aide de l'API REST d'Orchestrator.

### Procédure

- 1 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés des plug-ins.

- 2 Soumettez une demande GET au niveau de l'URL des objets du plug-in :

```
GET http://{orchestrator_host}:{port}/vco/api/plugins/
```

Si la demande GET aboutit, vous obtenez le code d'état 200.

## Importer un plug-in

Vous pouvez importer un plug-in à l'aide de l'API REST d'Orchestrator.

Selon la bibliothèque de votre application client REST, vous pouvez utiliser un code personnalisé qui définit les propriétés du plug-in.

---

**Remarque** Vous ne pouvez pas finaliser l'importation si un plug-in possédant le même nom est déjà installé.

---

### Prérequis

Le contenu binaire du plug-in doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

### Procédure

- 1 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés du plug-in que vous souhaitez importer.
- 2 Soumettez une demande POST au niveau de l'URL des objets du plug-in :

```
POST http://{orchestrator_host}:{port}/vco/api/plugins/
```

Si la demande POST aboutit, vous obtenez le code d'état 200.

## Exporter un plug-in

Vous pouvez exporter un plug-in à l'aide de l'API REST d'Orchestrator.

### Procédure

- 1 Ouvrez une application client REST et ajoutez un en-tête de demande avec les valeurs suivantes.
  - **Nom** : **accepter**
  - **Valeur** : **application/dar**
- 2 Soumettez une demande GET au niveau de l'URL du plug-in que vous souhaitez exporter :

```
GET http://{orchestrator_host}:{port}/vco/api/plugins/{plug-in_name}/
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Le contenu du plug-in s'affiche dans la réponse.

## Activer ou désactiver un plug-in

Vous pouvez activer ou désactiver un plug-in à l'aide de l'API REST d'Orchestrator.

Vous pouvez modifier l'état d'un plug-in en le faisant passer d'activé à désactivé ou de désactivé à activé en soumettant une demande PUT au niveau de l'URL du plug-in. Vous pouvez vérifier l'état d'un plug-in en récupérant les informations à propos des plug-ins Orchestrator. Reportez-vous à [Récupérer des informations à propos des plug-ins](#).

### Prérequis

Le contenu binaire du plug-in doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

### Procédure

- 1 Dans l'application client REST, ajoutez les en-têtes de demande afin de définir les propriétés du plug-in que vous souhaitez activer ou désactiver.
- 2 Soumettez une demande PUT au niveau de l'URL du plug-in que vous souhaitez activer ou désactiver :

```
PUT http://{orchestrator_host}:{port}/vco/api/plugins/{plug-in_name}/state/
```

Si la demande PUT aboutit, vous obtenez le code d'état 200.

## Opérations de configuration du serveur

L'API REST d'Orchestrator propose des services Web que vous pouvez utiliser pour effectuer différentes opérations associées à la configuration du serveur Orchestrator.

### Récupérer des informations sur la configuration du serveur Orchestrator

Vous pouvez récupérer des informations sur la configuration du serveur Orchestrator à l'aide de l'API REST d'Orchestrator.

### Procédure

- 1 Dans l'application client REST, ajoutez des en-têtes de demande afin de définir les propriétés du serveur pour lequel vous souhaitez récupérer des informations.
- 2 Soumettez une demande GET au niveau de l'URL des objets du plug-in :

```
GET http://{orchestrator_host}:{port}/vco/api/server-configuration/
```

Si la demande GET aboutit, vous obtenez le code d'état 200.

## Importer la configuration du serveur Orchestrator

Vous pouvez importer une configuration enregistrée à l'aide de l'API REST d'Orchestrator.

### Prérequis

Le contenu binaire de la configuration doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

### Procédure

- 1 Ouvrez une application client REST et ajoutez un en-tête de demande avec les valeurs suivantes.

- **Nom** : `content-type`
- **Valeur** : `multipart/form-data`

- 2 Soumettez une demande POST au niveau de l'URL de la configuration du serveur :

```
POST http://{orchestrator_host}:{port}/vco/api/server-configuration/
```

Si la demande POST aboutit, vous obtenez le code d'état 200.

## Exporter la configuration du serveur Orchestrator

Vous pouvez exporter la configuration du serveur à l'aide de l'API REST d'Orchestrator.

### Prérequis

Le contenu binaire de la configuration doit être disponible en tant que contenu en plusieurs parties. Pour des informations détaillées, reportez-vous à RFC 2387.

### Procédure

- 1 Ouvrez une application client REST et ajoutez un en-tête de demande avec les valeurs suivantes.

- **Nom** : `content-type`
- **Valeur** : `multipart/form-data`

- 2 Ajoutez un autre en-tête de demande avec les valeurs suivantes.

- **Nom** : `accepter`
- **Valeur** : `*/*`

- 3 Soumettez une demande POST au niveau de l'URL de la configuration du serveur :

```
POST http://{orchestrator_host}:{port}/vco/api/server-configuration/
```

Si la demande POST aboutit, vous obtenez le code d'état 200.

## Réaliser des opérations de balisage

L'API REST d'Orchestrator propose des services Web que vous pouvez utiliser pour effectuer différentes opérations permettant de faciliter la recherche des objets en exploitant les balises dans Orchestrator.

Vous pouvez faciliter la recherche des objets en dotant ces derniers de balises. Les balises sont des chaînes de 3 à 64 caractères qui ne doivent contenir aucun espace.

Vous pouvez ajouter des balises globales et des balises privées. Les balises globales sont visibles par tous les utilisateurs d'Orchestrator tandis que les balises privées ne peuvent être vues que par l'utilisateur qui les a créées. Les balises globales peuvent uniquement être créées et supprimées par des utilisateurs dotés de privilèges administrateur.

### Baliser un objet

L'API REST d'Orchestrator vous permet d'affecter des balises à un objet.

Vous pouvez créer des balises privées et des balises globales. Vous indiquez si la balise est privée ou globale dans la demande.

---

**Remarque** Pour créer des balises globales, vous devez être connecté en tant qu'utilisateur avec des privilèges administratifs.

---

Vous pouvez également affecter une valeur à la balise que vous créez. La valeur est un paramètre facultatif que vous pouvez utiliser pour filtrer les balises.

#### Procédure

- 1 Formulez la demande à l'aide de la syntaxe suivante.

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

---

**Remarque** Vous pouvez créer une balise globale en définissant la variable **globale** sur « **vrai** ».

---

- 2 Soumettez une demande POST au niveau de l'URL de l'objet :

```
POST http://{orchestrator_host}:{port}/vco/api/catalog/{namespace}/{objectType}/{objectId}/tags
```

Si la demande POST aboutit, vous obtenez le code d'état 200.

### Supprimer les balises d'un objet

L'API REST d'Orchestrator vous permet de supprimer les balises attribuées à un objet.

Vous pouvez supprimer des balises privées et des balises globales.

**Remarque** Pour supprimer des balises globales, vous devez être connecté en tant qu'utilisateur avec des privilèges administratifs.

#### Procédure

- ◆ Soumettez une demande DELETE pour supprimer des balises privées ou globales.
  - Pour supprimer une balise privée, soumettez une demande DELETE au niveau de l'URL de l'objet en formulant la syntaxe suivante :

```
DELETE http://{orchestrator_host}:
{port}/vco/api/catalog/{namespace}/{objectType}/{objectId}/tag/{tag_name}
```

- Pour supprimer une balise globale, soumettez une demande DELETE au niveau de l'URL de l'objet en formulant la syntaxe suivante :

```
DELETE http://{orchestrator_host}:
{port}/vco/api/catalog/{namespace}/{objectType}/{objectId}/tag/{tag_name}
```

Si la demande DELETE aboutit, vous obtenez le code d'état 200.

## Répertorier les balises d'un objet

Vous pouvez récupérer une liste de balises affectées à un objet à l'aide de l'API REST d'Orchestrator.

#### Procédure

- ◆ Soumettez une demande GET au niveau de l'URL de l'objet :

```
GET http://{orchestrator_host}:{port}/vco/api/catalog/{namespace}/{objectType}/{objectId}/tags
```

Si la demande GET aboutit, vous obtenez le code d'état 200.

## Répertorier les objets balisés par type

Vous pouvez utiliser l'API REST d'Orchestrator pour récupérer une liste d'objets balisés avec une balise spécifique et filtrer ces objets par type.

#### Procédure

- ◆ Soumettez une demande GET au niveau de l'URL du type de l'objet :

```
GET http://{orchestrator_host}:{port}/vco/api/catalog/{namespace}/{objectType}/?
tags=tag1&tags=tag2=value
```

Si la demande GET aboutit, vous obtenez le code d'état 200.

## Répertorier les propriétaires des balises

Vous pouvez récupérer une liste de propriétaires de balises à l'aide de l'API REST d'Orchestrator. Les propriétaires de balises sont les utilisateurs ayant créé au moins une balise.

### Procédure

- ◆ Soumettez une demande GET au niveau de l'URL suivante :

```
GET http://{orchestrator_host}:{port}/vco/api/tags
```

Si la demande GET aboutit, vous obtenez le code d'état 200. La liste que vous récupérez contient les utilisateurs ayant créé au moins une balise. Les balises globales sont répertoriées sous le nom d'utilisateur \_\_GLOBAL\_\_ du système.

## Répertorier les balises en fonction des utilisateurs

Vous pouvez utiliser l'API REST d'Orchestrator pour récupérer une liste de balises créées par un utilisateur spécifique.

Vous pouvez également récupérer des balises globales. Les balises globales sont répertoriées sous le nom d'utilisateur \_\_GLOBAL\_\_ du système.

### Procédure

- ◆ Soumettez une demande GET au niveau de l'URL de l'utilisateur.
  - Pour récupérer une liste des balises créées par un utilisateur spécifique, soumettez une demande GET à l'aide de la syntaxe suivante :

```
GET http://{orchestrator_host}:{port}/vco/api/tags/{user_name}
```

- Pour récupérer une liste des balises globales, soumettez une demande GET à l'aide de la syntaxe suivante :

```
GET http://{orchestrator_host}:{port}/vco/api/tags/__GLOBAL__
```

Si la demande GET aboutit, vous obtenez le code d'état 200.

## Répertorier les balises en fonction des utilisateurs filtrés par nom de balise

Vous pouvez utiliser l'API REST d'Orchestrator pour récupérer une liste d'instances de balises créées par un utilisateur spécifique, puis filtrer ces balises par nom.

Vous pouvez également récupérer les instances de balises globales. Les balises globales sont répertoriées sous le nom d'utilisateur \_\_GLOBAL\_\_ du système.

**Procédure**

- ◆ Soumettez une demande GET au niveau de l'URL de l'utilisateur.
  - Pour récupérer une liste filtrée des instances de balises créées par un utilisateur spécifique, soumettez une demande GET à l'aide de la syntaxe suivante :

```
GET http://{orchestrator_host}:{port}/vco/api/tags/{user_name}/{tag_name}
```

- Pour récupérer une liste filtrée des instances de balises globales, soumettez une demande GET à l'aide de la syntaxe suivante :

```
GET http://{orchestrator_host}:{port}/vco/api/tags/__GLOBAL__/{tag_name}
```

Si la demande GET aboutit, vous obtenez le code d'état 200. Les informations que vous récupérez se réfèrent à l'objet balisé, au nom de la balise et à la valeur de la balise. Elles indiquent également si l'instance de la balise est globale ou privée.

## Supprimer les balises en fonction des utilisateurs

Vous pouvez utiliser l'API REST d'Orchestrator pour supprimer toutes les balises créées par un utilisateur spécifique.

Vous pouvez également supprimer les balises globales. Les balises globales sont répertoriées sous le nom d'utilisateur \_\_GLOBAL\_\_ du système.

---

**Remarque** Pour supprimer des balises globales, vous devez être connecté en tant qu'utilisateur avec des privilèges administratifs.

---

**Procédure**

- ◆ Soumettez une demande DELETE au niveau de l'URL de l'utilisateur.
  - Pour supprimer les balises créées par un utilisateur spécifique, soumettez une demande DELETE à l'aide de la syntaxe suivante :

```
DELETE http://{orchestrator_host}:{port}/vco/api/tags/{user_name}
```

- Pour supprimer les balises globales, soumettez une demande DELETE à l'aide de la syntaxe suivante :

```
DELETE http://{orchestrator_host}:{port}/vco/api/tags/__GLOBAL__
```

Si la demande DELETE aboutit, vous obtenez le code d'état 204.