

管理指南

vCenter Orchestrator 4.0

ZH_CN-000127-00

vmware[®]

最新的技术文档可以从 VMware 网站下载:

<http://www.vmware.com/cn/support/pubs/>

VMware 网站还提供最近的产品更新信息。

您如果对本文档有任何意见或建议, 请把反馈信息提交至:

docfeedback@vmware.com

版权所有 © 2009 VMware, Inc. 保留所有权利。本产品受美国和国际版权及知识产权法的保护。VMware 产品受一项或多项专利保护, 有关专利详情, 请访问 <http://www.vmware.com/go/patents-cn>。

VMware 是 VMware, Inc. 在美国和/或其他法律辖区的注册商标或商标。此处提到的所有其他商标和名称分别是其各自公司的商标。

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

北京办公室
北京市海淀区科学院南路 2 号
融科资讯中心 C 座南 8 层
www.vmware.com/cn

上海办公室
上海市浦东新区浦东南路 999 号
新梅联合广场 23 楼
www.vmware.com/cn

广州办公室
广州市天河北路 233 号
中信广场 7401 室
www.vmware.com/cn

目录

关于本文档	5
1 VMware vCenter Orchestrator 简介	7
Orchestrator 平台的主要功能特征	7
Orchestrator 用户角色和相关任务	8
Orchestrator 架构	9
2 Orchestrator 客户端工作区	11
登录 Orchestrator 客户端	11
3 管理工作流	13
“Workflows” 视图	13
“Workflows” 视图组件	13
工作流库中的标准工作流	14
工作流的主要概念	15
工作流属性	15
工作流参数	15
工作流架构	16
查看工作流架构	16
工作流令牌	16
工作流令牌状况	16
对工作流设置用户权限	17
运行工作流	18
4 管理操作	21
“Actions” 视图	21
创建操作	22
复制操作	22
导出操作	23
导入操作	23
移动操作	23
查找执行操作的元素	24
5 管理任务	25
“Tasks” 视图	25
创建任务	25
编辑任务重复执行方式	26
6 策略	27

- 7 使用软件包 29
 - “Packages” 视图 29
 - 创建软件包 30
 - 对软件包设置用户权限 30
 - 导出软件包 31
 - 导入软件包 32
 - 获取并同步远程软件包 32

- 8 Orchestrator 默认插件 35

- 9 修改通过插件搜索获得的对象数 39

- 10 使用 JavaScript 访问 Java 类 41

- 11 维护和恢复 43
 - 更改服务器日志的大小 44

- 索引 45

关于本文档

《VMware vCenter Orchestrator 4.0 管理指南》提供有关如何使用和维护 VMware® vCenter Orchestrator 的信息和说明。本文档还将介绍如何管理工作流、插件、软件包、清单和策略。

目标读者

本文档的目标读者为熟悉虚拟机技术和数据中心操作的 vCenter 高级管理人员和经验丰富的系统管理员，以及任何希望实现以下目标的人员：

- 自动执行与虚拟环境管理相关的频繁重复过程。
- 管理异构系统中及相互之间的多个自动过程。
- 通过集中管理自动脚本来实现 IT 过程的透明化。
- 更快对虚拟环境中的非计划更改作出反应。

文档反馈

VMware 欢迎您提出宝贵建议，以便改进我们的文档。如有意见，请将反馈发送到 docfeedback@vmware.com。

技术支持和教育资源

您可以获得以下技术支持资源。有关本文档和其他文档的最新版本，请访问：
<http://www.vmware.com/cn/support/pubs>。

在线支持和电话支持

要通过在线支持提交技术支持请求、查看产品和合同信息以及注册您的产品，请访问 <http://www.vmware.com/cn/support>。

客户只要拥有相应的支持合同，就可以通过电话支持，以最快的速度获得对高优先级问题的答复。请访问

http://www.vmware.com/cn/support/phone_support.html。

支持服务项目

要了解 VMware 支持服务项目如何帮助您满足业务需求，请访问 <http://www.vmware.com/cn/support/services>。

VMware 专业服务

VMware 教育服务课程提供了大量实践操作环境、案例研究示例，以及用作作业参考材料的课程材料。这些课程可以通过现场指导、教室授课的方式学习，也可以通过在线直播的方式学习。关于现场试点项目及实施的最佳实践，VMware 咨询服务可提供多种服务，协助您评估、计划、构建和管理虚拟环境。要了解有关教育课程、认证计划和咨询服务的信息，请访问 <http://www.vmware.com/cn/services>。

VMware vCenter Orchestrator 简介

VMware vCenter Orchestrator 是一个开发和过程自动化平台，提供可扩展的工作流库，用于创建和运行自动化的可配置过程来管理 VMware vCenter 基础架构。

Orchestrator 公开了 vCenter Server API 中的每一步操作，以便您可以将所有这些操作整合到自动化过程中。通过 Orchestrator 的开放式插件架构，您还可以将其与其他管理解决方案集成在一起。

本章讨论了以下主题：

- 第 7 页，“Orchestrator 平台的主要功能特征”
- 第 8 页，“Orchestrator 用户角色和相关任务”
- 第 9 页，“Orchestrator 架构”

Orchestrator 平台的主要功能特征

Orchestrator 由三个截然不同的层组成：一个可提供耦合工具所需常用功能的耦合平台、一个集成了子系统控制功能的插件架构和一个既存过程库。Orchestrator 是一个开放式平台，它不但可以通过添加新插件和库进行扩展，还可以通过一组 API 集成到更大型的 SOAP 架构中。

下面列出了 Orchestrator 的主要功能特征。

持久性	使用生产级外部数据库存储相关信息，如过程、状况和配置信息。
集中式管理	Orchestrator 提供集中过程管理方式。通过基于应用程序服务器的平台（以及完整的版本历史记录），您可以将脚本和与过程相关的基元放在同一位置。采用这种方式，可以防止未实现版本管理和适当变更控制的脚本在您的服务器上传播。
检查点	过程的每一个步骤都将保存在数据库中，这使您可以重新启动服务器而不会丢失状态和相关环境。此功能对于长时间运行的过程特别有用。
版本管理	所有的 Orchestrator 平台对象都具有相关的版本历史记录。通过此功能，您可以在将过程分配到不同的项目阶段或位置时，执行基本的变更管理操作。
脚本引擎	Mozilla Rhino JavaScript 引擎提供一种为 Orchestrator 平台创建新的构建块的方法。该脚本引擎包含基本版本控制、变量类型检查、命名空间管理和异常情况处理等增强功能。它可在以下构建块中使用： <ul style="list-style-type: none">■ 操作■ 工作流■ 策略

工作流引擎	<p>通过工作流引擎，您可以捕获业务过程。它使用下列方法之一来逐步实现自动化：</p> <ul style="list-style-type: none"> ■ 库的构建块 ■ 客户提供的构建块 ■ 插件 <p>用户、调度任务或策略均可启动工作流。</p>
策略引擎	<p>使用策略引擎可以监控并生成事件以应对不断变化的情况。策略可以聚合来自平台或任何插件的事件，因此，您可以处理与任何集成技术有关的不断变化的情况。</p>
Web 2.0 前端	<p>Web 2.0 前端可以采用多种新方式表现，具备较强的灵活性。它提供一个用户可自定义的组件库来访问 vCO 耦合对象，并且使用 Ajax 技术动态更新内容，而无需重新加载整个页面。</p>
安全性	<p>Orchestrator 可提供以下高级安全功能：</p> <ul style="list-style-type: none"> ■ 公共密钥基础设施 (PKI)，可用于签署和加密在服务器之间导入和导出的内容。 ■ 数字权限管理 (DRM)，可用于控制导出内容的查看、编辑和重新分配方式。 ■ 安全套接字层 (SSL)，可用于加密桌面客户端和服务器之间的通信和通过 HTTPS 访问 Web 前端。 ■ 高级访问权限管理，可用于控制对过程和由这些过程操作的对象的操作权限。

Orchestrator 用户角色和相关任务

vCenter Orchestrator 会根据以下这三种全局用户角色的特定职责提供不同的工具和界面：管理员、开发人员和最终用户。

管理员	<p>此角色对所有的 Orchestrator 平台功能都具有完全的访问权限。基本的管理任务包括：</p> <ul style="list-style-type: none"> ■ 安装和配置 Orchestrator ■ 管理 Orchestrator 和应用程序的访问权限 ■ 导入和导出软件包 ■ 启用和禁用 Web 视图 ■ 运行工作流和调度任务 ■ 管理导入元素的版本控制
开发人员	<p>此角色的用户拥有访问 Orchestrator 客户端界面的权限并具有以下职责：</p> <ul style="list-style-type: none"> ■ 开发应用程序以扩展 Orchestrator 平台功能 ■ 通过自定义现有工作流和创建新的工作流实现过程自动化 ■ 使用 Web 2.0 技术自定义这些过程的 Web 前端
最终用户	<p>此角色的用户只拥有访问 Web 前端的权限。他们可以运行和调度工作流和策略。</p>

Orchestrator 架构

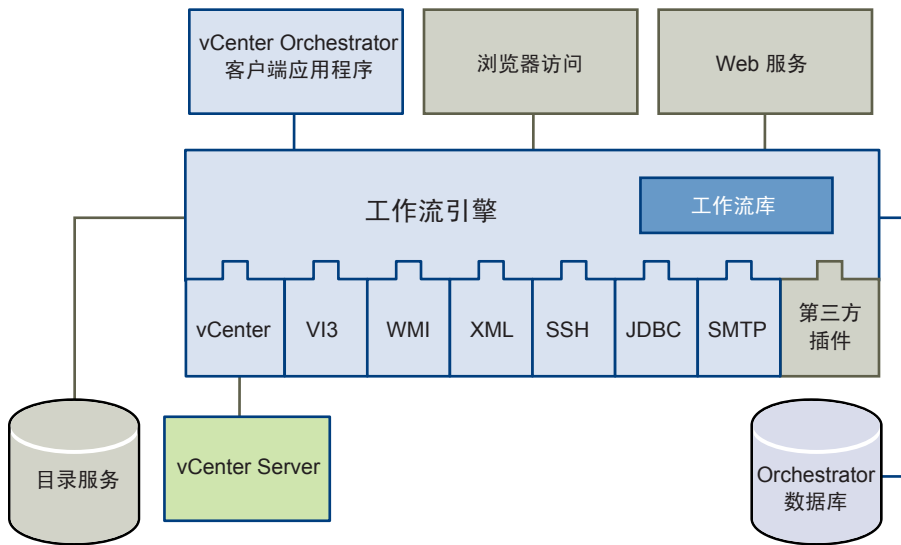
Orchestrator 包含工作流库和工作流引擎，可用于创建和运行自动执行耦合过程的工作流。可以在不同技术对象上运行工作流，Orchestrator 通过一系列插件访问这些技术。

Orchestrator 提供了一组标准插件（包括 VMware vCenter Server 4.0 的插件），使您可以耦合这些插件所在的不同环境中的任务。

此外，Orchestrator 还采用了一个开放式架构，使您可以将外部第三方应用程序插入耦合平台。您可以在由您自己定义的插件技术对象上运行工作流。Orchestrator 可以连接到目录服务服务器以管理用户帐户，此外还可以连接到数据库以存储所运行的工作流的信息。您可以通过 Orchestrator 客户端界面、Web 浏览器或 Web 服务访问 Orchestrator 及其公开的工作流和对象。

图 1-1 显示了 Orchestrator 的架构。

图 1-1 VMware vCenter Orchestrator 的架构



Orchestrator 客户端工作区

Orchestrator 客户端是一款易于使用的桌面应用程序，可用于执行日常管理任务，如导入软件包、运行和调度工作流和策略以及管理用户权限。此外，Orchestrator 客户端还可充当用于创建或自定义工作流的 IDE。

登录 Orchestrator 客户端

要执行常规管理任务，必须登录 Orchestrator 客户端界面。

前提条件

必须配置好 Orchestrator 服务器的所有组件并运行 VMware vCenter Orchestrator Server 服务。

步骤

- 1 访问 Orchestrator 客户端界面。

在 Windows 中，选择**开始 > 程序 > VMware > vCenter Orchestrator Client**。

- 2 使用 Orchestrator 用户名和密码登录。

此时将显示 **My Orchestrator** 视图。此视图汇总了服务器最近的活动，同时还显示已挂起和正在运行的工作流、正在运行的策略、已调度的任务、已完成的工作流和最近编辑过的元素。

下一步

您可以导入软件包、启动工作流或对系统设置根访问权限。

管理工作流

工作流是一系列的操作和决定，这些操作和决定将按顺序运行直到达成特定结果。**Orchestrator** 提供一个根据最佳做法执行常见管理任务的工作流库。此外，**Orchestrator** 还提供由工作流执行的各种操作组成的库。

工作流将操作、决定和结果组合在一起，当按特定顺序执行时，可在虚拟环境中完成特定任务或特定过程。例如，工作流可执行以下任务：置备虚拟机、备份、执行定期维护、发送电子邮件、执行 **SSH** 操作、管理物理基础架构以及其他常规实用程序操作。工作流会根据其功能接受相应信息。您可以创建根据已定义的调度任务运行的工作流，或者也可以创建当发生某些预期事件时运行的工作流。信息可以由您、其他用户、其他工作流或操作或者外部过程（如应用程序的 **Web** 服务调用）提供。工作流在运行之前会执行一些信息验证和筛选操作。

工作流可以调用其他工作流。例如，您可以在几个不同的工作流中重复使用启动虚拟机的工作流。

可以通过使用 **Orchestrator** 客户端界面的集成开发环境 (**IDE**)（提供对工作流引擎的访问）来创建工作流。工作流引擎可以组合来自 **Orchestrator** 的预定义对象和操作库中的工作流。工作流引擎还可以从您插入到 **Orchestrator** 中的外部库中获取对象。通过此功能，您可以自定义过程或实施第三方应用程序提供的功能。

本章讨论了以下主题：

- 第 13 页，““**Workflows**” 视图”
- 第 14 页，“工作流库中的标准工作流”
- 第 15 页，“工作流的主要概念”
- 第 17 页，“对工作流设置用户权限”
- 第 18 页，“运行工作流”

“**Workflows**” 视图

通过 **Orchestrator** 客户端界面上的 **Workflows** 视图可以访问 **Orchestrator** 工作流库。

通过 **Workflows** 视图，您可以查看有关每个工作流的信息，还可以创建、编辑、运行工作流并与工作流交互。



Orchestrator 客户端使用下列图标标识工作流：

“**Workflows**” 视图组件

Workflows 视图包含一组选项卡，分别显示与特定工作流相关的各类信息。

General	显示有关工作流的常规信息，包括名称、版本号、权限、描述以及工作流的全局属性列表。
Inputs	列出工作流运行时所需的全部输入参数。

Outputs	列出 workflows 运行时返回的参数。
Schema	显示 workflows 的图形表示。单击架构中的元素可在 Workflows 视图的下半部分显示该元素的相关信息。
Presentation	构建用户在运行 workflows 时看到的输入参数对话框。可以定义要在对话框中显示的输入参数组，并提供相关描述以便帮助用户提供正确的参数。此外，还可以定义任意参数属性或限制。
Parameters Reference	在一个视图中显示所有的输入和输出参数，并显示在 Presentation 选项卡中定义的相应描述和限制。该选项卡还可以标识使用或生成参数的架构元素。通过单击 Show Attributes 可以有选择性地查看此选项卡中的 workflows 属性。
Executions	提供有关特定 workflows 的不同运行的详细信息。这些信息包括该 workflows 的运行状态、将其启动的用户及其开始和结束的时间和日期。
Events	提供 workflows 运行期间发生的每个事件的相关信息。这些信息包括事件的运行状态、将其启动的用户及其开始和结束的时间和日期。该信息存储在 Orchestrator 数据库的 VM0_LogEvent 表中。
Permissions	列出授予用户或用户组以用于与 workflows 交互的权限。可能显示的权限包括“查看”、“执行”、“检查”、“编辑”和“管理”。

工作流库中的标准工作流

Orchestrator 提供了一个标准工作流库，可用于执行与 vCenter Server 管理有关的许多常见操作。您可以在不对其进行更改的情况下使用这些工作流，也可以对其进行扩展或自定义以满足您自身的需求。工作流库的内容可以通过 Orchestrator 客户端中的 **Workflows** 选项卡访问。

Orchestrator 提供的工作流库包含以下标准工作流类别。

JDBC	通过使用随 Orchestrator 附带的 JDBC (Java 数据库连接) 插件测试工作流和数据库之间的通信。
锁定	演示自动过程的锁定机制，该机制允许工作流锁定其使用的资源。
邮件	通过工作流发送和接收邮件。
Orchestrator	自动执行某些常见的 Orchestrator 操作。
重构	重构您创建的随 VMware Infrastructure 3.5 运行的工作流和应用程序，以便它们能够随 vCenter Server 4.0 运行。
SSH	实施 Secure Shell v2 (SSH-2) 协议。通过这些工作流，您可以使用密码和基于公用密钥的身份验证来实施远程命令和文件传输会话。使用 SSH 配置，您可以指定要在 Orchestrator 清单中公开的对象的路径。
vCenter	可用于访问 vCenter Server 4.0 Server API 的功能，以便将 vCenter Server 的所有功能合并到使用 Orchestrator 自动执行的管理过程中。
XML	可以在自动过程中实施的文档对象模型 (DOM) XML 分析程序。

工作流的主要概念

工作流由操作、属性、参数和架构组成。Orchestrator 会在工作流每次运行时记录一个工作流令牌，以便记录该特定工作流运行的详细信息。

工作流属性

工作流属性充当整个工作流的全局常量和全局变量。工作流元素会处理作为输入参数接收到的数据，并将生成的输出结果设置为工作流属性。

只读工作流属性充当工作流的全局常量。可写属性则充当工作流的全局变量。

工作流属性包含以下内容：

- 只读标志
- 名称
- 类型
- 值
- 链接
- 描述

您使用属性在工作流元素之间传递变量。可以通过以下方式获取属性：

- 在创建工作流时定义属性
- 将工作流元素的输出参数设置为工作流属性
- 从配置元素继承属性

工作流参数

工作流会在其运行时接收输入参数并生成输出参数。

输入参数

输入参数是一种运行时参数，用户、应用程序或其他工作流或操作会将其传递给某个工作流或操作，以便其在启动或运行时得到处理。

输入参数包含以下内容：

- 名称
- 类型
- 描述

将一个输入参数的值传递到工作流后，您将无法更改该参数的名称、类型或描述。

输出参数

工作流或操作的输出参数表示该工作流或操作的运行结果。输出参数会在工作流或工作流元素运行时发生变化。在它们运行时，工作流或操作可以接收其他工作流和操作的输出参数作为其输入参数。

workflow 架构

workflow 架构是 workflow 的图形表示，它采用互连的 workflow 元素流程图形式来显示 workflow。

查看 workflow 架构

在 Orchestrator 客户端中，您可以在某个 workflow 的架构选项卡中查看该流程的 workflow 架构。

有关架构元素和创建并编辑 workflow 架构的信息，请参见《vCenter Orchestrator 开发指南》。

前提条件

您必须获得 **检查** 特权才能查看架构和脚本。

步骤

- 1 在 Orchestrator 客户端中单击 **Workflows** 视图。
- 2 导航到 workflow 层次结构列表中的某个 workflow。
- 3 单击该 workflow。

有关该 workflow 的信息将显示在右窗格中。

- 4 在右窗格中选择 **Schema** 选项卡。

即会看到该 workflow 的图形表示。

下一步

您可以将架构元素从左边的 workflow 选项板拖到 workflow 架构图中，以便编辑该 workflow 架构。

workflow 令牌

workflow 令牌表示正在运行的或已经运行的 workflow。

workflow 是对过程的抽象描述，它用于定义通用步骤顺序以及所需的通用输入参数集。当使用一组真实输入参数运行某个 workflow 时，您将获得此抽象 workflow 的实例，该实例的执行方式基于您给定的具体输入参数。这个特定的运行实例或正在运行的 workflow 实例即称为 workflow 令牌。

workflow 令牌属性

workflow 令牌属性是 workflow 令牌运行时使用的特定参数。它是 workflow 的全局属性以及运行 workflow 令牌时使用的特定输入和输出参数的结合体。

workflow 令牌状况

每次运行 workflow 时，就会有一个 workflow 令牌作为 workflow 层次结构列表中的新的叶节点出现在该 workflow 下。通过单击层次结构列表中的 workflow 令牌，即可在右窗格中显示表示该 workflow 令牌信息的选项卡。

显示的信息包含该 workflow 的架构图、事件列表、workflow 令牌运行时使用的输入参数列表以及该 workflow 的日志文件。如果在其运行时单击 workflow 令牌，可看到选项卡中的信息正在实时更新。

[表 3-1](#) 介绍了 workflow 令牌的状况。

表 3-1 工作流令牌状况

状况	图标	描述
正在运行		工作流令牌正在运行。
正在等待用户交互		工作流令牌在其等待来自用户交互、外部应用程序或其他工作流的输入参数时处于挂起状态。
正在等待事件或定时器		工作流令牌在等待来自外部触发或计时器的信号以继续运行之前处于挂起状态。长时间运行的工作流会在等待信号以继续运行时进入此状况。等待期间，该工作流的线程将变为被动状态。
已取消		工作流令牌已被用户、外部应用程序或其他工作流取消。
失败		工作流令牌失败。
已完成		工作流令牌运行成功。但是，如果工作流定义中包含错误处理，则已完成的工作流令牌在其运行时可能会遇到错误。

对工作流设置用户权限

您可以对工作流设置不同级别的权限，以便限制不同用户或用户组对该工作流的访问。

从 Orchestrator LDAP 服务器的用户和用户组中选择要为其设置权限的不同用户和用户组。Orchestrator 定义了可对用户或用户组应用的权限级别。

查看	用户可以查看工作流中的元素，但无法查看架构或脚本。
检查	用户可以查看工作流中的元素，包括架构和脚本。
执行	用户可以运行工作流。
编辑	用户可以编辑工作流。
管理	用户可以对工作流设置权限。

重要事项 在生产环境中，应当将编辑工作流的权限仅授予 Orchestrator 管理员。允许其他用户编辑生产环境中的工作流可能会导致意外的工作流语义更改。

前提条件

您必须创建一个工作流，并将其打开以便在工作流工作台中编辑，此外还必须为其添加必要的元素。

步骤

- 1 在工作流工作台中单击 **Permissions** 选项卡。
- 2 单击 **Add access rights** 链接以为新用户或用户组定义权限。
- 3 通过在 **Search** 文本框中输入文本来搜索用户或用户组。
搜索结果会显示 Orchestrator LDAP 服务器中与搜索条件相匹配的所有用户和用户组。
- 4 选择用户或用户组并单击 **OK**。
- 5 右键单击用户并选择 **Add access rights**。
- 6 选中相应的复选框以为此用户设置权限级别，然后单击 **OK**。
权限级别不具累积性。要授予用户查看工作流、检查架构和脚本、运行和编辑工作流以及更改权限的权限，您必须选中所有的复选框。
- 7 单击 **Save and Close** 退出软件包编辑器。
您已对工作流设置了适当的用户权限。

运行工作流

运行标准的 Orchestrator 工作流可自动执行与虚拟环境管理相关的频繁重复过程，例如在与 Orchestrator 相连的 vCenter Server 中创建虚拟机。

前提条件

您必须配置 vCenter 插件。有关详细信息，请参见《vCenter Orchestrator 安装和配置指南》。

步骤

- 1 在 Orchestrator 客户端中单击 **Workflows** 视图。
- 2 选择 **Library > vCenter > Virtual Machine Management > Basic** 以导航到 **Create VM (Simple)** 工作流。
- 3 右键单击 **Create VM (Simple)** 工作流并选择 **Execute Workflow**。
此时将打开“input parameters”对话框。
- 4 输入下列信息以创建虚拟机。

选项	操作
VM name	输入虚拟机的名称。例如， orchestrator-test
VM folder	<ol style="list-style-type: none"> a 单击 Not Set 链接。 b 不要在 Search 文本框中输入任何文本，并按 Enter。 选择框将列出其基础架构包含的类型为 VC:VmFolder 的所有对象。 c 选择一个 VC:VmFolder 对象并单击 Select。
Size of the new disk in GB	输入合适的数值。
Memory size in MB	输入合适的数值。
Number of virtual CPUs	从 Number of virtual CPUs 下拉菜单中选择合适的 CPU 个数。
VM guest OS	单击 Not Set 链接并从列表中选择客户机操作系统。
Host on which VM will be created	单击 Not Set 链接并从推荐的列表选择一个主机。
Resource pool	<ol style="list-style-type: none"> a 单击 Not Set 链接。 b 在 vCenter Server 基础架构层次结构中导航到资源池。 c 单击 Select。

选项	操作
Network to connect to	单击 Not Set 链接并从列表中选择 VC:Network 对象。
Datastore on which the VM will be created	单击 Not Set 链接并从列表中选择 VC:Datastore 对象。

- 5 单击 **Submit** 以运行工作流。
工作流令牌将出现在 **Create VM (Simple)** 工作流下的叶节点上。该令牌显示工作流正在运行图标。
- 6 单击工作流令牌可查看工作流运行时的状态。
- 7 在工作流令牌视图中单击 **Events** 选项卡，按照工作流令牌的过程执行操作，直到该过程完成。
- 8 在 Orchestrator 客户端中单击 **Inventory** 视图。
- 9 在 vCenter Server 基础架构层次结构中导航到在 [步骤 4](#) 中定义的资源池。
orchestrator-test 虚拟机已出现在该资源池中。

Create VM (Simple) 工作流运行成功。

下一步

您可以登录 vSphere Client 并管理这一新虚拟机。

管理操作

操作代表您在工作流、Web 视图和脚本中用作构建块各个函数的函数。操作是具有多个输入参数和一个返回值的 JavaScript 函数。操作可以在 Orchestrator API 中的任何对象上调用，也可以在通过插件导入到 Orchestrator 中的任何 API 中的对象上调用。

当工作流运行时，操作会从该工作流的属性中获取输入参数。这些属性可以是工作流的初始输入参数，也可以是工作流中其他元素在其运行时设置的属性。

如果独立于调用操作的工作流单独定义操作，即可更轻松地更新或优化这些操作。还可以通过定义单个操作来使其他工作流重新使用操作。

本章讨论了以下主题：

- [第 21 页](#)，““Actions” 视图”
- [第 22 页](#)，“创建操作”
- [第 22 页](#)，“复制操作”
- [第 23 页](#)，“导出操作”
- [第 23 页](#)，“导入操作”
- [第 23 页](#)，“移动操作”
- [第 24 页](#)，“查找执行操作的元素”

“Actions” 视图

通过 Orchestrator 客户端界面中的 **Actions** 视图可以访问预定义操作库。在 **Actions** 视图中，您可以复制操作、将其导出到其他工作流或软件包或将其移到操作层次结构列表中的其他类别下。

通过展开操作层次结构列表中的节点，可以浏览可用操作。选择列表中的某个操作时，右窗格中会显示有关该操作的详细信息。

Actions 视图显示以下选项卡。

General	显示有关操作的常规信息，包括名称、版本号、用户可以执行的操作及其描述。
Scripting	显示操作的返回类型、输入参数和用于定义操作函数的 JavaScript 代码。
Events	显示此操作遇到或触发的所有事件。
Permissions	显示哪些用户和用户组具有此操作的访问权限。

创建操作

将一个函数定义为操作而不是将其直接编码为可编脚本任务工作流元素时，可在库中公开该操作以供其他工作流使用。

步骤

- 1 在 Orchestrator 客户端中单击 **Actions** 视图。
- 2 展开操作层次结构列表的根节点，并导航到您要在其中创建操作的模块。
- 3 右键单击该模块并选择 **Add action**。
- 4 在文本框中输入操作的名称，然后单击 **OK**。
- 5 右键单击操作并选择 **Edit**。
- 6 单击 **Scripting** 选项卡。
- 7 要更改默认的返回类型，请单击 **void** 链接。
- 8 通过单击箭头图标添加操作的输入参数。
- 9 编写操作脚本。
- 10 单击 **Save and close**。

您的自定义操作即会添加到操作库中。

下一步

此时便可以在工作流中使用新的自定义操作了。

复制操作

可以复制任何预定义操作并在脚本中重用它们。

步骤

- 1 在 Orchestrator 客户端中单击 **Actions** 视图。
- 2 展开操作层次结构的根节点，并导航到要复制的操作。
- 3 右键单击操作并选择 **Duplicate action**。
- 4 输入新操作的名称。
如果没有在此文本框中输入值，则将在原操作名称后附加一个编号。
- 5 对于 **Action module** 的值，选择要添加新操作的模块。
- 6 （可选） 如果不想复制版本历史记录，请选择 **No**。
导入操作时，系统会将其版本与本地内容的版本相比较，以便管理员可以决定是否导入。
- 7 选择 **Duplicate**。

新操作将出现在操作库中。

下一步

此时便可以在工作流中使用该操作了。

导出操作

可以导出操作以在其他 workflow、策略或 Web 视图中重用。

步骤

- 1 在 Orchestrator 客户端中单击 **Actions** 视图。
- 2 展开操作层次结构列表的根节点，并导航到要导出的操作。
- 3 右键单击该操作并选择 **Export action**。
- 4 (可选) 选择 **Encrypt content with name** 选项以加密导出的文件。

其他系统可以导入并运行此加密文件，但导入者不能编辑该文件。加密的文件内容为只读。

- 5 选择要在其中保存操作文件的位置并单击 **Save**。

操作即会导入到本地文件中。

下一步

此时便可以导入该操作并在 workflow 和脚本中使用它了。

导入操作

可以导入操作，并在 workflow、Web 视图和脚本中将其用作构建块。

步骤

- 1 在 Orchestrator 客户端中单击 **Actions** 视图。
- 2 展开操作层次结构列表的根节点，并导航到您要将操作导入其中的模块。
- 3 右键单击该模块并选择 **Import action**。
- 4 选择带 **.action** 扩展名的文件并单击 **Open**。

导入的操作即会显示在操作库中。

下一步

此时便可以在 workflow 和脚本中使用该操作了。

移动操作

要想重新排序操作层次结构列表中的操作或以其他方式组织脚本，可将操作移动到另一个模块中。

步骤

- 1 在 Orchestrator 客户端中单击 **Actions** 视图。
- 2 展开操作层次结构的根节点，并导航到要重定位的操作。
- 3 右键单击操作并选择 **Move this action**。
- 4 选择要在其中保存操作文件的位置并单击 **Save**。

此操作即会移动到新模块中。



小心 操作将根据操作模块名称和操作名称引用。请确保引用此操作的所有元素在移动该操作之后仍有效。

下一步

查找执行重定位操作的所有工作流和软件包。

查找执行操作的元素

如果对操作进行编辑时更改了它的行为，则有可能会不慎中断执行该操作的工作流或应用程序。**Orchestrator** 提供了一项功能，可查找执行某个给定元素的所有操作、工作流或软件包。您可以检查修改元素是否会影响其他元素的操作。

重要事项 **Find Elements that Use this Element** 功能会检查所有软件包、工作流和策略，但不会检查脚本。因此，修改操作可能会影响在 **Find Elements that Use this Element** 功能无法识别的某个脚本中调用此操作的元素。

步骤

- 1 在 **Orchestrator** 客户端中单击 **Actions** 视图。
- 2 展开操作层次结构列表的节点以导航到给定的操作。
- 3 右键单击该操作并选择 **Find Elements that Use this Element**。
此时将打开一个对话框，其中显示执行此操作的所有元素，如工作流或软件包。
- 4 双击结果列表中的某个元素以在 **Orchestrator** 客户端中显示该元素。
您已找到执行给定操作的所有元素。

下一步

可以检查修改此元素是否会影响其他任何元素。

管理任务

可以使用任务来调度一次工作流，或者采用重复执行的方式来多次调度该工作流。

本章讨论了以下主题：

- 第 25 页，““Tasks” 视图”
- 第 25 页，“创建任务”
- 第 26 页，“编辑任务重复执行方式”

“Tasks” 视图

Orchestrator 客户端中的 **Tasks** 视图显示系统中所有已调度工作流的列表。这些工作流按名称或日期排序，并显示其状态。可以使用 **Tasks** 视图创建、编辑、挂起、恢复和取消已调度的工作流。

在列表中选择任务后，**Tasks** 视图会显示下列选项卡，这些选项卡显示有关特定任务的详细信息。

General	显示有关任务的常规信息，包括名称、启动行为、描述、开始日期、启动用户、已调度工作流的名称以及工作流的输入值列表。
Recurrence	显示有关任务重复执行方式的详细信息。
Executions	显示有关特定已调度工作流的不同运行的详细信息。此信息包括工作流的运行状态、开始和结束日期以及启动用户。取消任务时，其日志信息将从系统中删除。挂起任务时，日志信息仍保留。
Permissions	显示授予用户或用户组以用于与工作流交互的权限。可能显示的权限包括“查看”、“执行”、“检查”、“编辑”和“管理”。

创建任务

可以从 Orchestrator 客户端的 **Tasks** 或 **Workflows** 视图调度工作流。用于启动工作流的用户凭据与用于调度任务的凭据相同。

前提条件

您必须获得**执行**权限才能创建任务。

步骤

- 1 在 Orchestrator 客户端中单击 **Tasks** 视图。
- 2 从下拉菜单中选择 **Create task**。
- 3 （可选）选择 **Create task as** 可使用其他用户的凭据调度任务。
- 4 搜索要调度的工作流。

- 5 右键单击工作流，然后单击 **Select**。
此时将打开“input parameters”对话框。
- 6 单击 **Execution date and time** 值的 **Not set** 链接。
将显示一个日历。
- 7 选择工作流的开始日期和时间，然后单击 **OK**。
- 8 从 **Recurrence** 下拉菜单中选择重复执行方式。
- 9 （可选）单击 **Recurrence end date** 值的 **Not Set** 链接，设置工作流的结束时间和日期。
- 10 在“input parameters”对话框中输入必要的信息。
- 11 单击 **Submit** 以调度工作流。
已调度工作流已列在 **Tasks** 视图中。

下一步

可以监控已调度工作流，并在任务完成后将任务删除。

编辑任务重复执行方式

重复执行方式用于指定给定工作流的调度方式。可以在 **Tasks** 视图中编辑任务的重复执行方式。

前提条件

您必须已创建一个重复执行的任务。

步骤

- 1 在 Orchestrator 客户端中单击 **Tasks** 视图。
- 2 右键单击您想要编辑其重复执行方式的已调度任务，然后选择 **Edit**。
- 3 单击 **Recurrence** 选项卡。
- 4 从下拉菜单中选择重复执行方式。
显示内容会根据所选方式而更改。
- 5 可以将无数条目添加到此方式中。
可以编辑每个条目。
- 6 完成后请单击 **Save and close**。

已调度工作流的新重复执行方式即会显示在 **Recurrence** 选项卡中。

下一步

可以在 **Executions** 选项卡上查看有关已调度工作流的不同运行的详细信息。

策略

策略是用于监控系统活动的事件触发器。策略会在响应某些已定义对象的状态或性能更改时运行预定义的事件。策略是一系列的规则、量表、阈值和事件过滤器，可在具体的预定义事件发生时运行某些特定工作流。只要策略处于运行状态，Orchestrator 就会持续评估该策略的规则。您可以执行策略量表和阈值，这些量表和阈值用于监控 VC:HostSystem 和 VC:VirtualMachine 类型的 vCenter Server 对象的行为。

Orchestrator 定义了以下策略类型：

策略模板

主策略。策略模板并不与真实的对象关联。它们是抽象的规则集，用于定义当发生某一特定抽象事件时要执行的行为。您可以在 Orchestrator 客户端的 **Policy Templates** 视图中查看现有策略模板并创建模板。

策略

策略是模板实例，或是与真实对象相关联的并由实际存在的事件触发的独立事件触发器实例。此外，您不使用模板也可以创建策略。您可以在 Orchestrator 客户端的 **Policies** 视图中查看现有策略并创建策略。

您也可以将策略模板组织成不同的类别，以便于导航。

使用软件包

软件包是将内容从一台 **Orchestrator** 服务器传输到另一台服务器时所使用的载体。软件包可以包含 workflow、操作、策略、Web 视图、配置或资源。

将一个元素添加到软件包中时，**Orchestrator** 会检查其依赖性并将所有相关元素都添加到软件包中。例如，如果添加了一个使用多步操作或其他 workflow 的 workflow，**Orchestrator** 会将这些操作和 workflow 添加到软件包中。

当您导入软件包时，服务器会将其中所包含的不同元素的版本与相匹配的本地元素相比较。比较结果将显示本地元素和导入的元素之间的版本差异。管理员可以决定是导入该软件包还是选择导入某些特定元素。

软件包通过数字权限管理功能来控制接收服务器使用软件包内容的方式。**Orchestrator** 会给软件包签名并对其进行加密以便保护其数据。软件包使用 X509 证书来监控导出和重新分配元素的用户。

本章讨论了以下主题：

- [第 29 页](#)，“**Packages**”视图”
- [第 30 页](#)，“创建软件包”
- [第 30 页](#)，“对软件包设置用户权限”
- [第 31 页](#)，“导出软件包”
- [第 32 页](#)，“导入软件包”
- [第 32 页](#)，“获取并同步远程软件包”

“Packages” 视图

通过 **Orchestrator** 客户端界面中的 **Packages** 视图，您可以对软件包执行添加、导入、导出和同步操作。

Packages 视图包含一组选项卡，分别显示与特定软件包相关的各类信息。在软件包编辑模式下，您可以在每个选项卡上插入和移除元素。

General	显示有关软件包的常规信息，包括名称、合法所有者和描述。
Workflows	显示所选软件包包含的所有 workflow。
Policies	显示与所选软件包中的 workflow 相关的策略模板。
Actions	显示在所选软件包的 workflow 中使用的操作。
Web View	显示所选软件包包含的 Web 视图。
Configurations	显示软件包中包含的配置元素。
Resources	显示所选软件包中嵌入的外部资源。

Used plug-ins	显示与所选软件包关联的插件相关的信息。插件可以有一个或多个软件包与其关联。
Permissions	显示授予用户或用户组以用于与软件包交互的权限。可能显示的权限包括“查看”、“执行”、“检查”、“编辑”和“管理”。

创建软件包

您可以采用软件包方式导出 workflow、策略、操作、插件、资源、Web 视图和配置元素。

前提条件

您必须具有要添加到软件包中的元素（如 workflow、操作和策略）。

步骤

- 1 在 Orchestrator 客户端中单击 **Packages** 视图。
- 2 在 **Packages** 层次结构列表的标题栏中单击菜单按钮，然后选择 **Add package**。
- 3 在打开的对话框中提供软件包名称，然后单击 **OK**。
软件包的命名约定为 `<域. 贵公司>.category.<软件包名称>`。例如，`com.vmware.mycategory.mypackage`。
- 4 右键单击软件包并选择 **Edit**。
此时将打开软件包编辑器。
- 5 在 **General** 选项卡中添加该软件包的描述。
- 6 单击 **Workflows** 选项卡以将 workflow 添加到软件包中。
 - 单击 **Insert Workflows (list search)** 以搜索并在选取对话框中选择所需的工作流。
 - 单击 **Insert Workflows (tree browsing)** 以浏览并在层次结构列表中选择所需的工作流。
- 7 单击 **Policies**、**Actions**、**Configurations** 和 **Resources** 选项卡以将策略模板、操作、配置元素和资源元素添加到软件包中。
- 8 在 **Web View** 选项卡中单击 **Insert Webview** 以将 Web 视图添加到软件包中。
- 9 在 **Used plug-ins** 选项卡中单击 **Insert used plug-in** 以将插件添加到软件包中。
您已将所需的元素添加到软件包。

下一步

您必须为此软件包设置用户权限。

对软件包设置用户权限

您可以对软件包设置不同级别的权限，以便限制不同用户或用户组对该软件包的访问。

从 Orchestrator LDAP 服务器的用户和用户组中选择要为其设置权限的不同用户和用户组。Orchestrator 定义了可对用户或用户组应用的权限级别。

查看	用户可以查看软件包中的元素，但无法查看架构或脚本。
检查	用户可以查看软件包中的元素，包括架构和脚本。
执行	用户可以运行软件包中的元素。
编辑	用户可以编辑软件包中的元素。
管理	用户可以对软件包中的元素设置权限。

前提条件

您必须创建一个软件包，并将其打开以便在软件包编辑器中编辑，此外还必须为其添加必要的元素。

步骤

- 1 在软件包编辑器中单击 **Permissions** 选项卡。
- 2 单击 **Add access rights** 链接为新用户或用户组定义权限。
- 3 通过在 **Search** 文本框中输入文本来搜索用户或用户组。
搜索结果会显示 Orchestrator LDAP 服务器中与搜索条件相匹配的所有用户和用户组。
- 4 选择用户或用户组并单击 **OK**。
- 5 右键单击用户并选择 **Add access rights**。
- 6 选中相应的复选框为此用户设置权限级别，然后单击 **OK**。
权限级别不具叠加性。要授予用户查看元素、检查架构和脚本、运行和编辑元素以及更改权限的权限，您必须选中所有的复选框。
- 7 单击 **Save and Close** 退出软件包编辑器。
您已创建软件包并设置了适当的用户权限。

导出软件包

可以导出软件包并在其他 Orchestrator 服务器上重新使用其内容。系统会为导出的软件包中包含的所有元素添加证书。将软件包导入到其他服务器中时，这些证书也会一起随之导入。

前提条件

必须已创建一个软件包并向其添加了必要的元素。

步骤

- 1 在 Orchestrator 客户端中单击 **Packages** 视图。
- 2 右键单击要导出的软件包并选择 **Export package**。
- 3 浏览以选择保存软件包的位置，然后单击 **Open**。
- 4 (可选) 单击 **Add target certificate** 以为该软件包签名。
 - a 在证书列表中，选择要用于导出的软件包的证书。
 - b 单击 **Select**。
- 5 (可选) 要对导出的软件包施加限制，请取消选择以下任意选项。

选项	描述
View content	如果选择该选项，则软件包导入者将允许查看此软件包中包含的元素的 JavaScript。
Re-Packageable	如果选择该选项，则软件包导入者将允许重新分配此软件包中包含的元素。
Edit element	如果选择该选项，则软件包导入者将允许修改此软件包中包含的元素。

- 6 (可选) 如果您不想导出此软件包的版本历史记录，则取消选中 **Export version history** 复选框。
- 7 单击 **Save**。
软件包已导出。

下一步

此时便可以在新的 Orchestrator 服务器上使用导出的软件包中的所有 workflow、操作、策略和 Web 视图了。

导入软件包

要在一台 Orchestrator 服务器上重新使用另一台服务器中的 workflow、操作、策略和 Web 视图，可将它们作为软件包导入。

重要事项 Orchestrator 3.2 生成的软件包可向上兼容 Orchestrator 4.0。可以将 Orchestrator 3.2 服务器中的软件包导入到 Orchestrator 4.0 服务器中。Orchestrator 4.0 中的软件包不向后兼容 Orchestrator 3.2。因此，无法将 Orchestrator 4.0 服务器生成的软件包导入到 Orchestrator 3.2 服务器中。

前提条件

在远程服务器上，必须已创建一个软件包并向其添加了必要的元素。

步骤

- 1 在 Orchestrator 客户端中单击 **Packages** 视图。
- 2 从下拉菜单中选择 **Import package**。
- 3 浏览以选择要导入的软件包，然后单击 **Open**。
此时会显示有关导入者的证书信息。
- 4 查看软件包导入详细信息并选择 **Import** 或 **Import and trust provider**。
此时会显示“Import package”视图。如果导入的软件包元素版本高于服务器版本，则系统会选择导入元素。
- 5 （可选）取消选择不希望导入的元素。
- 6 单击 **Import checked elements**。
导入的软件包即会显示在软件包列表中。

下一步

此时便可以在 Orchestrator 服务器上导入的软件包中的所有 workflow、操作、策略和 Web 视图作为新的构建块使用。

获取并同步远程软件包

Packages 视图提供了一种方法，可用于将一台 Orchestrator 服务器上的软件包与另一台服务器上的软件包同步。

如果本地服务器上已有一个软件包，可以使用 **Synchronize** 选项。如果要检索来自远程服务器的软件包，可使用 **Get remote package** 选项。

同步软件包是用来确保从远程服务器获取所有元素的唯一方法。如果同步的是多个单独的元素，则 Orchestrator 只会同步本地服务器上已存在的元素。要获取远程服务器中的任何新元素，您必须同步包含这些元素的软件包。

步骤

- 1 在 Orchestrator 客户端中单击 **Packages** 视图。
- 2 右键单击要同步的软件包，然后选择 **Synchronize**。

3 登录服务器。

此时会打开“Orchestrator Synchronization”对话框。它会显示软件包元素之间的差异。

4 查看本地和远程软件包元素之间的比较结果，然后单击 **Synchronize** 并选择一个选项。

选项	描述
none	本地和远程元素的版本号相同。无需同步。
commit	本地元素的版本较高。远程元素将被覆盖。
update	远程元素的版本较高。将更新本地元素。如果本地元素不存在，可以删除远程元素。
merge	本地和远程软件包都将被合并的引用列表所覆盖。引用的元素保持不变。

注意 如果远程服务器无法识别您的证书，则将无法提交元素。

同步的软件包已重新加载。

下一步

此时便可以在 workflow、操作、策略和 Web 视图中使用更新后的软件包内容。

Orchestrator 默认插件

Orchestrator 中包含一组插件。每个插件均向 Orchestrator 平台公开一个外部产品 API。这些插件可提供清单类、通过新的对象类型扩展脚本引擎以及发布来自外部系统的通知事件。每个插件还提供一个工作流库，用于代表集成产品的典型自动化用例。

有关在 Orchestrator 中默认安装的插件列表及其基本功能，请参见表 8-1。

有关可作为加载项使用并且可以单独下载的插件列表，请参见表 8-2。

表 8-1 默认安装的插件

插件	用途	配置	输入类型	脚本对象	清单
vCenter 4.0	可用于访问 vCenter 4.0 Server API，以便将 vCenter Server 的所有功能合并到使用 Orchestrator 自动执行的管理过程中。	请参见《vCenter Orchestrator 安装和配置指南》中的“配置 vCenter 4.0 插件”部分。	请参见《vCenter Orchestrator 开发指南》中的“使用 Orchestrator API”部分。	请参见《vCenter Orchestrator 开发指南》中的“使用 Orchestrator API”部分。	返回所有对象。
vCO 库	提供可充当客户端过程自定义和自动化模板的工作流。工作流库包含可用于生命周期管理、置备、灾难恢复、热备份以及许多其他标准过程的模板。可以访问以及复制和编辑库过程的来源。	无	请参见《vCenter Orchestrator 开发指南》中的“使用 Orchestrator API”部分。	请参见《vCenter Orchestrator 开发指南》中的“使用 Orchestrator API”部分。	不在清单中存储任何内容。
邮件	使用简单邮件传输协议 (SMTP) 发送电子邮件。	要使用的 EmailMessage 对象的默认值。请参见《vCenter Orchestrator 安装和配置指南》中的“配置 SSH 插件”部分。	无	EmailMessage	不在清单中存储任何内容。
SSH	提供 Secure Shell v2 (SSH-2) 协议的实施。允许使用密码和基于公用密钥的身份验证来实施远程命令和文件传输会话。(可选) SSH 插件可提供在 vCO 清单中直接浏览远程文件系统的功能。	请参见《vCenter Orchestrator 安装和配置指南》中的“定义默认 SMTP 连接”部分。	文件 文件夹 根文件夹 SSH 连接	KeyPairManager SSHCommand SSHFile SSHFolder SSHSession	不在清单中存储任何内容。

表 8-1 默认安装的插件（续）

插件	用途	配置	输入类型	脚本对象	清单
WebOperator	一种演示 Web 界面，可用于访问 Orchestrator 库中的工作流并通过网络与其进行交互。	无	无	无	不在清单中存储任何内容。
枚举	允许创建常用的枚举类型。	无	枚举	无	不在清单中存储任何内容。
Net	Jakarta Apache Commons Net Library 的包装器。可提供 Telnet、FTP 和 POP3 的实施。POP3 部分允许读取电子邮件。通过将邮件插件与 Net 插件结合使用，可提供完整的电子邮件收发功能。	无	无	FTPClient POP3Client POP3Message TelnetClient	不在清单中存储任何内容。
XML	可以在自动过程中实施的完整文档对象模型 (DOM) XML 分析程序。	无	无	XMLDocument XMLElement XMLManager XMLNameNodeMap XMLNode XMLNodeList	不在清单中存储任何内容。
数据库	Java 数据库连接 (JDBC) API 独立于数据库，是用于连接 Java 编程语言与广泛的数据库的行业标准。这些数据库包含 SQL 数据库和其他表格式数据源（如电子表格或平面文件）。JDBC API 为基于 SQL 的数据库访问提供了一个调用级 API。	无	无	Connection JDBCConnection PreparedStatement ResultSet	不在清单中存储任何内容。
重构	提供一个工作流集以帮助您重构为 VMware Infrastructure 3.5 构建的工作流和应用程序，以便它们可以在 vCenter Server 4.0 上运行。	无	无	RefactorDescription RefactorProcess	不在清单中存储任何内容。

表 8-2 其他插件

插件	备注	用途	配置	清单
Microsoft	可单独下载	提供对 Windows Management Instrumentation (WMI) 的访问, 并可读写 Microsoft Active Directory。	在 Orchestrator 配置界面中, 从 LDAP 选项卡中导入设置或对 WMI 使用不同的 LDAP 设置。 请参见《vCenter Orchestrator 安装和配置指南》中的“配置 LDAP 设置”部分。	计算机、ForeignSecurityPrincipals、程序数据、系统、用户、域控制器
VMware Infrastructure 3	可单独下载	具备向后兼容性。	请参见《vCenter Orchestrator 安装和配置指南》中的“配置 vCenter 4.0 插件”部分。	返回所有对象。
VMware Infrastructure 3.5	加载项位于 <安装_目录>/extras/plugins。	具备向后兼容性。	请参见《vCenter Orchestrator 安装和配置指南》中的“配置 vCenter 4.0 插件”部分。	返回所有对象。

修改通过插件搜索获得的对象数

默认情况下，使用 Orchestrator 客户端通过插件搜索对象时每次可返回 20 个对象。您可以修改插件配置文件以增加返回的对象数。

前提条件

您必须已在 Orchestrator 服务器上安装了一个插件。

步骤

- 1 导航到 Orchestrator 服务器系统上的插件配置文件夹。

在 Windows 中： <安装_目录>\VMware\Orchestrator\app-server\server\vm\conf\plugins

此文件夹包含安装在 Orchestrator 服务器上的每个插件的 XML 配置文件。

- 2 打开要更改其搜索结果个数的插件的 XML 配置文件。
- 3 将以下行添加到该插件的 XML 配置文件中。

```
<entry key="ch.dunes.database.fetch-limit">50</entry>
```

此行将要返回的搜索结果个数设置为 50。

- 4 保存该 XML 配置文件。
- 5 （可选）对每个要修改的插件重复 [步骤 2](#) 到 [步骤 4](#)。
- 6 重新启动 Orchestrator 服务器。

此时 Orchestrator 对某个特定插件显示的搜索结果个数已增加。

使用 JavaScript 访问 Java 类

默认情况下，Orchestrator 会限制 JavaScript 只能访问一组有限的 Java 类。如果您需要 JavaScript 能够访问更大范围的 Java 类，则必须设置一个 Orchestrator 系统属性以授予此访问权限。

允许 JavaScript 引擎对 Java 虚拟机 (JVM) 具有完全访问权限会导致潜在的安全问题。有缺陷或恶意的脚本可能有权访问运行 Orchestrator 服务器的用户能够访问的所有系统组件。因此，默认情况下，Orchestrator JavaScript 引擎只能访问 `java.util.*` 软件包中的类。

如果您需要 JavaScript 对除 `java.util.*` 软件包以外的类具有访问权限，可在一个配置文件中列出要对其授予 JavaScript 访问权限的 Java 软件包。然后，将 `com.vmware.scripting.rhino-class-shutter-file` 系统属性设置为指向此文件。

步骤

- 1 创建一个文本配置文件以存储要对其授予 JavaScript 访问权限的 Java 软件包列表。

例如，要授予 JavaScript 对 `java.net` 软件包中的所有类以及 `java.lang.Object` 类的访问权限，请将以下内容添加到文件中。

```
java.net.*
java.lang.Object
```

- 2 使用适当的名称将配置文件保存到适当的位置。
- 3 打开 `vmo.properties` 系统属性文件。

`vmo.properties` 文件位于以下位置：

- `<安装-目录>\VMware\Orchestrator\app-server\server\vmo\conf`（如果安装的是 Orchestrator 独立版本）。
- `<安装-目录>\VMware\Infrastructure\Orchestrator\app-server\server\vmo\conf`（如果使用 vCenter Server 安装程序安装 Orchestrator）。

- 4 通过将以下行添加到 `vmo.properties` 文件中，设置 `com.vmware.scripting.rhino-class-shutter-file` 系统属性。

```
com.vmware.scripting.rhino-class-shutter-file=您的配置文件的路径
```

- 5 保存 `vmo.properties` 文件。
- 6 重新启动 Orchestrator 服务器。

JavaScript 引擎将对您指定的 Java 类拥有访问权限。

维护和恢复

使用 Orchestrator 配置界面上的 **Troubleshooting** 选项卡，您可以执行与工作流和任务相关的多种批量操作。您可以使用 **Troubleshooting** 选项卡全局重置服务器以及移除以前运行的所有执行痕迹。

表 11-1 列出了可能的批量操作。

重要事项 在单击故障排除选项之前，请确保 vCO 服务器已停止运行。

表 11-1 故障排除选项

操作	描述
Cancel all running workflows	在数据库中将所有正在运行的工作流标记为取消，从而防止服务器在下次重新引导时重新启动这些工作流。让 Orchestrator 退出无限循环。
Delete all workflow executions	删除所有包含数据库脚本的令牌。
Suspend all scheduled tasks	取消所有已调度的任务，但并不停止运行或移除与其关联的工作流。
Clean all server temporary files	清除 JBoss 服务器使用的所有临时文件，以确保服务器持续运行。JBoss 服务器是一个应用程序服务器，它是 Orchestrator 服务器的基础。
Force plug-in reinstallation when server starts	请在插件发生更改时使用此操作，以确保服务器下次启动时正确更新该插件。 注意 如果您在配置和安装默认的插件后更改了 Orchestrator 数据库，则必须强制重新安装插件。强制重新安装插件时会删除用于保存已安装插件版本的 <安装_目录>\app-server\server\vm\plugins_VSOPluginInstallationVersion.xml 文件，并强制重新安装插件。插件将以其原始内容重新安装。所有更改都将丢失。

更改服务器日志的大小

如果服务器日志在一天内多次重新生成，则很难确定导致问题发生的原因。为了防止出现这一情况，您可以更改服务器日志的默认大小。服务器日志的默认大小为 5MB。

步骤

- 1 找到 <安装_目录>\app-server\server\vmo\conf\log4j.xml 文件。
- 2 在 log4j.xml 文件中，找到以下代码块：

```
<appender class="org.jboss.logging.appender.RollingFileAppender" name="FILE">
<errorHandler class="org.jboss.logging.util.OnlyOnceErrorHandler"/>
<param name="File" value="${jboss.server.home.dir}/log/server.log"/>
<param name="Append" value="true"/>

<!-- Rollover at 5MB and allow 4 rollover files -->
<param name="MaxFileSize" value="5120KB"/>
<param name="MaxBackupIndex" value="4"/>

<layout class="org.apache.log4j.PatternLayout">
<!-- The default pattern:Date Priority [Category] Message\n -->
<param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss.SSSZ} %-5p [%c{1}] %m%n"/>
</layout>
</appender>
```

- 3 更改以下各行：

```
<param name="MaxFileSize" value="5120KB"/>
<param name="MaxBackupIndex" value="4"/>
```

`MaxFileSize` 参数用于控制日志文件的大小，`MaxBackupIndex` 参数则用于控制滚动更新的文件个数。

注意 在保存之前，请确保该文件不存在文字错误。如果文件包含文字错误，则日志将会丢失。

系统会动态读取此文件。无需重新引导服务器。

索引

A

“Actions” 视图 21
安全性 7

B

版本管理 7
标准工作流 14

C

参数 15
操作
 查找元素执行 24
 重定位 23
 创建 22
 导出 23
 导入 23
 复制 22
 加密 23
 添加 22
 移动 23
 引用 23
策略 27
策略模板 27
策略引擎 7
插件, 搜索 39
持久性 7
重复执行的任务 26

F

服务器日志 44

G

工作流
 标准 14
 重复 26
 调度 25
 库 14
 权限 17
 图标 13
 运行 18
工作流参数 15
工作流令牌 16
工作流令牌属性 16
工作流令牌状况 16

工作流属性 15
工作流引擎 7
故障排除选项
 重置当前版本 43
 挂起任务 43
 清除目录 43
 取消执行 43
 移除执行 43

J

JavaScript 41
架构 15, 16
检查点 7
脚本
 访问 Java 类 41
 shutter 系统属性 41
脚本引擎 7

L

令牌 15

O

Orchestrator API 21
Orchestrator 架构 9
Orchestrator 客户端, 登录 11

P

“Packages” 视图 29
配置元素 15

R

任务
 创建 25
 添加 25
软件包
 创建 30
 导出 31
 导入 32
 获取 32
 签名 29, 31
 权限 30
 数字权限管理 29
 同步 32
 限制重新使用 31

S

输出参数 15

输入参数 15

属性 15

T

“Tasks” 视图 25

图标, workflow 13

W

我的 Orchestrator 视图 11

“Workflows” 视图 13

X

系统属性 41

Y

用户角色 8