Using VMware vFabric Application Director

vFabric Application Director 5.0

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see http://www.vmware.com/support/pubs.
You can find the most up-to-date technical documentation on the VMware Web site at:
http://www.vmware.com/support/
The VMware Web site also provides the latest product updates.
If you have comments about this documentation, submit your feedback to:
docfeedback@vmware.com

Copyright © 2012 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at http://www.vmware.com/go/patents.
VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.
Contents

Using VMware vFabric Application Director 7

1 Introducing vFabric Application Director 9
   vFabric Application Director Overview 9
   Core Architectural Principles 10
   Key Concepts 12

2 Getting Started with vFabric Application Director 15

3 Installing vFabric Application Director 17
   Preparing to Install vFabric Application Director 17
   Deploy the vFabric Application Director Appliance 21
   Start Up the vFabric Application Director Appliance 22
   Troubleshooting Problems Connecting to the vFabric Application Director Web Interface 24
   Unlock Your Darwin_user Account 24
   Restarting vFabric Application Director 25
   Configure vFabric Application Director to Use a Proxy for External URLs 25
   Upgrading vFabric Application Director 26

4 Setting Up Users and Groups 27
   Overview of Roles, Users, and Groups 27
   Predefined Users, Groups, and Roles 30
   Start the CLI in an Appliance 30
   Start the CLI Remotely 31
   Use the vFabric Application Director CLI to Create Users and Groups 32
   Create and Activate an LDAP Configuration 33
   Import a SSL Certificate for Secure LDAP Connection 34

5 Using the vFabric Application Director Web Interface 35
   Log In to vFabric Application Director 35
   Using the vFabric Application Director Web Interface 36

6 Creating and Updating Custom Virtual Machine Templates 39
   Virtual Machine Requirements for Creating Custom Templates 39
   Creating Windows Virtual Machine Templates in vCloud Director 41
   Create Linux Virtual Machine Templates in vCloud Director 44
   Create Amazon EC2 Virtual Machine Templates or AMIs 46
   Updating Existing Virtual Machine Templates 48
   Exporting Virtual Machine Templates with OVF Format 48
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Setting Up and Configuring Your Cloud Environment</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Setting Up Cloud Environments</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Registering a Cloud Provider and Template</td>
<td>52</td>
</tr>
<tr>
<td></td>
<td>Create a Cloud Tunnel to Connect to Amazon EC2</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>Creating a Deployment Environment</td>
<td>56</td>
</tr>
<tr>
<td>8</td>
<td>Developing vFabric Application Director Components</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>Defining Component Actions</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>Configuring Component Properties</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>vFabric Application Director Predefined Properties</td>
<td>66</td>
</tr>
<tr>
<td></td>
<td>Secured Component Properties</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>Required Component Properties</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>Best Practices for Developing Components</td>
<td>68</td>
</tr>
<tr>
<td>9</td>
<td>Managing the vFabric Application Director Catalog</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>Add a Logical Template to the Catalog</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>Add a Service to the Catalog</td>
<td>73</td>
</tr>
<tr>
<td></td>
<td>Add a Custom Task to the Catalog</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>Add Operating Systems to the Catalog</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>Add Tags to the Catalog</td>
<td>81</td>
</tr>
<tr>
<td>10</td>
<td>Creating Applications</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>Create an Application</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>Creating an Advanced Blueprint</td>
<td>86</td>
</tr>
<tr>
<td></td>
<td>Copy an Application Version</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>Copy an Application</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>Delete an Application Version</td>
<td>90</td>
</tr>
<tr>
<td>11</td>
<td>Deploying Applications</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>Create a Deployment Profile</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>Add Custom Tasks</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>Deploy an Application</td>
<td>96</td>
</tr>
<tr>
<td></td>
<td>Updating Application Deployments</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>Understanding the Deployment and Update Process</td>
<td>103</td>
</tr>
<tr>
<td></td>
<td>Checking the Status of a Deployment</td>
<td>104</td>
</tr>
<tr>
<td>12</td>
<td>Deploying Predefined Catalog Components</td>
<td>109</td>
</tr>
<tr>
<td></td>
<td>Using the Sample Templates</td>
<td>109</td>
</tr>
<tr>
<td></td>
<td>Available Sample Applications for Deployment</td>
<td>110</td>
</tr>
<tr>
<td></td>
<td>Sample vFabric Application Director Catalog Tasks</td>
<td>122</td>
</tr>
<tr>
<td></td>
<td>Sample vFabric Application Director Catalog Services</td>
<td>128</td>
</tr>
<tr>
<td>13</td>
<td>Managing Deployments</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>View Deployment Details for an Application</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>Teardown an Application from the Cloud</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>Delete an Application Deployment from vFabric Application Director</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>Cancel a Deployment or an Update Process</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>Collect Logs to Troubleshoot Deployment Failures</td>
<td>133</td>
</tr>
</tbody>
</table>
Using VMware vFabric Application Director

vFabric Application Director automates application provisioning in the cloud including deploying, configuring, and updating the application’s components and dependent middleware platform services on infrastructure clouds. vFabric Application Director 5.0 simplifies complex deployments of custom and packaged applications on infrastructure clouds that are based on vCloud Director and Amazon Elastic Compute Cloud (Amazon EC2).

This guide describes how to install and use vFabric Application Director to create, deploy, manage, and update applications across virtual and cloud-based infrastructures.

Intended Audience

This information is intended for anyone who wants to install and use vFabric Application Director for application deployments. This audience includes application architects and application deployers who work in collaboration with application infrastructure administrators and cloud administrators.
Introducing vFabric Application Director

VMware® vFabric™ Application Director is a model-based application provisioning solution that simplifies creating and standardizing application deployment topologies on multiple infrastructure clouds. Application architects can use a graphic-based canvas with a drag-and-drop interface to model application deployment topologies called application blueprints.

These application blueprints define the structure of the application, enable the use of standardized application infrastructure components, and include installation dependencies and default configurations for custom and packaged enterprise applications. Application blueprints are logical deployment topologies that are portable across VMware-based information as a service clouds and across public clouds such as Amazon EC2.

This chapter includes the following topics:
- “vFabric Application Director Overview,” on page 9
- “Core Architectural Principles,” on page 10
- “Key Concepts,” on page 12

vFabric Application Director Overview

vFabric Application Director has a model-driven, open, and extensible architecture. With its catalog of standard components, or services, vFabric Application Director automates and manages the update life cycle of deployments for multitier enterprise applications in hybrid cloud environments.

Enterprise users can standardize, deploy, configure, update, and scale complex applications in dynamic cloud environments. These applications can range from simple Web applications to complex custom applications and packaged applications.

Application architects can use the drag-and-drop canvas to create visual application blueprints. Application architects can use the prepopulated and extensible catalog of standard virtual machine templates, application infrastructure components, and scripts to model an application blueprint. These blueprints standardize the structure of the application, including software components, dependencies, and configurations, for repeated deployments.

Application blueprints are portable across deployment environments. After a blueprint is available, application development, QA, and release teams can work in the standards set by IT. Teams can repeatedly deploy a standard blueprint, customize configurations as allowed, and deploy within IT-approved deployment environments. You can also initiate an update process to scale clustered nodes of deployed applications and change the configuration or code of deployed applications when a new version is available.
From an application blueprint, you can create different deployments using deployment profiles to test prototypes or deploy mission-critical multitier applications in production environments. From these saved blueprints, the application deployer can generate execution plans for deploying the application to a private or public cloud. After an application is deployed, application operations teams can monitor and scale these applications by using the integrated vFabric Application Performance Manager (APM).

Optimized integration with vFabric APM refers to predefined monitoring agents available in the vFabric Application Director catalog.

**Core Architectural Principles**

vFabric Application Director is designed to automate deployments of complex applications across any IaaS cloud. vFabric Application Director 5.0 automates deployments in IaaS cloud environments based on vCloud Director 1.5, 1.5.1, or 5.1.1, and Amazon EC2.

- **Deploying Any Application and Middleware Service** on page 10
  With vFabric Application Director, application architects can use virtual machine templates and scripts to model an application deployment with any type of middleware service and application.

- **Multicloud Support** on page 10
  vFabric Application Director is designed to deploy the same application to multiple types of clouds.

- **vFabric Application Director Extensibility and Open Architecture** on page 11
  vFabric Application Director is optimized for vFabric components and is extensible to other components.

- **Standardization in vFabric Application Director** on page 11
  With vFabric Application Director, you can create reusable services using standardized configuration properties to meet strict requirements for IT compliance.

- **Policy-Based Security** on page 11
  vFabric Application Director grants local users, LDAP users, and LDAP groups specific roles for their functions.

**Deploying Any Application and Middleware Service**

With vFabric Application Director, application architects can use virtual machine templates and scripts to model an application deployment with any type of middleware service and application.

**Note** With vFabric Application Director 5.0, you can deploy applications on Windows and Linux operating systems in the vCloud Director cloud environment, and deploy applications on Linux operating systems in the Amazon EC2 cloud environment.

- Application architects can use virtual machine templates from a standardized cloud provider library defined for their enterprise.
- Application architects can also add application components to the middleware services defined in the application blueprint.

**Multicloud Support**

vFabric Application Director is designed to deploy the same application to multiple types of clouds.

- vFabric Application Director encapsulates deployment settings into deployment profiles, separate from application blueprints. Deployment profiles enable application blueprint portability across vCloud Director based private clouds and the Amazon EC2 based public clouds.
vFabric Application Director uses a cloud abstraction layer (CAL) to plug in cloud providers. vFabric Application Director automates deployments to the vCloud Director and Amazon EC2 cloud environment. vCloud Director uses open standards like the vCloud API and the Open Virtualization Format (OVF). vFabric Application Director integrates with vCloud Director through the publicly available vCloud based IaaS API. This integration allows you to deploy applications in private and public deployments of vCloud Director clouds. vFabric Application Director uses the Amazon AWS APIs and Amazon Machine Images (AMIs) to deploy applications to the Amazon EC2 cloud.

vFabric Application Director Extensibility and Open Architecture

vFabric Application Director is optimized for vFabric components and is extensible to other components.

- The vFabric Application Director catalog has predefined (out-of-the-box) services or applications for middleware services. In addition, you can add pointers to virtual machine templates residing in cloud catalogs. The vFabric Application Director catalog also lets you add definitions of dynamically installable custom services on virtual machine templates using install, configure, start, and update scripts, and appropriate configurations.

- Deployment execution plans are generated by the system based on the blueprint. These plans help users to track the status and progress of tasks during deployment.
  
  You can add custom task scripts to these plans to run in the deployed virtual machine. You can use these scripts to perform additional tasks such as security patches, audit integrations, quality and compliance reviews using third-party internal IT systems, and running smoke tests.

- To streamline the build to deployment process, organizations can further automate deployment by using the command-line interface to allow their continuous build systems or cloud provisioning portals to generate and deploy an application.

Standardization in vFabric Application Director

With vFabric Application Director, you can create reusable services using standardized configuration properties to meet strict requirements for IT compliance.

- vFabric Application Director provides a model-driven architecture that enables adding IT certified virtual machine templates and middleware services within the application blueprint.

- vFabric Application Director includes a delegation model for overriding configuration name value pairs between catalog administrator, application architect, and deployer to standardize configuration values for application and middleware service.

Policy-Based Security

vFabric Application Director grants local users, LDAP users, and LDAP groups specific roles for their functions.

You can group together local users, LDAP users, and LDAP groups to isolate applications, deployments, and deployment environments per group.
Key Concepts

To model a blueprint for deploying an application, you can use virtual machine templates that you obtain directly from a cloud provider library, as well as an extensible catalog of ready-to-use application infrastructure components and scripts. After you model your application deployment topology, you can create dependencies and edit configurations to finalize your execution plan. After you deploy the application, you can also initiate an update process to scale clustered nodes, or change the configuration or code of deployed applications when a new version is available.

The following definitions help you understand the details of this process.

**Application**
Logical deployment unit, consisting of application components and their dependent services and operating systems that can be distributed across multiple virtual machines.

**Application components**
Custom code packaged as EAR files, WAR files, and so on.

**Node**
Virtual machine or a cluster of virtual machines defined in the blueprint.

**Blueprint**
Logical topology of an application for deployment. A blueprint captures the structure of an application with logical nodes, their corresponding services and operating systems, dependencies, default configurations, and network topology requirements.

**Deployment profile**
Collection of deployment settings for a blueprint, including cluster size, CPU, memory, cloud templates, and networks.

**Cloud provider type**
Type of cloud infrastructure on which deployments can be made. vFabric Application Director supports only vCloud Director 1.5, 1.5.1, and 5.1.1, and Amazon EC2.

**Cloud provider**
A cloud instance for deployment. For vCloud Director, each vFabric Application Director cloud provider is mapped to a vCloud Director organization. For Amazon EC2, each vFabric Application Director cloud provider is mapped to an Amazon Region. You define several cloud providers for a cloud provider type.

**Deployment environment**
An environment in a cloud provider, for example, development, test, staging, and production. A cloud provider can have multiple deployment environments. For vCloud Director, a deployment environment maps to an organization virtual datacenter (vDC) for a defined cloud provider and uses resources from that vDC.

In vCloud Director, an organization vDC provides resources to an organization and is partitioned from a provider vDC. Organization vDCs provide an environment where virtual systems can be stored, deployed, and operated. They also provide storage for virtual media, such as floppy disks and CD ROMs. A single organization can have multiple organization vDCs.

For Amazon EC2, a deployment environment maps to a combination of Amazon Virtual Private Cloud (VPC) and one of the Availability Zones in a region.

**Catalog**
Library that contains logical templates, which are pointers to cloud templates. Reusable services that can be used in multiple applications and installed on a virtual machine. Tasks that can perform additional customized tasks in an application deployment.
### Logical template
A predefined virtual machine definition in vFabric Application Director. A logical template can be mapped to an actual template in the cloud catalog. Logical templates allow an application blueprint to remain cloud agnostic.

### Service
Scripted software that can be installed on a virtual machine and reused in multiple applications.

### Custom Tasks
From the execution plan, you can add custom tasks to perform additional customized tasks such as run security patches in an application deployment. You can create a custom task in the catalog and add it to an application deployment. vFabric Application Director also provides predefined tasks in the catalog that you can use to configure an APT repository, a YUM repository, or register a machine with a Red Hat Network.

### Tag
Organizes the lists of templates and services to enhance readability in the blueprint editor. A list of tags appears in the Catalog > Tags menu, and you can add new tags to the list.

### Operating system
Specifies an operating system supported by the IT organization for templates and services. A list of operating systems appears in the Catalog > Operating Systems menu, and you can add to the list.

### Properties
Configuration name-value pairs for services and application components. These are variables used by the scripts to set parameters on a script and run various configurations. For example, you can set the installation_path property value and configure installation scripts to use this property to specify the path to use to install a service during the application deployment process.

### Actions
Life cycle stages for the install, configure, start, and update scripts for services and application components to be installed.

### Execution plan
Task plan for viewing the order in which virtual machines are created and action scripts for catalog components are installed, configured, started, and updated.

### Logical network
An entity created as a logical abstraction for a network. A logical network is a cloud agnostic abstraction used to indicate network locality between nodes in an application. A logical network is dynamically mapped to a cloud network during the configuration of a deployment profile.

### Cloud network
A network defined within the cloud deployment environment.

### Teardown
Remove a vCloud Director vApp and associated virtual machines or Amazon EC2 instances of a deployed application from the cloud. All of the other resources such as storage and IP addresses that are used by these virtual machines are returned to their respective pools.
Complete these tasks to install vFabric Application Director, configure, and deploy a predefined (out-of-the-box) sample application to the vCloud Director or Amazon EC2 environment.

Table 2-1. List of Tasks to Get Started with vFabric Application Director

<table>
<thead>
<tr>
<th>Step</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Install and set up the vFabric Application Director virtual appliance. See Chapter 3, “Installing vFabric Application Director,” on page 17.</td>
</tr>
<tr>
<td>2</td>
<td>Configure vFabric Application Director to use a proxy. Perform this task when an application needs to download files from outside the corporate firewall. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.</td>
</tr>
<tr>
<td>3</td>
<td>(Optional) Enable the predefined user accounts and set account passwords using the CLI, if you plan to provide privileges to specific product areas. The admin user account is enabled by default. See “Predefined Users, Groups, and Roles,” on page 30.</td>
</tr>
<tr>
<td>4</td>
<td>Log in to vFabric Application Director and register a cloud provider. Registering a cloud provider links to a vCloud Director instance and organization vDC or an Amazon EC2 Region. See “Log In to vFabric Application Director,” on page 35 and “Registering a Cloud Provider and Template,” on page 52.</td>
</tr>
<tr>
<td>5</td>
<td>Create a deployment environment within the registered cloud provider. You must map the deployment environment to an organization vDC within vCloud Director or an Amazon VPC and associated Availability Zone before you can deploy an application. See “Creating a Deployment Environment,” on page 56.</td>
</tr>
<tr>
<td>6</td>
<td>Map the cloud template to a logical template. When you map a cloud template to a logical template, you are mapping the vFabric Application Director logical template to the actual template in the cloud provider’s catalog. See “Registering a Cloud Provider and Template,” on page 52.</td>
</tr>
<tr>
<td>7</td>
<td>Deploy a predefined sample application. vFabric Application Director includes predefined sample applications, services, and virtual machine templates to help you understand the basic concepts and start using the product. See Chapter 12, “Deploying Predefined Catalog Components,” on page 109.</td>
</tr>
<tr>
<td>8</td>
<td>Check the status of the deployment. During deployment, components are installed and configured based on the dependencies of an application. See “Understanding the Deployment and Update Process,” on page 103. You can use the user interface to check the status of an application deployment in real time. See “Checking the Status of a Deployment,” on page 104.</td>
</tr>
</tbody>
</table>
To install vFabric Application Director 5.0, you must create and deploy the virtual appliance in vCloud Director 1.5, 1.5.1, or 5.1.1.

This chapter includes the following topics:

- “Preparing to Install vFabric Application Director,” on page 17
- “Deploy the vFabric Application Director Appliance,” on page 21
- “Start Up the vFabric Application Director Appliance,” on page 22
- “Troubleshooting Problems Connecting to the vFabric Application Director Web Interface,” on page 24
- “Unlock Your darwin_user Account,” on page 24
- “Restarting vFabric Application Director,” on page 25
- “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25
- “Upgrading vFabric Application Director,” on page 26

Preparing to Install vFabric Application Director

Before you begin installing the vFabric Application Director appliance, verify that your computing environment meets the hardware and software system requirements.

vFabric Application Director requires that vCloud Director use specific configuration settings. For previously installed configurations of vCenter and vCloud Director servers, verify that these servers use the settings that work with vFabric Application Director.

- **vFabric Application Director System Requirements** on page 18
  The virtual appliance on which you run vFabric Application Director must meet certain hardware and software requirements. In addition, certain ports must be opened for vFabric Application Director.

- **Set Up vCloud Director for vFabric Application Director** on page 19
  You must configure a virtual appliance in vCloud Director to successfully install vFabric Application Director. If you have a previously installed version of vCloud Director, you must check the configuration settings to optimize vFabric Application Director.

- **Verify Your vCloud Director Setup** on page 20
  If you have a previously installed version of vCloud Director, verify that you can work with vApps, and that you can communicate with the virtual machine and external network IP addresses.
To optimize your storage and network to work with vFabric Application Director, you must verify that your vCenter cluster configurations meet certain requirements.

vFabric Application Director System Requirements

The virtual appliance on which you run vFabric Application Director must meet certain hardware and software requirements. In addition, certain ports must be opened for vFabric Application Director.

Disk Space and Memory Requirements

vFabric Application Director requires a minimum of 20GB of disk space, 2GB memory, and 1 vCPU with a speed of 2GHz.

**Note** For some organization vDCs, by default vCloud Director sometimes sets the virtual CPU to 0.24GHz, based on the vCloud Director setup. If this setting is the default in your environment, you must set the vCPU speed to 2GHz for the organization vDC in which the vFabric Application Director vApp is deployed. Otherwise, the performance of the vFabric Application Director virtual appliance is affected.

Port Requirements

You must open certain ports for the vFabric Application Director virtual machine.

<table>
<thead>
<tr>
<th>Port</th>
<th>Connection Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP Port 8443</td>
<td>External port for the vFabric Application Director user interface connection through HTTPS.</td>
</tr>
<tr>
<td>TCP Port 8080</td>
<td>Optional port for the vFabric Application Director user interface connection through HTTP.</td>
</tr>
<tr>
<td>TCP Port 80</td>
<td>External port for vami-lighttpd.</td>
</tr>
<tr>
<td>TCP Ports 5671</td>
<td>External port for vFabric RabbitMQ.</td>
</tr>
<tr>
<td>TCP Port 22</td>
<td>Optional port for the external SSH connection.</td>
</tr>
</tbody>
</table>

Licensing Requirements

You must type the serial number of a license edition to start a new vFabric Application Director appliance.

- VMware vFabric Application Director for Provisioning edition consists of the basic product features.
- VMware vFabric Application Director for Release Automation edition consists of all the features in the vFabric Application Director for Provisioning edition, updating a deployed application, and deploying applications to the Amazon EC2 environment capabilities.

Virtualization Software Requirements

To use vFabric Application Director, you must install and set up the following VMware products:

- vCloud Director 1.5, 1.5.1, or 5.1.1. See vCloud Director documentation [https://www.vmware.com/support/pubs/vcd_pubs.html](https://www.vmware.com/support/pubs/vcd_pubs.html).
Web Interface Support

vFabric Application Director supports the following Web browsers:

- Mozilla Firefox 14.0 and 15.0. Also requires Adobe Flash Player Plug-in 11.2 or later.
  The browser is supported on Mac 10.6 and Mac 10.7, Windows 7 64-bit, and Linux.
- Chrome 21.0 includes the Adobe Flash Player Plug-in 11.2 or later.
  The browser is supported on Mac 10.6 and Windows 7 64-bit.
- Internet Explorer is not supported.

VMRC Plug-In Support

The VMRC plug-in lets you connect to a deployed virtual machine directly from vFabric Application Director. This plug-in works only on supported versions of Mozilla Firefox running on Windows operating systems. The VMRC plug-in is not supported on the Mac operating system.

Supported Operating Systems for Virtual Machine Templates in the vFabric Application Director Catalog

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Operating Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>32-bit</td>
<td>Red Hat Enterprise Linux 6.1</td>
</tr>
<tr>
<td></td>
<td>Ubuntu 10.04.3</td>
</tr>
<tr>
<td></td>
<td>CentOS 5.6</td>
</tr>
<tr>
<td></td>
<td>CentOS 5.7</td>
</tr>
<tr>
<td>64-bit</td>
<td>Red Hat Enterprise Linux 6.1</td>
</tr>
<tr>
<td></td>
<td>Ubuntu 10.04.2</td>
</tr>
<tr>
<td></td>
<td>CentOS 5.6</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 R2 Enterprise with Service Pack 1</td>
</tr>
</tbody>
</table>

Note: Linux and Windows virtual machine templates with Federal Information Processing Standard (FIPS) enabled are not supported.

Set Up vCloud Director for vFabric Application Director

You must configure a virtual appliance in vCloud Director to successfully install vFabric Application Director. If you have a previously installed version of vCloud Director, you must check the configuration settings to optimize vFabric Application Director.

Prerequisites

- Familiarize yourself with the procedures for creating organizations and catalogs. See the latest vCloud Director documentation.
- Configure your vCenter settings for vFabric Application Director. See “Verify vCenter Cluster Configurations,” on page 20.
- Verify that the vCloud Director administrator uploaded templates to a catalog.

Note: If the templates are uploaded to a catalog contained in a different vCloud Organization from the catalog in which vFabric Application Director performs application deployments, you must publish the catalog and the templates. Check that the catalog has the correct attributes so that users can access it.
- Verify that you have a direct connect network with an external pool of IP addresses that vFabric Application Director can use. Check with your vCloud Director administrator to determine the number of IP addresses in the pool.

**Procedure**

- In vCloud Director, allocate a separate organization for vFabric Application Director.

**What to do next**


**Verify Your vCloud Director Setup**

If you have a previously installed version of vCloud Director, verify that you can work with vApps, and that you can communicate with the virtual machine and external network IP addresses.

vFabric Application Director supports deploying virtual machines to directly connected and NAT-routed networks. Use IPPOOL addressing for deployed virtual machines to communicate with the vFabric Application Director server over the network.

**NOTE** DHCP addressing for directly connected networks is not supported.

**Prerequisites**

- Familiarize yourself with the procedures for customizing vCloud Director. See the latest vCloud Director documentation and VMware knowledge base articles kb.vmware.com/kb/2005829 and kb.vmware.com/kb/2034092.

- Verify that the required TCP ports are open for the vFabric Application Director virtual machine. See “vFabric Application Director System Requirements,” on page 18.

**Procedure**

1. From the vCloud Director user interface, create a vApp with one virtual machine.

2. For the vCloud organization you map to the vFabric Application Director cloud provider, verify that the virtual machines in the vCloud organization have a vCloud network configuration that allows them to connect to the vFabric Application Director appliance.

3. Deploy the vApp.

4. Using the VMRC plug-in, connect to the virtual machines and ensure that they can ping an external IP.

5. Open TCP ports so that the deployed virtual machines can reach the vFabric Application Director appliance.

**What to do next**

Download and deploy the vFabric Application Director appliance. See “Deploy the vFabric Application Director Appliance,” on page 21.

**Verify vCenter Cluster Configurations**

To optimize your storage and network to work with vFabric Application Director, you must verify that your vCenter cluster configurations meet certain requirements.

**Prerequisites**

- Familiarize yourself with the procedures for creating vSphere 5.0 or 5.1 resource pools and vCenter clusters that have DRS enabled. See the vSphere documentation center.
Verify that you have one or more hosts running ESXi 5.0 or 5.1.

Verify that you have a vCenter cluster with DRS enabled.

Verify that you have the NTP client running on each of the ESXi hosts in your vCenter installation.

Synchronize the time on the host on which vFabric Application Director is deployed and on the hosts on which virtual machines are to be deployed. If the time is not synchronized, the vFabric Application Director server might experience problems when communicating with the provisioned virtual machines.

To synchronize the time on the hosts, set an NTP server to the Configuring Time option for each ESXi host underlying the vCloud Director system.

**Procedure**

1. Create a cluster with DRS enabled.
   
   You can only create a provider vDC backed by a vCenter resource pool that is part of a DRS cluster.

2. Check the configuration settings for datastore and network requirements.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
</table>
| Multiple hosts in a cluster | - Verify that all of the hosts have at least one shared datastore, for example, shared LUN, NFS, and so on.  
                            | - Verify that all of the hosts in the cluster have connectivity to at least one common network.  
                            | - To support live virtual machine migration, verify that all of the hosts in the cluster are identical. |
| Network or SAN storage   | - Verify that your host has shared storage.  
                            |   Although vCloud Director and vFabric Application Director can use local storage, shared storage ensures future scalability.  
                            | - Verify that the host has at least one network. |

**What to do next**

Download and deploy the vFabric Application Director appliance. See “Deploy the vFabric Application Director Appliance,” on page 21.

**Deploy the vFabric Application Director Appliance**

To install vFabric Application Director, you must download the appliance.

For information about adding vApps, see the vCloud Director documentation.

**Important** For vCloud Director 1.5 and 1.5.1, you must select the default Full Name that appears in the Configure Virtual Machines wizard. Changing the Full Name prevents vFabric Application Director vApp from starting up.

**Prerequisites**

Verify that you completed all of the prerequisites listed in “Preparing to Install vFabric Application Director,” on page 17.

**Procedure**

1. Download the vFabric Application Director appliance and save both files in the same folder without changing their file names.

   The appliance consists of the following files:
Replace * with the build number of the appliance.

2 Log in to vCloud Director and select the organization vDC where you want to deploy vFabric Application Director.

3 Select the Catalogs view and click the Upload button.

4 In the Upload OVF as a Template window, complete the requested information.

5 Click OK on any certificate warning pop-up windows to continue uploading the appliance.

Because of the large file size and depending on network connection speed, expect the upload process to take a significant amount of time.

6 Right-click the uploaded template and select Add to My Cloud.

7 Follow the prompts to add a vApp.

For vCloud Director 1.5 and 1.5.1, you must select the default Full Name that appears in the Configure Virtual Machines wizard. If you need to replace the Full Name of the vFabric Application Director virtual machine, edit the virtual machine properties after the instantiation process is complete.

8 Right-click the newly added vApp and click Start.

What to do next

Initialize the vFabric Application Director server. See “Start Up the vFabric Application Director Appliance,” on page 22.

Start Up the vFabric Application Director Appliance

You must start the newly added vApp to confirm that your vFabric Application Director installation was successful and complete the remaining set up procedures.

Depending on the serial number you type to install vFabric Application Director 5.0, you can access the vFabric Application Director for Provisioning or vFabric Application Director for Release Automation edition. The vFabric Application Director for Provisioning edition consists of the basic product features. The vFabric Application Director for Release Automation edition consists of the basic features, updating a deployed application, and deploying applications to the Amazon EC2 environment capabilities.

**Note** Upgrading from the vFabric Application Director for Provisioning edition to the vFabric Application Director for Release Automation edition is not supported.

To transfer the applications and their associated blueprints and deployment profiles, services, and available custom tasks from the vFabric Application Director for Provisioning edition to the vFabric Application Director for Release Automation edition, use the export and import features. See Chapter 15, “Using the CLI Import and Export Functionality,” on page 151.

**Prerequisites**

Verify that you have the VMRC plug-in installed to work with your browser. For information about compatible Web browsers, see “vFabric Application Director System Requirements,” on page 18.
Procedure

1. From the vCloud Director My Cloud view, click the image under the consoles column that corresponds to your vFabric Application Director appliance.
   
   It might take some time until the connection to the appliance is established and the vApp console view opens.

2. Click in the VMRC console.

3. At the prompt, type the vFabric Application Director serial number and press Enter.

4. Type a password for the root user account and press Enter.

5. At the prompt, retype the password to confirm it and press Enter.

6. Type a password for the darwin_user account and press Enter.
   
   Select a password for the darwin_user account that you can remember easily. When you are prompted for this password again, you have only three attempts to type the correct password.

7. At the prompt, retype the password to confirm it and press Enter.

   **NOTE** It takes some time for the boot scripts to install and start other required software processes in the background, before you are prompted to set the admin user account password. During this time, the system might appear to be unresponsive.

8. Type a password for the vFabric Application Director admin user account and press Enter.

9. At the prompt, retype the password to confirm it and press Enter.

   The boot up script should finish starting the necessary services and display the Web URL and DHCP or Static IP address for accessing the vFabric Application Director server.

10. Navigate to the Web URL using a supported browser.

    The Web URL format is https://Application_Director_IP:8443/darwin.

11. Log in as the admin user.

    Use the password you set for the admin user account. The admin user can use the CLI to enable the preconfigured user accounts and create additional users and groups with specific roles.

    The browser opens the vFabric Application Director user interface. See “Using the vFabric Application Director Web Interface,” on page 36.

   **NOTE** To shut down the vFabric Application Director appliance, use vCloud Director. Do not use vSphere vCenter to shut down the appliance.

What to do next

Register a Cloud Provider. See “Registering a Cloud Provider and Template,” on page 52.
Troubleshooting Problems Connecting to the vFabric Application Director Web Interface

A few situations can cause connection problems when you attempt to access the vFabric Application Director Web interface.

Table 3-1. Common Connection Errors

<table>
<thead>
<tr>
<th>Error</th>
<th>Possible Cause</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>The vFabric Application Director virtual appliance does not have a working IP address or network.</td>
<td>The following problems might cause a networking error:</td>
<td>You must resolve the networking issue.</td>
</tr>
<tr>
<td></td>
<td>■  Organizational vDC network is not configured properly.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■  Network settings not specified in the Configure Virtual Machines wizard.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■  IP address is not specified for IP Assignment setting in the Configure Virtual Machines wizard.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■  Static IP address is not available.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■  DHCP server is not available.</td>
<td></td>
</tr>
<tr>
<td>The vFabric Application Director virtual appliance temporarily fails to retrieve a working IP address during start up, or the IP address changes after start up.</td>
<td>The following problems might cause a networking error:</td>
<td>Run the following command in the appliance virtual machine with root privileges:</td>
</tr>
<tr>
<td></td>
<td>■  A problem was encountered with the network.</td>
<td>/home/darwin/tools/darwin_util.sh -a AUTO</td>
</tr>
<tr>
<td></td>
<td>■  Static IP address is not available.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■  DHCP server is not available.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■  The IP address was explicitly modified.</td>
<td></td>
</tr>
<tr>
<td>The vFabric tc Server service encounters an error in the vFabric Application Director virtual appliance.</td>
<td>The virtual appliance was not shut down properly and restarted.</td>
<td>Navigate to the log file at /home/darwin/tcservers/darwin/logs/catalina.out for error details and contact VMware technical support if needed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unlock Your darwin_user Account

If you do not provide the correct password after three attempts when you use the SSH client to log in with your darwin_user account, you are locked out of the darwin_user account.

Prerequisites

- Verify that you are logged in to the VMRC console of the vFabric Application Director appliance.
  - SSH is disabled for the root account.
- Verify that you have root privileges.

Procedure

1. In the VMRC console, log in as the root user.
2. Check the number of failed login attempts from the shell prompt.
   
   `faillog -u darwin_user`
3 Unlock the failed account.

    faillog -u darwin_user -r

### Restarting vFabric Application Director

In some cases, you might have to restart vFabric Application Director.

**CAUTION** Do not restart your vFabric Application Director virtual machine using the `/home/darwin/tcserver/darwin/bin/tcruntime-ctl.sh` restart command. If you use this command, the vFabric Application Director virtual machine uses default redis and agent jar paths causing deployments to fail.

**Procedure**

1. Log in to your vFabric Application Director virtual machine using the SSH client or vCloud Director console.
2. Restart vFabric Application Director.
   - If you are logged in with the `darwin_user` account, type `sudo service vmware-darwin-tcserver restart`.
   - If you are logged in with the root account, type `service vmware-darwin-tcserver restart`.

### Configure vFabric Application Director to Use a Proxy for External URLs

Even if you use vFabric Application Director only to deploy applications in a private cloud, some deployments might require access to URLs from outside the corporate firewall. For example, an action script might involve downloading application bits from an open-source Web site. You can configure vFabric Application Director to use a proxy for these use cases.

You must complete this task before you create new services and applications or before you deploy any existing predefined sample services and applications.

vFabric Application Director also contains a proxy file called `darwin_global_noproxy.conf`, which does not define a proxy. You can specify this file as the `global_conf` property value if a deployment environment does not require a proxy. For example, the Amazon EC2 deployment environment does not require a proxy to deploy an application.

**Prerequisites**

- Verify that you have access to the virtual machine where vFabric Application Director is installed and have the password for logging in to the operating system with the `darwin_user` account. This password was set during installation. See “Start Up the vFabric Application Director Appliance,” on page 22.
- Verify that your vFabric Application Director user account has the `ROLE_CATALOG_ADMIN` catalog administrator role assigned to it.
- Verify that your vFabric Application Director user account has the `ROLE_APP_ARCHITECT` application architect role assigned to it.
- Familiarize yourself with the procedure for creating new services. See “Add a Service to the Catalog,” on page 73.

**Procedure**

1. Configure the vFabric Application Director virtual appliance to use a proxy.
   a. Log in to the vFabric Application Director virtual appliance as `darwin_user`.
   b. Type `su` to switch to root user.
c Open the file /home/darwin/tcserver/darwin/webapps/darwin/conf/darwin_global.conf with a text editor.

d Update the proxy IP and port information and save the file.

The proxy URL format is http://proxy:PortNumber.

The proxy specified in this file is used by any existing predefined applications or catalog services that must access a repository by using a yum update or install command. The predefined services and applications have the script required to access this proxy.

2 Log in to vFabric Application Director as a user with the catalog administrator role and define proxy-specific properties and scripts to create a new service or customize an existing service that you already created.

   a For a new service, add a service to the catalog.

   b Add a property with the name global_conf to the service, of type Content, and define the value as https://DarwinServerIP:8443/darwin/conf/darwin_global.conf.

   Note If you add a value to the http_proxy, https_proxy, or ftp_proxy property with service scripts that use the darwin_global.conf as a file source, when the script runs, these properties override any existing proxy information in the deployed application.

   c Add the following lines to the beginning of each action script for the service that requires a proxy:

   # Import global conf
   . $global_conf

   d Save the service.

3 Log in to vFabric Application Director as a user with the application architect role and define proxy-specific properties and scripts in the application blueprint to configure an application to use a proxy while creating the application blueprint.

   a For the application component, add the global_conf property to the application component, of type Content, and define the value as https://DarwinServerIP:8443/darwin/conf/darwin_global.conf.

   b Add the following lines to the beginning of each action script for the application component that requires a proxy:

   # Import global conf
   . $global_conf

What to do next

Log in to vFabric Application Director and register a cloud provider. See “Log In to vFabric Application Director,” on page 35 and “Register the vCloud Director Cloud Provider and Template,” on page 52.

Upgrading vFabric Application Director

An older version of the vFabric Application Director appliance cannot be upgraded to the latest version.

You must create and deploy a new virtual appliance in vCloud Director 1.5, 1.5.1, or 5.1.1, and install the latest version of vFabric Application Director. For more information about licensing requirements, see “vFabric Application Director System Requirements,” on page 18.

To transfer the applications and their associated blueprints and deployment profiles, services, and available custom tasks between vFabric Application Director 5.0 appliances and product license editions, use the export and import features. See Chapter 15, “Using the CLI Import and Export Functionality,” on page 151.
Setting Up Users and Groups

After you install vFabric Application Director, you must create accounts for users and assign users to groups. Several preconfigured user accounts are available, but are disabled by default.

**Note** For vFabric Application Director 5.0, you must use the command-line interface (CLI) to create groups, user accounts, and LDAP configurations. You must also use the CLI to manage users, assign roles, and import existing users and groups from an LDAP directory.

This chapter includes the following topics:

- “Overview of Roles, Users, and Groups,” on page 27
- “Predefined Users, Groups, and Roles,” on page 30
- “Start the CLI in an Appliance,” on page 30
- “Start the CLI Remotely,” on page 31
- “Use the vFabric Application Director CLI to Create Users and Groups,” on page 32
- “Create and Activate an LDAP Configuration,” on page 33
- “Import a SSL Certificate for Secure LDAP Connection,” on page 34

**Overview of Roles, Users, and Groups**

You must determine who can use vFabric Application Director and what tasks those users are authorized to perform. You can selectively assign administrative permissions by assigning roles to specific users. You can limit access to specific deployment environments and cloud templates by associating each user with a specific group in vFabric Application Director.

**Roles**

You can specify which functions the user can perform in vFabric Application Director by associating a local user, an LDAP user, or an LDAP group with one or more roles. These functions include managing user accounts, managing the catalog of templates and services, managing the cloud providers and deployment environments, creating applications, and deploying applications.

The built-in admin user has the **ROLE_SYSTEM_ADMIN, ROLE_CLOUD_ADMIN, ROLE_CATALOG_ADMIN, ROLE_APP_ARCHITECT, ROLE_DEPLOYER, and ROLE_SYSTEM_INTEGRATOR** roles assigned to it. See “Predefined Users, Groups, and Roles,” on page 30.
Local Users

A local user has information, including password, stored in the vFabric Application Director database. A user with the `ROLE_SYSTEM_ADMIN` role can use the vFabric Application Director command-line interface to perform the following tasks:

- Create vFabric Application Director users.
- Specify any combination of roles a user has, depending on the tasks that the user is required to perform.
- Specify the group to which a user belongs.
- Change passwords.
- Enable or disable user accounts.
- Create LDAP configurations.
- Import LDAP users and groups.
- Manage LDAP users and groups.


All of the users of a group can view applications, deployments, cloud templates, and cloud providers that a local user created. Local users belonging to a vFabric Application Director group cannot view applications, deployments, cloud templates, and cloud providers of another group.

**Note** For vFabric Application Director 5.0, each local user can be associated with only one vFabric Application Director group. If a local user who belongs to a different vFabric Application Director group needs access to applications, deployments, templates, or cloud providers that belong to other groups, a system administrator must provide such users with multiple accounts to use and make sure that the accounts belong to the specific group that the user intends to access.

LDAP Users

LDAP users are user accounts that are authenticated by a remote LDAP server during login. Minimal information about an LDAP user is stored in the vFabric Application Director database. The following user information is stored:

- Username
- SID information of the user
- vFabric Application Director group with which the user is associated
- vFabric Application Director roles the user is assigned

**Note** vFabric Application Director does not save LDAP user password information.

LDAP users can belong to only one vFabric Application Director group. LDAP users that belong to one group cannot view applications, deployments, cloud templates, or cloud providers of any other group.

Only the user account with the `ROLE_SYSTEM_ADMIN` role can perform these actions:

- Import existing LDAP users and groups to vFabric Application Director.
- Assign the imported users and groups with any combination of roles, depending on the tasks that they are required to perform.
During the import process, the vFabric Application Director server communicates with the LDAP server to determine whether the user account exists in the LDAP directory. After the confirmation is received, the user’s SID information is copied from the LDAP directory and an entry is created in the vFabric Application Director database for that user.

No other synchronization takes place because vFabric Application Director does not store user data. Required user data is retrieved when the user logs into the system.

**vFabric Application Director Groups**

With vFabric Application Director, you can create vFabric Application Director groups and assign local users, LDAP users, and LDAP groups to one vFabric Application Director group.

vFabric Application Director groups provide isolated access to the various areas of the product. Assigning a user to a group determines which of the following vFabric Application Director components that the user can access:

- Applications, including specific application versions and deployments
- Logical templates
- Cloud providers, including deployment environments

Each local user account, LDAP user account, and LDAP group account is associated with only one vFabric Application Director group. You can associate an LDAP user or LDAP group with a vFabric Application Director group when you import the user or group from the LDAP directory. If multiple LDAP groups are imported that are associated with different vFabric Application Director groups, and a user belongs to multiple LDAP groups, then the user that belongs to multiple LDAP groups is assigned to only the first group association.

Logical templates, applications, cloud providers, and deployment environments are also associated with the group of the user that created them, to allow limited access to components of vFabric Application Director. You can associate a local user with a group when you create the user account.

For example, when a user in abcGroup creates an abcApp application, all of the local users, LDAP users, and LDAP groups in that group can access the application. A local user in xyzGroup can create an application xyzApp for all of the users in xyzGroup to access. A local user in abcGroup cannot access the xyzApp application and the same is true for a local user in xyzGroup. If a local user’s group is changed, any applications created as a member of that group remain in that group. The limited group access also applies to cloud providers, deployment environments, logical templates, and deployments.

The built-in Default group includes all of the predefined sample applications and logical templates. The built-in user accounts, including the admin user, belong to this Default group.

**LDAP Groups**

LDAP groups are imported from the LDAP server and associated with vFabric Application Director roles. An LDAP group can be assigned a set of roles, in the same way that a local user or LDAP user can be assigned roles. Importing an LDAP group allows all the LDAP users in the group to log in to the vFabric Application Director appliance without being individually imported.

When LDAP users log in, their LDAP group credentials are evaluated. The actual roles assigned to the user is an aggregate of all the roles assigned to the imported LDAP user account and the roles for all of the groups to which the user belongs to. This accretion of roles depends on whether the LDAP groups are imported to vFabric Application Director and assigned roles.

Each LDAP group is associated with one vFabric Application Director group.
Predefined Users, Groups, and Roles

vFabric Application Director provides five predefined user accounts with roles that map to specific privileges within product areas.

<table>
<thead>
<tr>
<th>User</th>
<th>Roles Assigned</th>
<th>Product Area Managed</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin</td>
<td>ROLE_SYSTEM_ADMIN, ROLE_CLOUD_ADMIN, ROLE_CATALOG_ADMIN, ROLE_APP_ARCHITECT, ROLE_DEPLOYER, ROLE_SYSTEM_INTEGRATOR</td>
<td>Users, groups, and roles</td>
<td>System administrator manages users and groups. Because this account includes all roles, you can also perform the actions listed for the other users.</td>
</tr>
<tr>
<td>cloudAdmin</td>
<td>ROLE_CLOUD_ADMIN</td>
<td>Cloud providers and deployment environments</td>
<td>Manage cloud providers and deployment environments.</td>
</tr>
<tr>
<td>catalogAdmin</td>
<td>ROLE_CATALOG_ADMIN</td>
<td>Catalog</td>
<td>Maintain and add services, templates, and tasks to the catalog.</td>
</tr>
<tr>
<td>appArchitect</td>
<td>ROLE_APP_ARCHITECT</td>
<td>Applications</td>
<td>Design and create applications.</td>
</tr>
<tr>
<td>deployer</td>
<td>ROLE_DEPLOYER</td>
<td>Deployments</td>
<td>Initiate deployments, update an existing deployment, and teardown a deployed application from the cloud.</td>
</tr>
</tbody>
</table>

All of the predefined user accounts except the admin user are disabled by default. You must enable these built-in user accounts and set the passwords.

The password for the admin user is the admin password that was set the first time the appliance was started.

Start the CLI in an Appliance

You can start the command-line interface (CLI) from a vFabric Application Director appliance.

**Prerequisites**

Verify that you know the password for the darwin_user. This password was set during installation. See “Start Up the vFabric Application Director Appliance,” on page 22.

**Procedure**

1. Use the SSH client to log in to the vFabric Application Director appliance as the user darwin_user.
   
   The password for this account was set during installation.

2. Change directory to /home/darwin/tools.

3. Start the vFabric Application Director CLI.
   
   java -jar darwin-cli.jar

4. Log in to the vFabric Application Director server.
   
   login --serverUrl https://DarwinServerIP:8443/darwin --username UserName

   Replace DarwinServerIP with the vFabric Application Director server IP address.

   If you are running the CLI in the vFabric Application Director server, you can specify the localhost for the DarwinServerIP.

   If you run the --password parameter with the log in command, your password is saved as plain text in the log.roo file. For added security, delete this log file.
5 Type the password when prompted.
   The system returns a message similar to this:

   You are logged in to https://DarwinServerIP:8443/darwin as UserName.

What to do next

Create, enable, and manage the predefined users and groups. See “Use the vFabric Application Director CLI to Create Users and Groups,” on page 32 or “Managing Users and Groups,” on page 142.

Start the CLI Remotely

You can start the vFabric Application Director CLI from a remote machine.

Prerequisites

- Verify that you know the password for the darwin_user. This password was set during installation. See “Start Up the vFabric Application Director Appliance,” on page 22.
- Familiarize yourself with the roles available for users. See “Predefined Users, Groups, and Roles,” on page 30.
- Verify that you installed Java (JDK 1.6) on your remote machine.
- Make sure that the remote machine can connect to the vFabric Application Director appliance using HTTPS.

Procedure

1 On the vFabric Application Director appliance, open a command prompt and change directories.
   cd /home/darwin/tools
2 Copy the darwin-cli.jar file to the remote machine.
3 To open the vFabric Application Director CLI from a remote machine, log in to the remote machine.
4 Change directory to the folder containing the copied darwin-cli.jar file.
5 Start the client.
   java -jar darwin-cli.jar
6 Log in to the vFabric Application Director server.
   login --serverUrl https://DarwinServerIP:8443/darwin --username UserName
   Replace DarwinServerIP with the vFabric Application Director server IP address.
   If you are running the CLI in the vFabric Application Director server, you can specify the localhost for the DarwinServerIP.
   If you run the --password parameter with the log in command, your password is saved as plain text in the log.roo file. For added security, delete this log file.
7 Type the password when prompted.
   The system returns a message similar to this:

   You are logged in to https://DarwinServerIP:8443/darwin as UserName.

What to do next

Determine the type of users and groups to create in vFabric Application Director and use the CLI options to enable and manage them. See “Use the vFabric Application Director CLI to Create Users and Groups,” on page 32 and “Managing Users and Groups,” on page 142.
Use the vFabric Application Director CLI to Create Users and Groups

The vFabric Application Director CLI is a Spring Roo-based client that communicates to the vFabric Application Director server over HTTPS using REST APIs. You must use the CLI to create users and groups.

Prerequisites

- Verify that your user account has the ROLE_SYSTEM_ADMIN system administrator role assigned to it.
- Familiarize yourself with the roles available for users. See “Predefined Users, Groups, and Roles,” on page 30.
- Verify that you know the password for the darwin_user. This password was set during installation. See “Start Up the vFabric Application Director Appliance,” on page 22.
- Start the vFabric Application Director CLI. See “Start the CLI in an Appliance,” on page 30 and “Start the CLI Remotely,” on page 31.

Procedure

1. In the roo shell prompt, press the Tab key to display the list of available commands.
2. Type the command to perform the appropriate action.

<table>
<thead>
<tr>
<th>Option</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>List usage information</td>
<td>help</td>
</tr>
<tr>
<td>Create a group</td>
<td>create-group --name GroupName --description Description</td>
</tr>
<tr>
<td>Create a user</td>
<td>create-user --username UserName --firstName FirstName --lastName LastName --enabled true --roles ROLE_SYSTEM_ADMIN, ROLE_CATALOG_ADMIN, ROLE_CLOUD_ADMIN, ROLE_DEPLOYER, ROLE_APP_ARCHITECT, ROLE_SYSTEM_INTEGRATOR --group GroupName --email EmailAddress</td>
</tr>
</tbody>
</table>

If you run the --password parameter with the log in command, your password is saved as plain text in the log.roo file. For added security, delete this log file.

If you specify all the roles shown in this example, the user is the equivalent of the admin user. Specify only the roles the specific user must have.

The valid values for the enabled option are true, yes, 1, false, no, and 0.

For a vFabric Application Director object or entity, if you use multiple words with spaces between the words, enclose the words in quotation marks.

For example, to create a group called Test Group, use the command create-group --name "Test Group".

Often after you enter a command, the system displays many details relating to the user, in addition to indicating whether the command was successful.

What to do next

Use the CLI to perform tasks such as changing passwords or enabling and disabling accounts for users and groups. See “Managing Users and Groups,” on page 142.
Create and Activate an LDAP Configuration

LDAP is a central authentication mechanism that lets you use a login credential to access multiple servers and groups with which you are associated.

If an organization already uses LDAP to handle its authentication and directory services, vFabric Application Director server can integrate with the existing LDAP authentication mechanism. You can use the CLI to create and manage LDAP configurations in vFabric Application Director.

**Note** You can have multiple LDAP servers defined in a vFabric Application Director server, but you can have only one LDAP configuration active at any given time for authentication.

**Prerequisites**

- Verify that your user account has the **ROLE_SYSTEM_ADMIN** system administrator role assigned to it.
- Verify that you know the password for the darwin_user. This password was set during installation. See “Start Up the vFabric Application Director Appliance,” on page 22.
- Start the vFabric Application Director CLI. See “Start the CLI in an Appliance,” on page 30 and “Start the CLI Remotely,” on page 31.

**Procedure**

1. In the root shell, follow the prompts to create an LDAP configuration.
   
   ```
   create-ldap-config
   ```
   
   vFabric Application Director creates the LDAP configuration and saves the configuration in its database.

2. (Optional) Check the existence of a user name in the LDAP directory to confirm server connectivity.
   
   ```
   test-named-ldap-config --configname LDAPName --username UserName
   ```

3. (Optional) Activate an LDAP configuration if it was not activated during the initial creation or to activate another LDAP configuration.
   
   ```
   activate-ldap-config --configname LDAPName
   ```

   Activating an LDAP configuration in the vFabric Application Director server allows authentication against the named LDAP configuration.

   Even after the LDAP configuration is activated, vFabric Application Director always searches the local database first for valid users before performing an LDAP authentication. This verification action ensures that local users are not locked out if the same user name exists on the local database and the LDAP server. If multiple user names exist on the local database, the first entry that was added is used for authentication.

**What to do next**

To connect to your LDAP server over a secure channel when your LDAP certificate is signed by a local authority, or self-signed, import the LDAP certificate to the vFabric Application Director server openssl trusted list. See “Import a SSL Certificate for Secure LDAP Connection,” on page 34.

Use the CLI to perform tasks such as importing users and groups and updating existing LDAP configurations. See “Managing LDAP Configurations,” on page 143.
Import a SSL Certificate for Secure LDAP Connection

vFabric Application Director server supports most public issuers of SSL certification by using the JDK cacerts file to securely connect to the LDAP server. If the LDAP server certificate is signed by a local authority, or self-signed, you must import the LDAP certificate to the vFabric Application Director server openssl trusted list.

Prerequisites

- Verify that your user account has the ROLE_SYSTEM_ADMIN system administrator role assigned to it.
- Verify that you know the password for the darwin_user. This password was set during installation. See “Start Up the vFabric Application Director Appliance,” on page 22.
- Copy the LDAP server certificate file as the certificate.pem file to a directory on the vFabric Application Director server.

Procedure

1. Use the SSH client to log in to the vFabric Application Director appliance as the user darwin_user.

   The password for this account was set during installation.

2. Open a command prompt.

3. Switch user from darwin_user to darwin.

   su darwin

4. Change to the keystore directory.

   cd /home/darwin/keystore

5. Copy the LDAP certificate.pem file to the directory.

   cp Directory_certificate_copied/certificate.pem /home/darwin/keystore/

6. Add the certificate file to the vFabric Application Director server trusted list.

   keytool -import -trustcacerts -alias UniqueAlias -file certificate.pem -keystore ./appd.truststore -storepass ""

7. Restart the vFabric Application Director server.

   sudo service vmware-darwin-tcserver restart

What to do next

Use the CLI to perform tasks such as importing users and groups and updating existing LDAP configurations. See “Managing LDAP Configurations,” on page 143.
Using the vFabric Application Director Web Interface

You can use the vFabric Application Director Web interface to register cloud providers, maintain the catalog of virtual machine templates and services, create applications, deploy applications, update deployed applications, and teardown deployed applications from the cloud.

**Note** For vFabric Application Director 5.0, you must use the CLI to create groups and user accounts, manage users, assign roles, manage LDAP configurations, and import and export packages between different instances.

You can also use the CLI to deploy applications, update deployed applications, and teardown deployed applications. See “General CLI Options,” on page 141.

This chapter includes the following topics:

- “Log In to vFabric Application Director,” on page 35
- “Using the vFabric Application Director Web Interface,” on page 36

**Log In to vFabric Application Director**

To see the sample applications included with vFabric Application Director, you must log in using an account that belongs to the Default group.

Whether you can perform a particular task after you log in depends on the roles assigned to the user account. The objects you can see depend on the group associated with the user. See Chapter 4, “Setting Up Users and Groups,” on page 27.

**Prerequisites**

- Verify that you have the URL for the vFabric Application Director Web interface. This Web URL appears in the console of the virtual machine that hosts vFabric Application Director when installation is finished. See “Start Up the vFabric Application Director Appliance,” on page 22.
- Verify that you have credentials for a user account that was set up in vFabric Application Director. You can also use the admin user account and the password that was set during installation. The admin user has all roles associated with it and can perform all of the functions in vFabric Application Director.
- Verify that the supported versions of a Web browser and Adobe Flash Player are installed on your computer. See “vFabric Application Director System Requirements,” on page 18.

**Procedure**

1. Open a Web browser and type the vFabric Application Director URL. An example of a URL for accessing the Web interface is https://192.0.2.255:8443/darwin.
2. Type the credentials of a user account with the necessary roles assigned to it.
You can access the product areas that are assigned to your role. For example, if your user account has the deployer role assigned to it, when you log in to the vFabric Application Director Web interface, the Deployments page opens.

**Using the vFabric Application Director Web Interface**

With the vFabric Application Director Web interface, you can access almost all of the available functions.

The Web user interface includes the standard features of a Web application. For example, with the search box, you can select filtering criteria that are related to the objects you are searching for. If your search string includes an underscore, vFabric Application Director ignores the underscore as a wildcard. For example, if you have an application called test_qe and another application called testapp, when you type the search string `test_` in the search box, both of the applications appear in the search result.

You can access applications, deployments, and the cloud environment depending on the group associated with your user account when you log in to vFabric Application Director. All users can see the sample services, tasks, operating systems, and tags included in the vFabric Application Director catalog. The vFabric Application Director title bar includes a drop-down menu that includes the main categories Applications, Deployments, Clouds, and Catalog. In addition, the drop-down menu displays the subcategories under Clouds and Catalog.

From the vFabric Application Director title bar, click the **Applications** drop-down menu and select a category to navigate from one page to another.

**Figure 5-1. Title Bar Menu Options**

The Applications page organizes the sample applications. From this page, you can configure the sample applications to deploy to a cloud environment. The Deployments page lists both successful and failed deployments. From this page, you can initiate an update process to scale clustered nodes or modify the configuration of a deployed application. The Clouds category lets you create cloud providers and deployment environments. The Catalog category lets you create services, logical templates, tasks, operating systems, and tags for your application. You can also use the sample services and other components in the catalog to model an application blueprint.

Click the user name in the vFabric Application Director title bar to view the **Logout** option and exit the current session.
Table 5-1. vFabric Application Director Buttons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Add Relation" /></td>
<td>Add Relation</td>
<td>Available in the blueprint. Creates relations between components in a blueprint.</td>
</tr>
<tr>
<td><img src="image" alt="Convert to Node Array" /></td>
<td>Convert to Node Array</td>
<td>Available in the blueprint, Creates a clustered node, you can also use the button to convert a clustered node to a single node.</td>
</tr>
<tr>
<td><img src="image" alt="Add" /></td>
<td>Add</td>
<td>Available in the blueprint. Adds a NIC to a node from the NICs tab or to add application component properties from the Properties tab.</td>
</tr>
<tr>
<td><img src="image" alt="Reset" /></td>
<td>Reset</td>
<td>Available on the Deployment Profile wizard. Reverts to the original value. Also available on the Update Deployment Profile wizard, service version, logical template version, and task version pages.</td>
</tr>
<tr>
<td><img src="image" alt="View Task Information" /></td>
<td>View Task Information</td>
<td>Available in execution plans. View action script text, properties, and logs.</td>
</tr>
<tr>
<td><img src="image" alt="Expand" /></td>
<td>Expand</td>
<td>Available in execution plans. Displays all virtual machines in a cluster node.</td>
</tr>
<tr>
<td><img src="image" alt="Collapse" /></td>
<td>Collapse</td>
<td>Available in execution plans. Minimizes all virtual machines in a cluster node.</td>
</tr>
<tr>
<td><img src="image" alt="Add Script Task" /></td>
<td>Add Script Task</td>
<td>Available in execution plans. Adds custom tasks to an application deployment.</td>
</tr>
<tr>
<td><img src="image" alt="Notification" /></td>
<td>Notification</td>
<td>Appears when a task is incomplete or a user performs a task that is not allowed.</td>
</tr>
<tr>
<td><img src="image" alt="Expand" /></td>
<td>Expand</td>
<td>Maximizes a status window on the deployment summary page.</td>
</tr>
</tbody>
</table>
Table 5-1. vFabric Application Director Buttons (Continued)

<table>
<thead>
<tr>
<th>Icon</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Minimize" /></td>
<td>Minimize</td>
<td>Collapses a status window in the deployment summary page.</td>
</tr>
<tr>
<td><img src="image" alt="View Logs" /></td>
<td>View Logs</td>
<td>Available on a deployment details page or an execution plan after deployment.</td>
</tr>
<tr>
<td><img src="image" alt="Copy Application Version" /></td>
<td>Copy Application Version</td>
<td>Creates a new version of an existing application.</td>
</tr>
<tr>
<td><img src="image" alt="Quick Deploy" /></td>
<td>Quick Deploy</td>
<td>Available on the Applications page and application version card. Deploys an application without configuring any of the elements in the Deployment Profile wizard.</td>
</tr>
<tr>
<td><img src="image" alt="Update Deployment" /></td>
<td>Update Deployment</td>
<td>Initiates an update process to scale a clustered node or modify configuration of a deployed application.</td>
</tr>
<tr>
<td><img src="image" alt="Teardown Deployment" /></td>
<td>Teardown Deployment</td>
<td>Removes a deployed application from a cloud environment.</td>
</tr>
</tbody>
</table>

Understanding the vFabric Application Director Icons

The vFabric Application Director Web interface includes icons that represent the availability of a function.

Table 5-2. vFabric Application Director Icons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Auto-Bind Consume" /></td>
<td>Auto-Bind Consume</td>
<td>Appears in the blueprint canvas, next to properties that can Auto-Bind with another property, which is set to Expose.</td>
</tr>
<tr>
<td><img src="image" alt="Auto-Bind Expose" /></td>
<td>Auto-Bind Expose</td>
<td>Appears in the blueprint canvas, next to properties that can be used to customize another property, which is set to Consume.</td>
</tr>
</tbody>
</table>
Creating and Updating Custom Virtual Machine Templates

In vFabric Application Director 5.0, you can use supported Windows and Linux operating systems to create custom virtual machine templates for your applications. You can also customize existing Amazon Machine Images (AMIs) or virtual machine templates and use them in vFabric Application Director.

To create custom virtual machine templates to use in vFabric Application Director, verify that certain software and virtualization requirements are met and that certain Linux and Windows commands are available.

Familiarize yourself with the key concepts relating to creating and updating custom virtual machine templates. See “Key Concepts,” on page 12.

This chapter includes the following topics:

- “Virtual Machine Requirements for Creating Custom Templates,” on page 39
- “Creating Windows Virtual Machine Templates in vCloud Director,” on page 41
- “Create Linux Virtual Machine Templates in vCloud Director,” on page 44
- “Create Amazon EC2 Virtual Machine Templates or AMIs,” on page 46
- “Updating Existing Virtual Machine Templates,” on page 48
- “Exporting Virtual Machine Templates with OVF Format,” on page 48

Virtual Machine Requirements for Creating Custom Templates

In vFabric Application Director, you map logical templates to cloud templates created in vCloud Director and Amazon EC2. These cloud templates must meet certain requirements to work properly in vFabric Application Director.

Table 6-1. Virtual Machine Requirements for Custom Templates

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Use one of the following operating systems:</td>
</tr>
<tr>
<td></td>
<td>- Supported operating systems with 32-bit Red Hat Enterprise Linux 6.1, Ubuntu 10.04.3, CentOS 5.6, and CentOS 5.7</td>
</tr>
<tr>
<td></td>
<td>- Supported operating systems with 64-bit Red Hat Enterprise Linux 6.1, Ubuntu 10.04.2, CentOS 5.6, and Windows Server 2008 R2 Enterprise with Service Pack 1</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> vFabric Application Director supports the CentOS operating system in the Amazon EC2 virtual machine templates or AMIs.</td>
</tr>
<tr>
<td>VMware Tools</td>
<td>VMware Tools must be installed and the version must be from vCloud Director 1.5, 1.5.1, or 5.1.1 or vCenter Server 5.0.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> VMware Tools requirement is not applicable for Amazon EC2 virtual machine templates or AMIs.</td>
</tr>
<tr>
<td></td>
<td>See the Installing and Configuring VMware Tools guide.</td>
</tr>
<tr>
<td>Requirement</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CD/DVD drive</td>
<td>At least one CD/DVD drive must be available on the vCloud Director virtual machine. See the vCloud Director documentation. Note: CD/DVD drive requirement is not applicable for Amazon EC2 virtual machine templates or AMIs.</td>
</tr>
</tbody>
</table>
| JRE                  | JRE 1.6.0 must be installed. The preferred and supported JRE can be installed from one of these packages, which are available in the vFabric Application Director virtual appliance: JRE for Windows http://Application_Director_IP/agent/jre-1.6.0_31-win64.zip JRE for Linux  
  - http://Application_Director_IP/agent/jre-6u31-linux-amd64.rpm  
  - http://Application_Director_IP/agent/jre-6u31-linux-i586.rpm  
  - http://Application_Director_IP/agent/jre-6u31-linux-x64.bin  
  - http://Application_Director_IP/agent/jre-6u31-linux-i586.bin  
To install JRE on an RPM-based or DEB-based virtual machine template, see “Create Linux Virtual Machine Templates in vCloud Director,” on page 44. To install JRE on an RPM-based AMI, see “Create Amazon EC2 Virtual Machine Templates or AMIs,” on page 46. |
| Linux agent bootstrap service | Download the vmware-appdirector-agent-service file from the VMware product download site. Install the Linux agent bootstrap script from one of the following packages: Linux agent bootstrap service on vCloud Director  
  - http://Application_Director_IP/agent/vmware-appdirector-agent-service_5.0.0.0-0_x86_64.rpm  
  - http://Application_Director_IP/agent/vmware-appdirector-agent-service_5.0.0.0-0_i386.rpm  
  - http://Application_Director_IP/agent/vmware-appdirector-agent-service_5.0.0.0-0_amd64.deb  
  - http://Application_Director_IP/agent/vmware-appdirector-agent-service_5.0.0.0-0_i386.deb  
Linux agent bootstrap service on Amazon EC2  
  - http://Application_Director_IP/agent/vmware-appdirector-agent-service-ec2_5.0.0.0-0_x86_64.rpm  
  - http://Application_Director_IP/agent/vmware-appdirector-agent-service-ec2_5.0.0.0-0_i386.rpm  
To install the agent bootstrap service on an RPM-based or DEB-based virtual machine template, see “Create Linux Virtual Machine Templates in vCloud Director,” on page 44. To install the agent bootstrap service on an RPM-based AMI, see “Create Amazon EC2 Virtual Machine Templates or AMIs,” on page 46. |
| Windows agent bootstrap service on vCloud Director | Download the http://Application_Director_IP/agent/vmware-appdirector-agent-bootstrap-windows_5.0.0.0.zip file from the vFabric Application Director virtual appliance. To install the agent bootstrap service on a Windows-based virtual machine template, see “Create and Configure Windows Virtual Machine Templates,” on page 41. |
| Supported Windows scripting | vFabric Application Director supports scripting with Windows CMD or PowerShell 2.0. |
| Supported Linux scripting | vFabric Application Director supports scripting with bash. |
Table 6-1. Virtual Machine Requirements for Custom Templates (Continued)

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux commands</td>
<td>The following Linux commands must be available on the virtual machine:</td>
</tr>
<tr>
<td></td>
<td>- wget</td>
</tr>
<tr>
<td></td>
<td>- md5sum</td>
</tr>
<tr>
<td></td>
<td>- grep</td>
</tr>
<tr>
<td></td>
<td>- sed</td>
</tr>
<tr>
<td></td>
<td>- setsid</td>
</tr>
<tr>
<td></td>
<td>- awk</td>
</tr>
<tr>
<td></td>
<td>- ifconfig</td>
</tr>
<tr>
<td>Optional services</td>
<td>If you plan to remotely access the virtual machine using Linux ssh logging or Windows remote desktop for troubleshooting or for other reasons, the OpenSSH server and client for Linux or Remote Desktop Services (RDS) for Windows must be installed and running properly.</td>
</tr>
</tbody>
</table>

**IMPORTANT** Because the boot process must not be interrupted, configure the virtual machine so that nothing causes the virtual machine’s boot process to pause before reaching the final operating system login prompt. For example, verify that no processes or scripts prompt for user interaction when the virtual machine starts. This requirement applies only to virtual machine templates created for the vFabric Application Director catalog.

Creating Windows Virtual Machine Templates in vCloud Director

With Windows virtual machine templates, you can use vFabric Application Director on vCloud Director 1.5, 1.5.1, or 5.1.1 to create an application blueprint and deploy Windows-based services and applications such as .NET applications.

Before you create a Windows virtual machine template, you must set up the vCloud Director environment.

Create and Configure Windows Virtual Machine Templates

When you create a Windows virtual machine template in vCloud Director you must install the supported operating system, Java Runtime Environment, agent bootstrap service, and VMware Tools.

**Prerequisites**

- Verify that vCloud Director 1.5, 1.5.1, or 5.1.1 is installed and configured.
- Verify that the vCloud organization that you plan to use with vFabric Application Director is created and configured with a user account that has privileges to create and share vCloud Director catalog templates.
- Familiarize yourself with how to create a vApp in vCloud Director. See vCloud Director documentation.
- For the recommended storage and memory values and Remote Desktop Services for set up instructions, see the Windows Server 2008 R2 Enterprise with Service Pack 1 system requirements.
- Familiarize yourself with how to install and configure VMware Tools. See the *Installing and Configuring VMware Tools* guide.

**Procedure**

1. Log in vCloud Director and create a vApp with one virtual machine.
   
   Allocate the appropriate virtual storage and virtual memory to support the Windows Server 2008 R2 Enterprise with Service Pack 1 operating system and your application.

2. Install Windows Server 2008 R2 Enterprise with Service Pack 1 in the virtual machine, using the **Full Installation** option.
3 Install VMware Tools in the virtual machine. The VMware Tools version must be from vCloud Director 1.5, 1.5.1, or 5.1.1.

4 Log in to the Windows virtual machine as a user belonging to the local Administrators group.

5 Download and install the supported Java SE 6 Runtime Environment from http://Application_Director_IP/agent/jre-1.6.0_31-win64.zip.
   a Create a DirectoryPath\java folder and unzip the JRE file to the folder.
   b Select Start and right-click the Computer icon.
   c Select Properties > Advanced System Settings > Advanced Tab > Environment Variables.
   d Double-click the Path variable from the System variables list and append ;DirectoryPath\java\bin to the string at the end of the variable value.
      For example, if you saved the JRE folder in the C:\ directory, append the variable string as ;C:\java\bin.
   e Confirm and close all of the dialog boxes.
   f Open a PowerShell command window and type java --version to verify the installation.
      The installed version of Java appears.

6 Download and install the vFabric Application Director agent bootstrap file from http://Application_Director_IP/agent/vmware-appdirector-agent-bootstrap-windows_5.0.0.0.zip.
   a Unzip the vmware-appdirector-agent-bootstrap-windows_5.0.0.0.zip file to a new folder.
   b Add the NTRights.exe utility to the newly created folder.
      The NTRights.exe utility is used in the install.bat command to configure the agent bootstrap service to run in the darwin user account. The utility is included in the Windows Server 2003 Resource Kit Tools (rktools.exe), which you can download from the Microsoft download Web site.
   c (Optional) If the NTRights.exe utility is not available, manually configure the agent bootstrap service to run in a specific user account after you run the install.bat command.
   d To install the agent bootstrap service, open a Windows CMD console, navigate to the new folder, and type install.bat Password.
      The Password must meet the Windows password requirements. The install.bat command creates a user account called darwin for the agent bootstrap service.
   e From the command-line, type services.msc and open the VMware vFabric Application Director agent bootstrap service.
   f On the Log On tab, set the Log on as option to This account and type the login credentials.
      The user name for this user account is .\darwin.

7 Open a PowerShell command window and type net start AppDAgentBootstrap to verify that the service starts successfully.

8 Type net stop AppDAgentBootstrap to stop the service.

9 Delete the log files generated during this verification process from the c:\opt\vmware-appdirector\bootstrap.log directory.

10 (Optional) Enable Remote Desktop to remotely access future instances of this virtual machine.
    You can also use the remote access for troubleshooting purposes.

11 Shut down the Windows virtual machine and stop the vApp in vCloud Director.
What to do next

Before you save your vApp to a catalog in vCloud Director, you can also select the Change SID and Join Domain features to allow the Windows template to become a member of a Windows Active Directory domain. See “Enable SID Change and Domain Join for Windows Virtual Machine Templates,” on page 43.

Add the vApp template to the vCloud Director catalog so that you can register it for use in vFabric Application Director. See “Add Windows Virtual Machine Templates to the vCloud Director Catalog,” on page 44.

Enable SID Change and Domain Join for Windows Virtual Machine Templates

You can use Change SID and enable Join Domain for a Windows virtual machine template. With the Change SID option, all of the deployed virtual machines can acquire a unique security identifier (SID). Enable the option when you prepare a Windows virtual machine template. With the Join Domain option, the deployed virtual machines can become members of a specified Windows Active Directory domain.

If you need to deploy a large number of Windows virtual machines that must join a corporate managed domain, use Windows templates with preconfigured domain settings. For smaller deployments or deployments that need the flexibility of joining different domains, a more effective approach is to use Windows templates without the preconfigured domain settings and add the Join Domain Predefined Task in the execution plan for deployment. See “Add Join Domain Predefined Task,” on page 122.

Prerequisites

- Verify that you have correctly configured, created, and registered the Windows template. See “Add Windows Virtual Machine Templates to the vCloud Director Catalog,” on page 44.
- Verify that the Windows domain name, domain account, and account password are readily available.
- Verify that the virtual machine can connect to the designated domain controller.
  
  A DHCP service must be present on the network and the DNS server specified by the DHCP can resolve the domain name specified in the guest customization settings.

Procedure

1. Instantiate a vApp from the source template and locate the associated Windows virtual machine.
2. Log in to the Windows virtual machine as a user belonging to the local Administrators group.
3. Configure all of the virtual machine network interfaces to use DHCP.
   a. Open the Network control panel.
   b. Select Network and Sharing Center > Change Adapter Settings.
   c. Open each network connection and select Properties.
   d. Set the Internet Protocol Version 4/6 to Obtain an IP address Automatically and Obtain DNS server address automatically.
4. Shut down the Windows virtual machine and stop the vApp in vCloud Director.
5. In vCloud Director, open the properties for the virtual machine and select the Guest OS Customization tab to enable Guest Customization.
6. On the Guest OS Customization tab, select Change SID.
7. Select Join Domain and type the Windows domain name, domain account, and account password.
8. Select Add to Catalog to create a new vApp template from the vApp.

During deployment, Active Directory authenticates the login credentials and allows the Windows virtual machine to become a member of the domain.
What to do next
Add the vApp template to the vCloud Director catalog. See “Add Windows Virtual Machine Templates to the vCloud Director Catalog,” on page 44.

Add Windows Virtual Machine Templates to the vCloud Director Catalog
Add the Windows virtual machine template to the vCloud Director Catalog and register it so that it works with vFabric Application Director.

Prerequisites
- Verify that your user account has the ROLE_CLOUD_ADMIN cloud administrator role assigned to it.
- Verify that vCloud Director 1.5, 1.5.1, or 5.1.1 is installed and configured.
- Verify that the Windows template meets the vFabric Application Director virtual machine requirements. See “Virtual Machine Requirements for Creating Custom Templates,” on page 39.
- Complete the preparation tasks required to create a Windows template. See “Create and Configure Windows Virtual Machine Templates,” on page 41.
- Familiarize yourself with the process of registering a cloud template in vFabric Application Director. See “Register the vCloud Director Cloud Provider and Template,” on page 52.

Procedure
1. Log in to the vCloud Director Web user interface.
2. Navigate to the vCloud Org location where the Windows vApp resides.
3. Select the vApp and click Add to Catalog to create a vApp template in the catalog.

The Windows template is added to the list of cloud templates that you can choose from when you map a logical template to a cloud template.

What to do next
Register the Windows template for use in vFabric Application Director. See “Register the vCloud Director Cloud Provider and Template,” on page 52.

Create Linux Virtual Machine Templates in vCloud Director
With Linux virtual machine templates, you can use vFabric Application Director on vCloud Director 1.5, 1.5.1, or 5.1.1 to create an application blueprint and deploy Linux-based services and applications.

Prerequisites
- Verify that vCloud Director 1.5, 1.5.1, or 5.1.1 is installed and configured.
- Verify that the vCloud organization that you plan to use with vFabric Application Director is created and configured with a user account that has privileges to create and share vCloud Director catalog templates.
- Familiarize yourself with how to create a vApp in vCloud Director. See vCloud Director documentation.
  Creating multiple virtual machine vApp templates is not supported.
- Familiarize yourself with how to install and configure VMware Tools. See the Installing and Configuring VMware Tools guide.
Procedure

1. Log in to vCloud Director and create a vApp with one virtual machine.
   Allocate the appropriate virtual storage and virtual memory to support the Linux operating system and your application.

2. Install the supported Linux operating system in the virtual machine.

3. Install VMware Tools in the virtual machine.
   The VMware Tools version must be from vCloud Director 1.5, 1.5.1, or 5.1.1.

4. Log in the virtual machine with root privileges and open a terminal.

5. For an RPM-based virtual machine, download and install the supported JRE packages.
   a. Change directory to /tmp/ folder.
   b. Download the JRE package from http://Application_Director_IP/agent/jre-6u31-linux-ArchName.rpm.
   c. Type `rpm -i jre-6u31-linux-ArchName.rpm` to install the package.

6. For a DEB-based virtual machine, download and install the supported JRE packages.
   a. Remove any existing JRE packages from the /usr/java/default, /usr/java/latest, and /usr/java/jre1.6.0_31 directories.
   b. Create a folder called java in the /usr/ directory.
   c. Download the JRE package to the /tmp/ folder and type `wget http://Application_Director_IP/agent/jre-6u31-linux-ArchName.bin`.
   d. Set file permissions on the JRE binary file for execute permission `chmod +x jre-6u31-linux-ArchName.bin`.
   e. Change the working directory to /usr/java and run the self-extracting archive `/tmp/jre-6u31-linux-ArchName.bin`
   f. Create a symbolic link between the jre1.6.0_31 and latest directories `ln -s jre1.6.0_31 latest` and a symbolic link between the latest and default directories `ln -s latest default`.
   g. Remove the JRE file and verify that the java 1.6.0_31 version is available at /usr/java/default/bin/java --version.
      `rm jre-6u31-linux-ArchName.bin`
      The vFabric Application Director agent bootstrap specifically searches the /usr/java/default/bin/java folder for the java binary files. No further configuration is needed in the system environment variables such as PATH, JAVA_PATH, CLASSPATH.

7. Download and install the vFabric Application Director agent bootstrap service.
   a. Change directory to /tmp/ folder.
   b. For an RPM-based virtual machine, download the agent bootstrap from http://Application_Director_IP/agent/vmware-appdirector-agent-service_5.0.0.0-0_ArchName.rpm to the folder and type `rpm -i vmware-appdirector-agent-service_5.0.0.0-0_ArchName.rpm` in the terminal to install the service.
   c. For a DEB-based virtual machine, download the agent bootstrap from http://Application_Director_IP/agent/vmware-appdirector-agent-service_5.0.0.0-0_ArchName.deb to the folder and type `dpkg -i vmware-appdirector-agent-service_5.0.0.0-0_ArchName.deb` in the terminal to install the service.

8. Shut down the Linux virtual machine and stop the vApp in vCloud Director.

9. Navigate to the vCloud Org location where the Linux vApp resides.
10 Select the vApp and click **Add to Catalog** to create a vApp template in the catalog.

**What to do next**

Register the Linux template for use in vFabric Application Director. See “Register the vCloud Director Cloud Provider and Template,” on page 52.

**Verifying Configuration of a Cloud Template from the vCloud Director Catalog**

After you upload the virtual machine template to vCloud Director and register it as a vApp template in a vCloud Director catalog, verify the configuration before you use the template in vFabric Application Director.

Use the vCloud Director Web interface to manually instantiate a request to create a copy of the vApp to your cloud environment. For instructions, see the vCloud Director documentation.

Certain items must be functioning correctly in the instantiated vApp.

**Table 6-2. Verify Cloud Template Configuration**

<table>
<thead>
<tr>
<th>Item</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guest customization</td>
<td>To verify that the customization process was successful, open the <code>/var/log/vmware-imc/customization.log</code> file and make sure it says that the customization completed with a status of success. Check that an active and correct IP address exists for the IP assignment from vCloud Director. See the vCloud Director documentation. If your cloud template has a customization script, vFabric Application Director overwrites the script with its own guest customization script. You use the guest customization script to set up the virtual machine so that it can communicate with the vFabric Application Director server to complete the deployment process.</td>
</tr>
<tr>
<td>Agent bootstrapping</td>
<td>Open and examine the log file located at <code>/opt/vmware-appdirector/agent/logs/agent_bootstrap.log</code>. Verify that the vFabric Application Director agent boot service ran and attempted to download the vFabric Application Director agent JAR file from the vFabric Application Director server. The attempt to download the JAR file is expected to fail at this point. The download process is successful when vFabric Application Director deploys the vApp from a vFabric Application Director execution plan.</td>
</tr>
<tr>
<td>Required components</td>
<td>Verify that the items listed in “Virtual Machine Requirements for Creating Custom Templates,” on page 39 are present in the vApp and that they function properly.</td>
</tr>
</tbody>
</table>

**Create Amazon EC2 Virtual Machine Templates or AMIs**

To use Amazon AMIs in vFabric Application Director you must create an AMI by customizing an existing AMI.

**IMPORTANT** To deploy an application to Amazon EC2, you must install the vFabric Application Director for Release Automation edition.

**Prerequisites**

- Familiarize yourself with the AWS management console and the steps to launch, configure, and terminate an Amazon EC2 instance or virtual machine.
- Verify that you have the proper credentials to access the AWS management console.
- Identify the AWS Region where the vFabric Application Director applications will run and confirm that the AMIs are accessible to the Region.
- Verify that the Amazon EC2 AMI meets the vFabric Application Director virtual machine requirements. See “Virtual Machine Requirements for Creating Custom Templates,” on page 39. For more information about using AMIs, see AWS Documentation.
Procedure

1. Log in the AWS console and locate an EBS-backed AMI in the appropriate Amazon Region to deploy applications.

2. Use the AMI to start an Amazon EC2 instance and log in as a root user.

3. Download and install the supported JRE packages for an RPM-based AMI.
   a. On a machine in the same network as vFabric Application Director, download the JRE package from http://Application_Director_IP/agent/jre-6u31-linux-ArchName.rpm.
   b. Upload the JRE package to the Amazon EC2 instance.
   c. From the Amazon EC2 instance, type `rpm -i jre-6u31-linux-ArchName.rpm` to install the package.

4. Download and install the vFabric Application Director agent bootstrap service for an RPM-based AMI.
   a. On a machine in the same network as vFabric Application Director, download the agent bootstrap service from http://Application_Director_IP/agent/vmware-appdirector-agent-service-ec2_5.0.0.0-0_ArchName.rpm.
   b. Upload the agent bootstrap service to the Amazon EC2 instance.
   c. From the Amazon EC2 instance, type `rpm -i vmware-appdirector-agent-service-ec2_5.0.0.0-0_ArchName.rpm` to install the service.

5. (Optional) Add a second Ethernet device eth1 on the CentOS operating system.
   a. Locate the `ifcfg-eth0` file.
   b. Copy the contents of the file to the `ifcfg-eth1` file.
   c. In the `ifcfg-eth1` file, replace all of the `DEVICE=eth0` strings with `DEVICE=eth1` and save your changes.

6. (Optional) Erase the Linux command history to secure the AMI you are going to create.
   a. Delete the Linux command history saved in the `.bash_history` file.
      ```
      cat /dev/null > ~/.bash_history
      ```
   b. Delete the Linux command history in the current Bash login session.
      ```
      history -c
      ```

7. In the AWS console, stop the Amazon EC2 instance.
   IMPORTANT If you terminate the instance, you lose all of your changes.

8. To create an Amazon EBS-backed AMI, select the stopped Amazon EC2 instance, provide the Image Name, and initiate the process.

   The AMI creation might take several minutes.

   After you create the AMI you can terminate the Amazon EC2 instance.

What to do next

Prepare the Amazon EC2 environment to establish a secure connection with the vFabric Application Director appliance. See “Setting Up Cloud Environments,” on page 49.
Updating Existing Virtual Machine Templates

To update the content of an existing Linux or Windows template, or to use it for creating a new template, you must run applicable commands to reset or stop the agent bootstrap service.

For a Linux template, the `agent_reset.sh` command resets the vFabric Application Director agent bootstrap status. You can log in to the virtual machine as root and run this command:

```
/opt/vmware-appdirector/agent-bootstrap/agent_reset.sh
```

For a Windows template, in a PowerShell command window type `net stop AppDAgentBootstrap` to stop the agent bootstrap service. Delete all of the contents in the `C:/opt/vmware-appdirector/agent` directory. Remove all of the `.log` file extensions from the `C:/opt/vmware-appdirector/agent` directory.

After you run the applicable command, you can place the vApp instance back in the catalog as a new vApp template.

Exporting Virtual Machine Templates with OVF Format

With vFabric Application Director you can create virtual machine templates in the OVF format and export the templates to a vCloud Director catalog.

If you plan to export virtual machine templates in the OVF format to a vCloud Director catalog, do not add custom OVF properties to the export packages. Custom OVF properties disable the Guest OS Customization feature in vCloud Director. With this feature disabled, vFabric Application Director cannot access virtual machine templates.
When you prepare to deploy applications using vFabric Application Director 5.0, you must set up one or more cloud environments. Your cloud environment can be either a vCloud environment or an Amazon EC2 environment. As part of the configuration process, you must register a cloud provider, map a cloud template or AMI to a logical template in vFabric Application Director, and create a deployment environment.

vFabric Application Director supports vCloud Director 1.5, 1.5.1, 5.1.1, and Amazon EC2 VPC cloud providers. Registering a cloud provider involves specifying either a vCloud Director organization or an Amazon EC2 Region. Registering a template links a cloud template or AMI from the cloud environment to a logical template in the vFabric Application Director catalog. Creating a deployment environment in this cloud provider involves selecting a contained vCloud Director organization vDC or a combination of an Amazon Virtual Private Cloud (VPC) and Availability Zone.

If a cloud provider has deployment environments mapped to it, then that cloud provider cannot be deleted. When a cloud provider is deleted, all of the mappings to the cloud templates of the cloud provider from the logical templates are also removed.

Familiarize yourself with the key concepts relating to setting up and configuring a cloud environment. See “Key Concepts,” on page 12.

This chapter includes the following topics:

- “Setting Up Cloud Environments,” on page 49
- “Registering a Cloud Provider and Template,” on page 52
- “Create a Cloud Tunnel to Connect to Amazon EC2,” on page 55
- “Creating a Deployment Environment,” on page 56

Setting Up Cloud Environments

To deploy applications using vFabric Application Director, you must set up one or more cloud environments in a vCloud Director environment or an Amazon EC2 environment.

As part of setting up a vCloud Director environment, you must install vCloud Director and create a vCloud Director organization with one or more user accounts and an organization vDC to allow vFabric Application Director to create cloud providers and register them to the organization vDC. See “Set Up vCloud Director for vFabric Application Director,” on page 19.

Setting up an Amazon EC2 cloud environment involves signing up for an Amazon Web Services (AWS) user account and creating and configuring a VPC in an Amazon Region. With vFabric Application Director, you can create cloud providers and link them to a VPC in an Amazon Region.

You must create virtual machine templates in the vCloud Director and Amazon EC2 environments. These templates are used during application deployment to create virtual machines. See Chapter 6, “Creating and Updating Custom Virtual Machine Templates,” on page 39.
To deploy a vFabric Application Director application to Amazon EC2, you must create a special virtual machine in the VPC, called an Endpoint VM. The Endpoint VM serves as the destination of a secure channel, called a cloud tunnel, from the vFabric Application Director appliance in your corporate network. Application deployments in the VPC communicate with the vFabric Application Director appliance through the Endpoint VM.

You can manage the cloud tunnels using the CLI. See “Managing Cloud Tunnels,” on page 145.

Configure Amazon EC2 Environment for vFabric Application Director

Setting up an Amazon EC2 environment requires you to create a VPC as a target location in an Amazon Region for application deployments. You must configure this VPC for vFabric Application Director applications to be deployed.

**IMPORTANT** To deploy an application to Amazon EC2, you must install the vFabric Application Director for Release Automation edition.

For deployments to Amazon EC2, NICs on the external networks receive Elastic IP addresses. In addition, a new security group is created for each deployment to allow communication between the instances in the deployment. This security group allows external access to 80, 8080, 8081, 8443, and 22 ports. For any other ports that you need to open, use the Amazon EC2 management console to locate the new security group and add the appropriate rules.

**Prerequisites**

- Set up an Amazon AWS user account.
- The default limits for Amazon EC2 instances in a Region is 20 and for Elastic IP addresses is 5. Contact Amazon support to request an increase in the instance or Elastic IP address limit. For more information on Amazon EC2, see Amazon AWS Documentation.
Procedure

1. Log in to the AWS Management Console using your AWS user account.

2. On the AWS Identity and Access Management (IAM) Dashboard, set up a new user with either an Administrator Access or Power User Access user privilege and generate an access key for this user.
   
   You can also set up an access key with the master AWS user account without setting up a new IAM user.

3. Download and save this access key for later use with vFabric Application Director.

4. To log in the deployed virtual machines in Amazon EC2, generate an SSH key-pair called titan_keypair and save the corresponding private key.

5. On the VPC Dashboard of the AWS Management Console, create a VPC with a public subnet and a private subnet.
   
   a. Use the wizard option to create a VPC with public and private subnets.
      
      Two subnets are created whose instances have access to the Internet. The subnets are useful for downloading content during the application deployment.
   
   b. Specify an Availability Zone for the subnets.
      
      You can use the same Availability Zone for both subnets.
   
   c. Use the titan_keypair SSH key-pair.
      
      You can also use your own key-pair.

6. In the VPC, create a security group called EndpointAccess and set inbound and outbound port access rules.

<table>
<thead>
<tr>
<th>Port Access</th>
<th>Source or Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound 22</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>Inbound ALL</td>
<td>SubnetofVPC</td>
</tr>
<tr>
<td>Outbound ALL</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>(Optional) Inbound 2222</td>
<td>0.0.0.0/0</td>
</tr>
</tbody>
</table>

What to do next

Create a Linux virtual machine in your VPC. See “Create an Endpoint VM,” on page 51.

Create an Endpoint VM

Create a Linux virtual machine in Amazon EC2 called Endpoint VM in your VPC to deploy applications using vFabric Application Director.

A vFabric Application Director appliance can work with multiple Endpoint VMs. However, an Endpoint VM can only be used by a single vFabric Application Director appliance.

Prerequisites

- Verify that an Amazon AWS user account is set up.
- Verify that the VPC is available as a target location in an Amazon Region for application deployments. See “Configure Amazon EC2 Environment for vFabric Application Director,” on page 50.

Procedure

1. Install one Endpoint VM in each VPC to manage using vFabric Application Director.
2 Create an Amazon-based CentOS virtual machine on the external subnet.
   a Use either the amazon/ami-vpc-nat-1.0.0-beta.i386-ebs AMI or the amazon/ami-vpc-nat-1.0.0-
      beta.x86_64-ebs AMI.
   b Use the EndpointAccess security group for this virtual machine.
3 Set an Elastic IP address to use in the VPC and assign the IP address to the Endpoint VM.
4 Use an OpenSSH client to log in to the Endpoint VM with the Elastic IP address and the private key.
   ssh -i PathToPrivateKeyFile ec2-user@ApplicationDirectorEndpointVM
5 In the CLI, open the /etc/ssh/sshd_config configuration file, add the GatewayPorts yes,
   ClientAliveInterval 30, and ClientAliveCountMax 3 lines.
6 (Optional) If the outbound SSH is blocked by your corporate firewall, ask your firewall administrator to
   set an alternative port to 2222 for SSH traffic.
   Add the line Port 2222 to the /etc/ssh/sshd_config configuration file.
7 Restart the SSH daemon.
   sudo service sshd restart
8 Create an iptable rule to reroute the internal port 80 access to 8080.
   sudo iptables -t nat -I PREROUTING --source 0/0 --destination internal-ip-address-of-endpoint-
   vm -p tcp --dport 80 -j REDIRECT
   --to-ports 8080
9 Log out of the Endpoint VM.

What to do next
Map the Amazon EC2 Region to a vFabric Application Director cloud provider and provide access to the AMIs
or virtual machine templates. See “Register the Amazon EC2 Cloud Provider and Template,” on page 54.

Registering a Cloud Provider and Template

vCloud Director and Amazon EC2 provision the cloud infrastructure for vFabric Application Director to
deploy predefined or custom applications.

You can create a custom virtual machine template or AMI in your cloud environment. When you register a
custom virtual machine template, you create a mapping for the logical items in the
vFabric Application Director catalog to use later.

Register the vCloud Director Cloud Provider and Template

vFabric Application Director registers a cloud provider by connecting to a vCloud Director instance and
organization. A catalog that is accessible to the vCloud Director user can provide access to virtual machine
templates. The vCloud organization’s vDCs map to deployment environments.

You cannot delete a cloud provider that is linked to a deployment environment. When you delete a cloud
provider that is not linked to a deployment environment, all of the cloud template mappings from logical
templates are removed.

Prerequisites

- Verify that your user account has the ROLE_CLOUD_ADMIN cloud administrator role assigned to it.
- Verify that vCloud Director 1.5, 1.5.1, or 5.1.1 is installed and configured.
- Create virtual machine templates that meet vFabric Application Director requirements. See “Virtual
  Machine Requirements for Creating Custom Templates,” on page 39.
- Verify that the vCloud organization that you plan to use with vFabric Application Director is created and configured with organization administrator privileges.

- When you map cloud templates for a cloud provider in vFabric Application Director, verify that the vCloud Director template that you are mapping to is a single virtual machine. Multiple virtual machine vApp templates are not supported.

  Templates with multiple virtual machines cannot be registered to a vFabric Application Director cloud provider.

For information about setting up vCloud Director for use with vFabric Application Director, see “Preparing to Install vFabric Application Director,” on page 17.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Clouds > Cloud Providers**.
2. Click **New** in the toolbar.
3. Complete the cloud provider information.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name and Description</strong></td>
<td>Include the vCloud Director organization name in either of these fields. The text from these fields appears in the Deployment Profile wizard under the Cloud Provider column.</td>
</tr>
<tr>
<td><strong>Cloud Provider Type</strong></td>
<td>vCloud Director 1.5, 1.5.1, or 5.1.1 is the supported type.</td>
</tr>
<tr>
<td><strong>Cloud IP/Hostname</strong></td>
<td>The cloud IP or host name must match the IP address or host name of the vCloud Director instance.</td>
</tr>
<tr>
<td><strong>Organization Name</strong></td>
<td>The organization name must match the name of an organization in vCloud Director.</td>
</tr>
</tbody>
</table>
| **User Name and Password** | Credentials for an organization user with organization administrator privileges for that organization. This user must not be the vCloud Director system administrator.  
  **Note:** If the user name includes an @ symbol, replace it with %40 when you provide the login credentials. |

4. To test whether the values you entered are correct, click **Validate Connection**.
5. Click **Save** to register the cloud provider.

   If required entries for the cloud provider were invalid or left blank, you are prompted to correct them when you click **Save**.

6. To register vCloud Director virtual machine templates, click **Edit** in the toolbar.
7. In the Templates section, click **New** to register templates.
8. Select a catalog from the drop-down menu and click **Get Templates**.

    Templates with multiple virtual machines cannot be registered to a vFabric Application Director cloud provider.

    Cloud templates with a single virtual machine appear in the drop-down menu.

9. Select the check boxes next to one or more templates in the list, and click **OK**.

    The templates you registered are added to the list of cloud templates that you can choose from when you map a logical template to a cloud template.

10. To change existing information for a vCloud Director cloud provider, click **Edit** in the toolbar, make your changes, and click **Save**.
The cloud provider is registered and its virtual machine templates and virtual datacenters are available for use in vFabric Application Director.

**What to do next**

Map these vCloud Director cloud templates to the vFabric Application Director logical templates. See “Add a Logical Template to the Catalog,” on page 69.

Create a deployment environment to deploy an application to the vCloud Director environment. See “Create a vCloud Director Deployment Environment,” on page 56.

**Register the Amazon EC2 Cloud Provider and Template**

In vFabric Application Director 5.0, registering a cloud provider means connecting to an Amazon EC2 Region using access keys established in a valid AWS account.

**IMPORTANT** To deploy an application to Amazon EC2, you must install the vFabric Application Director for Release Automation edition.

**Prerequisites**

- Verify that your user account has the ROLE_CLOUD_ADMIN cloud administrator role assigned to it.
- Verify that an AWS user account is set up with access keys.
- Create virtual machine templates that meet vFabric Application Director requirements. See “Virtual Machine Requirements for Creating Custom Templates,” on page 39.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Clouds > Cloud Providers**.
2. Click **New** in the toolbar.
3. Complete the cloud provider information.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name and Description</strong></td>
<td>Include detailed information in the Description text box. The text from the text box appears in the Deployment Profile wizard under the Cloud Provider column.</td>
</tr>
<tr>
<td><strong>Cloud Provider Type</strong></td>
<td>Amazon EC2 is the supported type.</td>
</tr>
<tr>
<td><strong>Amazon Region</strong></td>
<td>Specify the Region name. For example, if your Region is US East (N. Virginia) the Amazon Region is <strong>us-east-1</strong>.</td>
</tr>
<tr>
<td><strong>Proxy URL</strong></td>
<td>If your corporate network requires a proxy server for external connections, add the Web proxy URL to enable the vFabric Application Director server to connect to Amazon EC2.</td>
</tr>
<tr>
<td><strong>User Name and Password</strong></td>
<td>Access keys set up on the AWS account.</td>
</tr>
</tbody>
</table>

4. To test whether the values you entered are correct, click **Validate Connection**.
5. Click **Save** to register the cloud provider.
6. To register Amazon AMIs, click **Edit** in the toolbar.
7. In the Templates section, click **New** to continue and register AMIs or virtual machine templates.
8 Select a category from the drop-down menu and click **Get Templates**.

The category choices let you select AMIs owned by the AWS account or those that are private to the AWS account.

Filtered AMIs or cloud templates that you created or private to the AWS account appear in the drop-down menu.

9 Select the check boxes next to one or more templates in the list, and click **OK**.

The templates you registered are added to the list of cloud templates that you can choose from when you map a logical template to a cloud template.

10 To change existing information for an Amazon EC2 cloud provider, click **Edit** in the toolbar, make your changes, and click **Save**.

The Amazon EC2 Region is registered and its AMIs are available for use in vFabric Application Director.

**What to do next**

Map these AMIs to the vFabric Application Director logical templates. See “Add a Logical Template to the Catalog,” on page 69.

Set up a cloud tunnel for Amazon EC2 deployments to communicate with the vFabric Application Director server. See “Create a Cloud Tunnel to Connect to Amazon EC2,” on page 55.

### Create a Cloud Tunnel to Connect to Amazon EC2

You must create and enable a cloud tunnel instance so that deployments in the Amazon EC2 VPC can communicate with the vFabric Application Director server.

**IMPORTANT** To deploy an application to Amazon EC2, you must install the vFabric Application Director for Release Automation edition.

Your network from vFabric Application Director to the Endpoint VM should have a minimum upload bandwidth of 1Mbps for every Amazon EC2 instance that is deployed. For faster downloads, store your applications on the Amazon Simple Storage Service instead of downloading them from the vFabric Application Director appliance over the cloud tunnel.

**Prerequisites**

- Verify that your user account has the **ROLE_CLOUD_ADMIN** cloud administrator role assigned to it.
- Verify that the Endpoint VM is properly set up and configured. See “Configure Amazon EC2 Environment for vFabric Application Director,” on page 50.
- Verify that at least one Amazon cloud provider is registered in vFabric Application Director. See “Register the Amazon EC2 Cloud Provider and Template,” on page 54.
- Verify that the elastic IP address and the private IP address of the Endpoint VM are readily available.
- Verify that the private key for the Endpoint VM is available to establish a cloud tunnel from the corporate network to the Endpoint VM.
- Determine whether a proxy server is required to access Amazon EC2 from the network where vFabric Application Director is running.
  
  The proxy server or the network must permit access to the standard SSH port 22 outside the network.
- Start the vFabric Application Director CLI. See “Start the CLI in an Appliance,” on page 30 and “Start the CLI Remotely,” on page 31.
Procedure

1. Use the SSH client to copy the downloaded private key file for the Endpoint VM to the vFabric Application Director appliance and copy the file to the /tmp directory.

2. In the root shell, create a secure cloud tunnel instance.

   ```
   create-cloud-tunnel --name TunnelName --description "TunnelDescription" --enabled false --
   externalAddress EndpointVMPublicIP
   --sshPort 22 --internalAddress EndpointVMPrivateIP --proxyUrl ProxyURL --username ec2-user --
   privateKeyPath PrivateKeyFilePath
   ```

   You can use the --sshPort parameter to designate a port other than 22. The --proxyUrl parameter that you can specify the proxy server to use to connect to the Endpoint VM.

3. Enable the secure cloud tunnel connection.

   ```
   enable-cloud-tunnel --name TunnelName
   ```

4. Confirm that the secure cloud tunnel connection is established.

   ```
   test-cloud-tunnel --name TunnelName
   ```

Verifying the secure cloud tunnel connection might take a few minutes.

vFabric Application Director establishes a connection to the Endpoint VM on Amazon EC2.

What to do next

Create a deployment environment to deploy an application to the vCloud Director environment. See “Create an Amazon EC2 Deployment Environment,” on page 57.

You can manage the existing cloud tunnel connection in CLI. See “Managing Cloud Tunnels,” on page 145.

Creating a Deployment Environment

A deployment environment is required to deploy an application to the cloud. The deployment environment varies between cloud types.

Before you create a deployment environment in vFabric Application Director, you need cloud-specific information to help you define required fields such as Amazon VPC and associated Availability Zone or a vCloud Director organization vDC.

Create a vCloud Director Deployment Environment

You must map a deployment environment to a vCloud Director organization vDC before you can deploy an application.

vCloud Director vDCs provide an environment where virtual systems can be stored, deployed, and operated. For example, you might have separate deployment environments for development, testing, staging, and production. For vCloud Director, a deployment environment is mapped to an organization vDC.

If a deployment environment is currently being used in a deployment profile or a current deployment, you cannot delete it.

Prerequisites

- Verify that your user account has the ROLE_CLOUD_ADMIN cloud administrator role assigned to it.
- Verify that at least one cloud provider is registered in vFabric Application Director. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
Verify that an organization vDC is created and configured in the vCloud Director organization. The vFabric Application Director deployment environment is equivalent to the organization vDC in vCloud Director.

For information about setting up vCloud Director for use with vFabric Application Director, see “Preparing to Install vFabric Application Director,” on page 17.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select Clouds > Deployment Environments.

2. Click New in the toolbar.

3. Complete the deployment environment information.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deployment Environment name and Description</td>
<td>Include the vDC name in either of these fields. The text from these fields appears in the Deployment Profile wizard under the Deployment Environment column.</td>
</tr>
<tr>
<td>Cloud Provider</td>
<td>If the cloud provider does not appear in the list, cancel the dialog box and select Clouds &gt; Cloud Providers to add the cloud provider.</td>
</tr>
<tr>
<td>Organization vDC URI</td>
<td>Click Select to select from a list of vDCs that the cloud provider you selected provides.</td>
</tr>
</tbody>
</table>

4. Select an organization vDC and click OK.

A vCloud Director organization vDC is mapped to the vFabric Application Director deployment environment name.

5. Click Save.

6. To change existing information for a vCloud Director deployment environment, click Edit in the toolbar, make your changes, and click Save.

The deployment environment you created is added to the list of deployment environments that you can select from when you create a deployment profile.

**Create an Amazon EC2 Deployment Environment**

You must map a deployment environment to an Amazon EC2 VPC and associated Availability Zone before you can deploy an application to the cloud.

With a VPC, you can provision a private and isolated area of the AWS Cloud where you can deploy applications from vFabric Application Director.

If a deployment environment is currently being used in a deployment profile or a current deployment, it cannot be deleted.

**IMPORTANT** To deploy an application to Amazon EC2, you must install the vFabric Application Director for Release Automation edition.

**Prerequisites**

- Verify that your user account has the ROLE_CLOUD_ADMIN cloud administrator role assigned to it.
- Verify that at least one Amazon cloud provider is registered in vFabric Application Director. See “Register the Amazon EC2 Cloud Provider and Template,” on page 54.
- Verify that the Amazon EC2 environment is configured to establish a secure connection with the vFabric Application Director appliance. See “Configure Amazon EC2 Environment for vFabric Application Director,” on page 50.
**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Clouds > Deployment Environments**.
2. Click **New** in the toolbar.
3. Complete the deployment environment information.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deployment Environment and Description</td>
<td>Include the name of the VPC and the Availability Zone name in either of these fields. The text from these fields appears in the Deployment Profile wizard under the Deployment Environment column.</td>
</tr>
<tr>
<td>Cloud Provider</td>
<td>If the cloud provider does not appear in the list, cancel the dialog box and select <strong>Clouds &gt; Cloud Providers</strong> to add the cloud provider.</td>
</tr>
<tr>
<td>Virtual Private Cloud</td>
<td>Click <strong>Select</strong> to view a list of available Amazon VPCs in the Region.</td>
</tr>
<tr>
<td>Availability Zone</td>
<td>Click <strong>Select</strong> to view a list of Availability Zones in the Amazon Region.</td>
</tr>
<tr>
<td>Cloud Tunnel</td>
<td>Select the cloud tunnel instance that connects the vFabric Application Director server to an Endpoint VM. This Endpoint VM must reside in the Amazon EC2 VPC already set in the Virtual Private Cloud section for this deployment environment. Click <strong>Validate</strong> to check the secure SSH connection status of the cloud tunnel.</td>
</tr>
</tbody>
</table>

4. Click **Save**.

5. To change existing information for an Amazon EC2 deployment environment, click **Edit** in the toolbar, make your changes, and click **Save**.

The deployment environment is added to the list of deployment environments that you can select from when you create a deployment profile.
Developing vFabric Application Director Components

vFabric Application Director provides an open framework to create and develop components that can be installed in a virtual machine.

The vFabric Application Director components in the catalog include predefined sample services such as JBoss and MySQL, predefined tasks such as scripts to configure the APT repository, and blueprint application components such as WAR and SQL_SCRIPT.

In this information, components include services, application components, and custom tasks. All of these components use actions and properties as their common underlying framework.

**NOTE** Only use the sample components in the vFabric Application Director catalog in a test environment.

Familiarize yourself with the key concepts that appear frequently in topics about developing deployable components. See “Key Concepts,” on page 12.

This chapter includes the following topics:
- “Defining Component Actions,” on page 59
- “Configuring Component Properties,” on page 61
- “vFabric Application Director Predefined Properties,” on page 66
- “Secured Component Properties,” on page 67
- “Required Component Properties,” on page 67
- “Best Practices for Developing Components,” on page 68

**Defining Component Actions**

Each component includes the predefined life cycle stages or actions to install, configure, start, and update scripts for a service or application component.

The catalog administrator must provide a bash, Windows CMD, or PowerShell script for at least one of the INSTALL, CONFIGURE, or START life cycle stages. These scripts are customized to use the component properties.

For example, to deploy an Apache Tomcat server in a virtual machine, you might add the following scripts:

**INSTALL**

Download the Tomcat server installation bits and install the Tomcat service.

**CONFIGURE**

Set the JAVA_OPTS, CATALINA_OPTS, and any other required configuration.
START
Start the Tomcat service using the start command in the Tomcat server.

UPDATE
Modify the configuration of Tomcat service using the update script or change the cluster size to scale a deployed application and manage the clustered nodes using a load balancer.

The application architect can parameterize the script by declaring, for example, the installer location, installation path, and Tomcat start command as properties in the script. The parameters render the scripts generic. You can deploy the service on different environments without modifying these generic scripts.

You can also modify parameters from the action script. These modified properties can be referred to as property values for other components. See VMware vFabric Application Director Catalog Services.

The scripts defined for an action are run in the /tmp/ directory. The Linux script is located at /tmp/runId/ComponentName-LifecycleStageName. The runId is the unique job identifier for each deployment, which is available on the status window of the deployment summary page. The Windows script is located at/Users/darwin/AppData/Local/Temp.

Verify that no processes are prompting for user interaction when the action script is running. Interruptions pause the script, causing it to remain in an idle state indefinitely, and eventually fail. In addition, if a Windows CMD script exits with a non-zero exit status, the vFabric Application Director agent stops the entire deployment and marks it as Failed Deployment. Use exit /b 0 to indicate success status and exit /b non-zero for error status. See “Supported Action and Custom Task Scripts,” on page 60.

You can add a service or custom task in an application deployment and define the component action. See “Add a Service to the Catalog,” on page 73 and “Add a Custom Task to the Catalog,” on page 77.

Supported Action and Custom Task Scripts
vFabric Application Director supports authoring in bash script for a Linux-based application and authoring in Windows CMD or PowerShell scripts for a Windows-based application.

To select the appropriate action script type for a life cycle stage, in the blueprint canvas, double-click the Script Type column and select the script from the drop-down menu. For a custom task script, select one or more operating systems. You can set supported scripts from the drop-down menu.

Based on the script type you select, you can author code and access the relevant properties through the variables in the script. For an action script of a life cycle stage, you can also use different script types in the same operating system family for each life cycle in the same component. For example, you can use the Windows CMD script for the INSTALL stage and a PowerShell script for the CONFIGURE stage. See “Types of Properties,” on page 61.

When authoring an action script, the exit and return codes vary between script types. The application architect should set proper exit codes in the script that are applicable to the application deployment. If the script does not have exit and return codes, the last command that ran in the script becomes the exit status.

Table 8-1. Action Script Exit and Return Codes

<table>
<thead>
<tr>
<th>Script Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bash</td>
<td>You can use return 0 or exit 0 codes in action scripts to indicate success status. To indicate error status, you can use return non-zero or exit non-zero.</td>
</tr>
<tr>
<td>Windows CMD</td>
<td>Do not use exit 0 and exit non-zero codes in the action script. If you use these codes in the script, the computed properties task processing is stopped prematurely. Use exit /b 0 to indicate success status and exit /b non-zero for error status.</td>
</tr>
<tr>
<td>Windows PowerShell</td>
<td>You can use exit 0 to indicate success status and exit non-zero for error status.</td>
</tr>
</tbody>
</table>

See “Understanding the Deployment and Update Process,” on page 103.

When you use Windows PowerShell to author a script, the warning, verbose, debug, and host calls are not supported in an action script.
Configuring Component Properties

Component properties are used to parameterize scripts so that vFabric Application Director can pass the defined properties as environment variables to scripts running in a virtual machine.

Before running a script from the life cycle stage, the vFabric Application Director agent in the virtual machine communicates with the vFabric Application Director server to resolve the properties. The agent then proceeds to create script-specific variables from these properties and passes them to the scripts.

- **Types of Properties** on page 61
  vFabric Application Director supports string, array, content, and computed properties.

- **Defining Property Values** on page 63
  A catalog administrator can define properties in the catalog for services and custom tasks.

- **Binding to Other Properties** on page 64
  In several deployment scenarios, a component needs the property value of another component to customize itself. In vFabric Application Director, this process is called binding to other properties.

- **Auto-Binding to Other Properties** on page 65
  You can Auto-bind to other properties by using the Auto-Bind meta data setup in a service and default meta data in nodes, node arrays, services, and the WAR application component.

- **Passing Property Values Between Life Cycle Stages and Components** on page 66
  Properties defined for each component can be used in an action script for each of the life cycle stages.

### Types of Properties

vFabric Application Director supports string, array, content, and computed properties.

**Note** The names of properties are case-sensitive and can contain only alphabetic, numeric, hyphen (-), or underscore (_) characters.

#### String Property

The property value can be a string or the value bound to another string property. A string value can contain any ASCII characters. For a bound property, use the `Properties` tab in the blueprint canvas to select the appropriate property for binding. The property value is then passed to the action scripts as raw string data.

<table>
<thead>
<tr>
<th>Sample String Property</th>
<th>Script Syntax</th>
<th>Sample Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>admin_email = &quot;admin@email987.com&quot;</code></td>
<td>Bash - <code>admin_email</code></td>
<td><code>echo $admin_email</code></td>
</tr>
<tr>
<td></td>
<td>Windows CMD - <code>%admin_email%</code></td>
<td><code>echo %admin_email%</code></td>
</tr>
<tr>
<td></td>
<td>Windows PowerShell - <code>admin_email</code></td>
<td><code>write-output $admin_email</code></td>
</tr>
</tbody>
</table>

#### Array Property

The property value can be an array of strings defined as `["value1", "value2", "value3"...]` or the value bound to another array property. For an array of strings, the value in the array elements can contain any ASCII characters. To properly encode a backslash character in an Array property value, add an extra backslash, for example, `"c:\test\test2"`. For a bound property, use the `Properties` tab in the blueprint canvas to select the appropriate property for binding.

For example, consider a load balancer virtual machine that is balancing the load for a cluster of application server virtual machines. In such a case, an array property is defined for the load balancer service and set to the array of IP addresses of the application server virtual machines.
These load balancer service configure scripts use the array property to configure the appropriate load balancing scheme on the Red Hat, Windows, and Ubuntu operating systems.

<table>
<thead>
<tr>
<th>Sample Array Property</th>
<th>Script Syntax</th>
<th>Sample Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>operating_systems = [&quot;Red Hat&quot;, &quot;Windows&quot;, &quot;Ubuntu&quot;]</td>
<td>for (( i = 0 ; i &lt; ${operating_systems[@]} ; i++ )); do echo ${operating_systems[$i]} done</td>
<td>for the entire array of strings</td>
</tr>
<tr>
<td></td>
<td>for ${operating_systems[N]}</td>
<td>for the individual array element</td>
</tr>
<tr>
<td></td>
<td>for /F &quot;delims== tokens=2&quot; %%A in ('set operating_systems_') do ( echo %%A )</td>
<td>for /F &quot;delims==&quot; tokens=2 %%A in (&quot;set operating_systems_&quot;) do ( echo %%A )</td>
</tr>
<tr>
<td></td>
<td>foreach ($os in $operating_systems) { write-output $os }</td>
<td>foreach ($os in $operating_systems) { write-output $os }</td>
</tr>
</tbody>
</table>

Content Property

The property value is a URL to a file to download content. vFabric Application Director agent downloads the content from the URL to the virtual machine and passes the location of the local file in the virtual machine to the script.

Content properties must be defined as a valid URL with the HTTP or HTTPS protocol. For example, the sample Hyperic HQ agent has a property value HQ_PACKAGE_32 set to http://DarwinServerIP/artifacts/services/hyperic/hyperic-hq-agent-4.6-x86-linux.tar.gz. The Hyperic artifacts are hosted in the vFabric Application Director appliance and the URL points to that location in the appliance. The vFabric Application Director agent downloads the artifacts from the specified location into the deployed virtual machine.

<table>
<thead>
<tr>
<th>Sample String Property</th>
<th>Script Syntax</th>
<th>Sample Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>HQ_PACKAGE = &quot;<a href="http://DarwinServerIP/hyperic/hyperic-hq-agent-linux.tar.gz">http://DarwinServerIP/hyperic/hyperic-hq-agent-linux.tar.gz</a>&quot;</td>
<td>Bash - $HQ_PACKAGE</td>
<td>tar -zxvf $HQ_PACKAGE</td>
</tr>
<tr>
<td></td>
<td>Windows CMD - %HQ_PACKAGE%</td>
<td>start /wait c:\unzip.exe %HQ_PACKAGE%</td>
</tr>
<tr>
<td></td>
<td>Windows PowerShell - &amp; c:\unzip.exe $HQ_PACKAGE</td>
<td>&amp; c:\unzip.exe $HQ_PACKAGE</td>
</tr>
</tbody>
</table>

Computed Property

This property type does not allow an initial value when being defined. Instead, computed property type takes the value from the INSTALL, CONFIGURE, or START life cycle scripts. The assigned value is propagated to the subsequent available life cycle and dependent components.

**Note**  Any computed properties defined in the action scripts are not available to the UPDATE life cycle script. You must set the value for the computed property in the UPDATE life cycle script.

<table>
<thead>
<tr>
<th>Sample String Property</th>
<th>Script Syntax</th>
<th>Sample Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>my_unique_id = &quot;&quot;</td>
<td>Bash - $my_unique_id</td>
<td>export my_unique_id=&quot;0123456789&quot;</td>
</tr>
<tr>
<td></td>
<td>Windows CMD - %my_unique_id%</td>
<td>set my_unique_id=0123456789</td>
</tr>
<tr>
<td></td>
<td>Windows PowerShell - $my_unique_id</td>
<td>$my_unique_id = &quot;0123456789&quot;</td>
</tr>
</tbody>
</table>
**Property Type Reference**

Property type references show which property types you can refer to for configuration purposes.

<table>
<thead>
<tr>
<th>Property Type</th>
<th>Property Type to Bind</th>
<th>Auto-Bind Expose</th>
<th>Auto-Bind Consume</th>
<th>Accept Initial Property Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>String, Array, Content, and Computed</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Content</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Yes</td>
</tr>
<tr>
<td>Array</td>
<td>String, Content, and Computed</td>
<td>N/A</td>
<td>Yes</td>
<td>N/A</td>
</tr>
<tr>
<td>Computed</td>
<td>N/A</td>
<td>Yes</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Note**  
Array properties can also bind to a predefined node array property all in a cluster. The all property, in a regular expression, is a method to collect all of the values for a given property in a clustered node.

For more information about binding properties, see “Binding to Other Properties,” on page 64. To understand the Auto-Bind expose and Auto-Bind consume concepts, see “Auto-Binding to Other Properties,” on page 65.

Example of a string property value when binding to different types of properties.

<table>
<thead>
<tr>
<th>Sample Property Type</th>
<th>Property Type to Bind</th>
<th>Binding Outcome (A binds to B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>String (property A)</td>
<td>String (property B=&quot;Hi&quot;)</td>
<td>A=&quot;Hi&quot;</td>
</tr>
<tr>
<td>String (property A)</td>
<td>Content (property B=&quot;<a href="http://my.com/content">http://my.com/content</a>&quot;)</td>
<td>A=&quot;<a href="http://my.com/content">http://my.com/content</a>&quot;</td>
</tr>
<tr>
<td>String (property A)</td>
<td>Array (property B=[&quot;1&quot;,&quot;2&quot;])</td>
<td>A=[&quot;1&quot;,&quot;2&quot;]</td>
</tr>
<tr>
<td>String (property A)</td>
<td>Computed (property B=&quot;Hello&quot;)</td>
<td>A=&quot;Hello&quot;</td>
</tr>
</tbody>
</table>

Example of an array property value when binding to different types of properties.

<table>
<thead>
<tr>
<th>Sample Property Type</th>
<th>Property Type to Bind</th>
<th>Binding Outcome (A binds to B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Array (property A)</td>
<td>String (property B=&quot;Hi&quot;)</td>
<td>A=&quot;Hi&quot;</td>
</tr>
<tr>
<td>Array (property A)</td>
<td>Content (property B=&quot;<a href="http://my.com/content">http://my.com/content</a>&quot;)</td>
<td>A=&quot;<a href="http://my.com/content">http://my.com/content</a>&quot;</td>
</tr>
<tr>
<td>Array (property A)</td>
<td>Computed (property B=&quot;Hello&quot;)</td>
<td>A=&quot;Hello&quot;</td>
</tr>
</tbody>
</table>

**Defining Property Values**

A catalog administrator can define properties in the catalog for services and custom tasks.

To allow for customization of a component script, the default value for a property in the catalog can be overridden in a blueprint to accommodate the needs of a specific application in which the service is used. The property value can be further overridden in the deployment profile to comply with the deployment environment where the service is deployed.

**Note** Property values are case-sensitive. A new property value does not take effect if the value is typed incorrectly.

For example, to deploy an Apache Tomcat server, the catalog administrator might have configured the value of the JVM HEAP_SIZE to 512MB in the service. The application architect can redefine the value to 1024MB for a larger application. The deployer might override the value to 2048MB when deploying the application in a production deployment environment to handle large loads.
Property values are non-overridable by default, but the catalog administrator can choose to enable the overridable feature. For properties that can be overridden, the application architect might force a value for the property in some regulated environments, and disable the overridable feature for the deployer.

You can define property values when you add a service or custom task in an application deployment. See “Add a Service to the Catalog,” on page 73 and “Add a Custom Task to the Catalog,” on page 77.

Binding to Other Properties

In several deployment scenarios, a component needs the property value of another component to customize itself. In vFabric Application Director, this process is called binding to other properties.

The catalog administrator can modify property definitions in the script. For example, a WAR component might need the installation location of the Apache Tomcat server. The WAR component can set the server_home property value to the Apache Tomcat server install_path property value.

The bash script for a component can use only its own properties. In addition to setting a property to a hard-coded value, vFabric Application Director allows a property to be bound to another property in the blueprint. When you bind to another property, you can customize a script based on the value of another component’s properties and virtual machine properties such as IP addresses. To bind a property to another property, select the property value from the Blueprint Value drop-down menu in the Edit Property dialog box.

For a single virtual machine node, the referenced properties in the Blueprint Value drop-down menu are, NodeName:ComponentName:PropertyName.

For clustered virtual machines node, the referenced properties in the Blueprint Value drop-down menu are, all(NodeName:ComponentName:PropertyName). When another property refers to this cluster property, it gets the PropertyName property values from all of the virtual machines in the cluster. The predefined all(NodeName:node_array_index) property for clustered virtual machines gets the collection of node array indexes in the cluster. See “Predefined Node Array Index Property,” on page 67.

For cluster and single nodes, the self:ComponentName:PropertyName value is used to indicate a component property from the virtual machine where the target component is running. For instance, if a WAR component is deployed in an Apache Tomcat server, the WAR server_home property can be set to self:tomcat:install_path to refer to the Apache Tomcat server running in the current virtual machine.

Node level properties such as IP address display as, NodeName:ip or self:ip. These properties belong exclusively to the virtual machine and not to any specific component in the virtual machine.

For IP addresses of virtual machines with multiple NICs, vFabric Application Director provides either a NodeName:NICx_ip or self:NICx_ip property. Where x reflects the NIC number. See “Predefined IP Address Property,” on page 66.

Figure 8-1. Binding to Another Property in a Clustered Node
Auto-Binding to Other Properties

You can Auto-bind to other properties by using the Auto-Bind meta data setup in a service and default meta data in nodes, node arrays, services, and the WAR application component.

Based on the Auto-Bind meta data defined in a service by the catalog administrator, vFabric Application Director displays the Auto-Bind type as consume or expose to allow other properties to bind to it in the blueprint. In the blueprint canvas, an Expose icon appears next to the Auto-Bind Expose properties and a Consume icon appears next to the Auto-Bind Consume properties. The tooltips for the icons display the Auto-Bind type and Auto-Bind tag information. For example, the Consume icon next to the webapps_dir property shows the Auto-Bind Consume: Servlet Container, DIR information in the tooltip. The catalog administrator sets the service Auto-Bind type and tag parameters when creating a service or editing an existing service. See “Add a Service to the Catalog,” on page 73.

Preconfigured Components for Automatic Binding

WAR is the only application component that is preconfigured for auto-binding. You cannot edit the predefined Auto-Bind type and tag parameters.

Adding an application component onto a service implies that properties can be set on either. In this case, auto-binding happens twice. The source properties on the application component are set from target properties on the service and the parent node or node array. Then, the source properties on the service are set from target properties on the application component.

Table 8-2. Preconfigured WAR Application Component

<table>
<thead>
<tr>
<th>Application Component</th>
<th>Property</th>
<th>Auto-Bind Type</th>
<th>Auto-Bind Tag</th>
<th>Script</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux WAR file</td>
<td>Context</td>
<td>Expose</td>
<td>Context</td>
<td>catalog_oob_data.xml</td>
</tr>
<tr>
<td>Linux WAR file</td>
<td>service_start</td>
<td>Consume</td>
<td>Servlet Container, Start</td>
<td>catalog_oob_data.xml</td>
</tr>
<tr>
<td>Linux WAR file</td>
<td>service_stop</td>
<td>Consume</td>
<td>Servlet Container, Stop</td>
<td>catalog_oob_data.xml</td>
</tr>
<tr>
<td>Linux WAR file</td>
<td>webapps_dir</td>
<td>Consume</td>
<td>Servlet Container, DIR</td>
<td>catalog_oob_data.xml</td>
</tr>
</tbody>
</table>

Some nodes and node arrays have predefined Auto-Bind type and tag parameters for properties. You cannot edit these parameters. When a service or application component is added to a node or node Array, the source properties are taken from the service or application component and the target properties are taken from the node or node array.

Table 8-3. Predefined Node and Node Array Properties

<table>
<thead>
<tr>
<th>Node Type</th>
<th>Property</th>
<th>Auto-Bind Type</th>
<th>Auto-Bind Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node, Node Array</td>
<td>hostname</td>
<td>Expose</td>
<td>Hostname</td>
</tr>
<tr>
<td>Node, Node Array</td>
<td>ip</td>
<td>Expose</td>
<td>IP Address</td>
</tr>
<tr>
<td>Node Array</td>
<td>node-array-index</td>
<td>Expose</td>
<td>Node Array Index</td>
</tr>
</tbody>
</table>

When dependencies exist, the source properties are taken from the component that the link originates. Target properties are taken from the component from which the link is drawn, and from parent service or node components.

Property Compatibility

Auto-binding depends on the compatibility between Source and Target property types and Node type.
Table 8-4. Property Type and Node Type Compatibility

<table>
<thead>
<tr>
<th>Source Property Type</th>
<th>Target Property Type</th>
<th>Node Type</th>
<th>Compatible</th>
</tr>
</thead>
<tbody>
<tr>
<td>Array</td>
<td>String</td>
<td>Node</td>
<td>Yes</td>
</tr>
<tr>
<td>Array</td>
<td>Computed</td>
<td>Node</td>
<td>Yes</td>
</tr>
<tr>
<td>Array</td>
<td>String</td>
<td>Node Array</td>
<td>Yes</td>
</tr>
<tr>
<td>Array</td>
<td>Computed</td>
<td>Node Array</td>
<td>Yes</td>
</tr>
<tr>
<td>String</td>
<td>String</td>
<td>Node</td>
<td>Yes</td>
</tr>
<tr>
<td>String</td>
<td>Computed</td>
<td>Node</td>
<td>Yes</td>
</tr>
<tr>
<td>String</td>
<td>String</td>
<td>Node Array</td>
<td>No</td>
</tr>
<tr>
<td>String</td>
<td>Computed</td>
<td>Node Array</td>
<td>No</td>
</tr>
</tbody>
</table>

Passing Property Values Between Life Cycle Stages and Components

Properties defined for each component can be used in an action script for each of the life cycle stages.

For a computed property, you can modify the value of a property and pass the value to the next life cycle stage of the action script. For example, if component A has the progress_status value defined as staged, in the INSTALL and CONFIGURE life cycle stage you change the value to progress_status=installed in the respective action scripts. If component B is bound to component A, the property values of progress_status in the life cycle stages of the action script are the same as component A.

Define in the blueprint that component B depends on A. This dependency defines the passing of correct property values between components whether they are in the same node or across different nodes.

For example, you can update a property value in an action script by using the supported scripts.

- Bash `progress_status="completed"`
- Windows CMD `set progress_status=completed`
- Windows PowerShell `$progress_status="completed"`

**Note** Array and content property do not support passing modified property values between action scripts of life cycle stages and components.

**vFabric Application Director Predefined Properties**

vFabric Application Director provides some commonly used properties as predefined properties. These properties are available for clustered nodes.

Usually, most services need the IP address of the virtual machines they are running in. For clustered nodes, a service might need the IP addresses of all the virtual machines in the cluster. Therefore, the virtual machine IP address is designated as a predefined property.

**Predefined IP Address Property**

A node can have multiple NICs, with each NIC assigned one IP address in the deployed virtual machine.

These IP addresses are available in the `NodeName:NICx_ip` properties, where `-x-` is the NIC number.
In the sample Clustered Duke’s Bank application, the Load Balancer node properties are shown as load_balancer:NIC0_ip and load_balancer:NIC1_ip because the Load Balancer node has two NICs defined. It is not guaranteed that NIC0 and NIC1 will be assigned to eth0 and eth1, respectively, in the virtual machine. The NICs are logical names in the blueprint for the network interfaces. These NICs are mapped to logical networks, which are mapped to specific cloud networks. The property NIC0_ip returns the IP address assigned to the virtual machine as defined in the blueprint, not the eth0 IP address in the virtual machine.

The load_balancer:ip property is also provided. This property refers to the IP address of the first NIC and NIC0. If a component needs IP addresses of all virtual machines in a cluster, it uses the all(NodeName:NICx_ip) property. To refer to the IP address of the current virtual machine, self:ip property is available. This property is useful for clustered nodes, as a component might need to know the virtual machine it is in, rather than the IP addresses of all the virtual machine in the cluster. To get the properties for a specific virtual machine in a cluster, you can use the all(NodeName:node_array_index) property. See “Predefined Node Array Index Property,” on page 67.

Predefined Node Array Index Property

For clustered nodes, a special property called node_array_index identifies the position of the current virtual machine in the cluster.

The node_array_index property value of the first virtual machine in the cluster is 0, the node_array_index property value of the second virtual machine in the cluster is 1, and so on.

For example, in a deployment with a clustered node, if the first virtual machine is the master virtual machine, then it manages other virtual machines in the cluster as subordinate machines. This master virtual machine must be configured differently. For example, if one of the properties for the component is myPosition, then it must be bound to the predefined self:node_array_index property. The component script can verify whether the value of myPosition property is 0, and if so it can perform an additional configuration.

For clustered nodes, the predefined node array property is all(NodeName:node_array_index). The property gives the indexes of all the virtual machines in the cluster and can be used to identify the size of the cluster.

Secured Component Properties

Properties are used to configure deployed components. In some cases, they are used to store sensitive data, such as passwords.

For example, a WAR component might need to store sensitive data such as passwords to access the database. These properties can be marked as secured. Values of secured properties are masked and shown as asterisks in vFabric Application Director.

If a property is changed from secured to unsecured, for security purposes, vFabric Application Director resets the property value, for security purposes. You must set a new value for the property.

**IMPORTANT** If secured properties are printed in the script using the echo command or other similar commands, these values appear in plain text in the log files. The values in the log files are not masked.

Required Component Properties

You can set properties as they are required. For example, a deployment might fail if properties are not defined for scripts that rely on them during the deployment.

For example, to run an Apache Tomcat server, Java is required and the JAVA_HOME property value must be set.
When a property is marked as required, a value must be provided in at least one of the life cycle stages of the property, such as the catalog, blueprint, or deployment profile. For example, a catalog administrator can define a required property, mark it as overridable, and not set any value for the property in the catalog. The application architect must provide a value for this property in the blueprint or mark it as overridable in the deployment profile. If the application architect has not set a value for this property, the deployer is required to set a value for this property in the deployment profile before deploying the application.

Best Practices for Developing Components

Familiarize yourself with the sample components in the catalog, as they include a number of examples of how to define properties and action scripts.

**NOTE** Use the sample components in the vFabric Application Director catalog only in a test environment.

Follow these best practices when developing components in vFabric Application Director.

- Some installers might need access to the tty console. Redirect the input from `/dev/console`.

  For example, the predefined RabbitMQ service uses the `./rabbitmq_rhel.py --setup-rabbitmq < /dev/console` command in its install script.

- Content property with a defined URL downloads the content without using a proxy. If your deployment environment requires a proxy, define the property as a string and use `wget` in the script to download the content.

- When a component uses multiple life cycle stages, the property value can be changed in the INSTALL life cycle stage. The new value is sent to the next life cycle stage. Action scripts can compute the value of a property during deployment to supply the value to other dependent scripts.

  **NOTE** You cannot change the content property value for a component that uses multiple life cycle stages.

For example, in the Clustered Duke’s Bank sample application, JBossAppServer service computes the JVM_ROUTE property during the install life cycle stage. This property is used by the JBossAppServer service to configure the life cycle. Apache load balancer service then binds its JVM_ROUTE property to the `all(appserver:JbossAppServer:JVM_ROUTE)` property to get the final computed value of node0 and node1.

If a component requires a property value from another component to complete an application deployment successfully, you must state explicit dependencies in the application blueprint.

- For a script to run without any interruptions, the return value must be set to zero (0).

  This setting allows the agent to capture all of the properties and send them to the vFabric Application Director server.
Managing the vFabric Application Director Catalog

With the prepopulated, extensible catalog of standard logical templates, sample services, task scripts, operating systems, and tags, application architects can quickly create a blueprint of a multiter enterprise application.

**IMPORTANT** vFabric Application Director includes many predefined logical templates and services. A catalog administrator can use or modify these predefined templates and services to avoid having to create them themselves. A best practice is to create a copy of the service to preserve the original for future reference.

The services, tasks, operating systems, and tags in the catalog are available to all user groups in vFabric Application Director. You must add logical templates for each group outside the Default group.

For example, you can use the vFabric Application Director catalog to create custom tasks, which are customized scripts that you can add to the execution plan in a deployment profile.

Familiarize yourself with the key concepts relating to managing the catalog. See “Key Concepts,” on page 12.

This chapter includes the following topics:

- “Add a Logical Template to the Catalog,” on page 69
- “Add a Service to the Catalog,” on page 73
- “Add a Custom Task to the Catalog,” on page 77
- “Add Operating Systems to the Catalog,” on page 80
- “Add Tags to the Catalog,” on page 81

**Add a Logical Template to the Catalog**

You can add compatible services and scripts to custom or sample logical templates when you model an application blueprint and map them to actual cloud templates from vCloud Director or Amazon EC2. Logical templates allow an application blueprint to remain cloud agnostic.

As part of the logical template definition, you can describe which services are already installed in the template along with the operating system. Typically, in IT organizations, a few performance monitoring agents or virus scanners are installed in a template. Also, for example, vFabric tc Server might be preinstalled in the logical template to speed up deployments. If you always use a particular service when you deploy a logical template, you can preinstall it to avoid having to add it for every deployment.

**NOTE** Save your changes frequently. The vFabric Application Director user interface session expires after 30 minutes of inactivity. If the session expires, any changes that were not saved are lost.

**Prerequisites**

- Verify that your user account has the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.
Create cloud templates in vCloud Director or Amazon EC2 that meet the requirements for working properly in vFabric Application Director. See “Virtual Machine Requirements for Creating Custom Templates,” on page 39.

Verify that at least one cloud provider is registered in vFabric Application Director. See “Registering a Cloud Provider and Template,” on page 52.

If the application requires access to URLs from outside the corporate firewall, configure the applicable services and application components to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.

Verify whether you must add items to the operating systems and descriptive tag lists. See “Add Operating Systems to the Catalog,” on page 80 and “Add Tags to the Catalog,” on page 81.

If you plan to have application or services preinstalled in a template, create and configure the services before you can add them to the template. See “Add a Service to the Catalog,” on page 73.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Catalog > Logical Templates**.

2. Click **New** in the toolbar.

3. Set the name of the logical template, add a description, and click **Save**.

   To keep track of which cloud template or operating system you are using, include the name of the cloud template or operating system.

4. Click **Create LT Version** to create a logical template version.

   You can create multiple versions for a logical template.

   A page opens for creating a logical template version.

5. Complete the template version information.

    | Option       | Description |
    |--------------|-------------|
    | Name         | Logical template version name stays the same. |
    | Version      | Add a qualifier to denote major and minor versions such as 1.2.1 or 1.0.1-CentOS56-32bit. |
    | Description  | Add detailed information about the logical template. For example, describe a script you are using or the amount of disk size required. |
    | Tags         | Categorize logical templates based on the functions that they provide. vFabric Application Director organizes the templates based on tags you see when you create a deployment blueprint for an application. **Note** Only tags designated as Server Type appear in the drop-down menu. You can add multiple tags. To use a tag that is not in the list, click **Cancel**, and select **Catalog > Tags** to create a tag. |
    | Supported OS | Specifies the operating system installed in the logical template. This OS information is used in the application blueprint to limit which services you can add to this template. Some services do not run on all operating systems. For example, if you specify an Ubuntu operating system, when you use this template in a blueprint and try to add a service that is not compatible with Ubuntu, vFabric Application Director prevents you from adding that service. To use an operating system name that is not in the list, click **Cancel**, and select **Catalog > Operating Systems** to create an operating system name. |
6 Map a cloud template to the logical template.

You can add multiple cloud templates to one logical template version or select different cloud templates for different clouds. Even if you are using the same cloud provider, you might need to select from different cloud templates at deployment time to allow for different template configurations.

Duplicate cloud template and logical template mappings and empty rows are not saved.

For example, with multiple cloud templates, you can use the same logical template. If you are deploying to a production environment, you can select a cloud template that has a large amount of disk space. For a test or staging environment, you can select a cloud template with a small amount of disk space.

a In the Cloud Provider Name column, click the down arrow to select a cloud provider.

A list of cloud providers that you created appears.

b In the Cloud Template column, click the down arrow to designate a cloud template.

Cloud templates that belong to the same group as the user appear in the drop-down menu. If the list of cloud templates is empty, the existing cloud templates do not belong to your group or a cloud template was not registered.

7 (Optional) To map multiple cloud templates to a logical template, repeat Step 6.

8 (Optional) Specify a preinstalled service.

a In the Service Name column, click the down arrow to select a preinstalled service.

b To use a service that is not in the list, click Cancel, and select Catalog > Services to create a service and the action scripts it includes.

If any preinstalled services are added to a logical template after creating a blueprint, the new preinstalled services are not added to the node.

c (Optional) To add the new preinstalled services to the node, drag the logical template with the preinstalled service to the blueprint, transfer the services and components to the new template, recreate any applicable dependencies, and delete the old template.

9 When you finish creating the template, click Save.

The logical template that you created is added to the Logical Templates page. The template also appears in the list of logical templates that you can include when you create a deployment blueprint for an application.

What to do next

Copy an existing logical template version, edit, or delete the template. See “Maintain Template Versions,” on page 71.

Maintain Template Versions

To map an existing logical template to another cloud template or add preinstalled services, you can edit a logical template version.

Prerequisites

- Verify that your user account has the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.
- Create cloud templates in vCloud Director or Amazon EC2 that meet the requirements for working properly in vFabric Application Director. See “Virtual Machine Requirements for Creating Custom Templates,” on page 39.
- If the application requires access to URLs from outside the corporate firewall, configure the applicable services and application components to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
Verify whether you must add items to the operating systems and descriptive tag lists. See “Add Operating Systems to the Catalog,” on page 80 and “Add Tags to the Catalog,” on page 81.

If you plan to have application or services preinstalled in a template, you must create and configure the services before you can add them to the template. See “Add a Service to the Catalog,” on page 73.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Catalog > Logical Templates.

2. Copy a logical template from an existing logical template version.
   - Open the logical template version and click Copy in the toolbar.
     All of the cloud templates, services, operating systems, and a list of descriptive tags to identify application components are retained. You cannot change the name of the existing logical template version.
   - Change the version number, update the description, and if needed update the cloud provider, cloud template, or services.
     The version number and description differentiates the new logical template from the parent version.
     A best practice is to create a copy of the template to preserve the original for future reference.
   - Click Save.

3. To change existing cloud provider, cloud template, or services for a logical template, click Edit in the toolbar, make your changes, and click Save.

What to do next

If you have an older version of a logical template, you can delete it from the catalog. See “Delete a Logical Template from the Catalog,” on page 72.

Delete a Logical Template from the Catalog

You can delete a logical template from the Logical Templates page.

Prerequisites

- Verify that your user account has the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.
- Verify that the logical templates in application blueprints and all of the other objects referencing the template are deleted.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Catalog > Logical Templates.

2. Open the logical template.

3. Remove the existing logical template versions and confirm your selection.

4. Click the arrow next to the logical template name to return to the Logical Templates page.

5. Delete the logical template and confirm your selection.
Add a Service to the Catalog

Use the vFabric Application Director catalog to create software services. A service comprises scripts for installing, configuring, starting, and updating the software services that your application requires.

The vFabric Application Director catalog contains predefined services. Sometimes, rather than creating a new service, you might prefer to edit the scripts and variables in the predefined service. As a best practice, create a copy of the predefined service before you make changes. See “Maintain Service Versions,” on page 76.

**Note** Save your changes frequently. The vFabric Application Director user interface session expires after 30 minutes of inactivity. If the session expires, any changes that were not saved are lost.

**Prerequisites**

- Verify that your user account has the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.
- Verify whether you must add items to the operating systems and descriptive tag lists. See “Add Operating Systems to the Catalog,” on page 80 and “Add Tags to the Catalog,” on page 81.
- If you plan to use a script that downloads software from an external Web site, verify that the virtual machine you use for deploying the application has access to an external network.
- If you plan to add an operating system with SELinux enabled, verify that the permissive mode is enabled or specific exceptions are in place for the applications being installed to avoid any installation failures.
- Familiarize yourself with the basic concepts of defining and configuring component properties and actions. See Chapter 8, “Developing vFabric Application Director Components,” on page 59.
- If a service is used in a blueprint or included as a preinstalled service in a logical template, the service cannot be deleted. Before you delete a service, delete the referenced blueprint or logical template with the preinstalled service.
- To view examples about how to name or describe a service, select Catalog > Services to view the predefined services.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select Catalog > Services.
2. Click New in the toolbar.
3. Set the name of the service, add a description, and click Save.
   
   Use a name that corresponds to the software component to which the action scripts relate and append qualifiers for tracking purposes. For example, if you use the service on a particular operating system, you can set the service name as **ServiceName_Windows** or **ServiceName_Linux**.
4. Click Create Service Version to create a service version.
   
   You can create multiple versions for a service.
   
   A page for creating a service version opens.
5. Complete the service version information.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Service version name stays the same.</td>
</tr>
<tr>
<td>Version</td>
<td>Specify major, minor, or micro releases, with or without qualifiers. For example, you might use version numbers such as 1.0, 1.5, or 1.0.1-Linux.</td>
</tr>
</tbody>
</table>
### Description
If you are creating a service version for a specific operating system, describe the operating system version used and any applicable required configuration.

### Tags
You use tags to organize the list of services you see when you create a deployment blueprint for an application.

**Note**: Only tags designated as Server Type appear in the drop-down menu. You can add multiple tags.

To create a tag that is not in the list, click **Cancel** and select **Catalog > Tags**.

### Supported OSes
If the scripts used in this service can run only on particular operating systems, select those operating systems here. In the blueprint editor, vFabric Application Director prevents the service from being added to a template unless the template contains one of these operating systems. Leave this field blank if the service can be used in any operating system.

To use an operating system name that is not in the list, click **Cancel**, and select **Catalog > Operating Systems** to create an operating system name.

### Supported Components
If only certain types of application component can run in this service, specify those components here.

For example, only WAR and JAR components can run in a vFabric tc Server instance. Only SQL scripts can run in a database server. The components that you select restrict what application components can be added to this service in an application blueprint. Leave this field blank if you can add components to the service.

### Pre-installed in a Template
Indicates that the service is already installed in a template. Only those services that have this check box selected appear in the Services Included section while you create or edit a logical template. Preinstalled services appear in the OS Templates section of the application blueprint and are available as part of the template.

6 In the Properties section, define the variables used in the scripts for the service.

vFabric Application Director passes defined properties as environment variables to scripts running in a virtual machine.

### Property Name and Description
Define a variable and add an optional description.

The property name cannot begin with a digit.

For example, you might create a property for a port number or a URL that is used in the script.

### Type
To change the type, select a property type from the drop-down menu. You cannot add types to the menu.

If the property type is an Array, you cannot set the Auto-Bind type to Expose.

If the property type is Computed, you cannot set the Auto-Bind type to Consume.

If the property type is Content, you cannot set the Auto-Bind type to Expose or Consume.

### Value
Type the value to substitute for this property when the script runs. For example, for a property called `http_port`, you might type `80` in this field.

**Note**: If you add a value to the `http_proxy`, `https_proxy`, or `ftp_proxy` property with service scripts that use the `darwin_globol.conf` file as a file source, when the script runs, these properties override any existing proxy information in the deployed application.

If the **Required** check box is selected and the **Overridable in Blueprint** check box is not selected, the property must have a value.

If the property type is Computed, you cannot type a value.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Auto-Bind Type**         | Describes the role that a property is assigned to for auto-binding in the blueprint canvas.  
When you select the Auto-Bind type None, the property does not Auto-Bind.  
When you select Consume, the property value is set with auto-binding.  
When you select Expose, the property can be used to customize another property value with auto-binding.  
**Note**: A property cannot have the Consume and Expose roles.  
An Auto-Bind type of Consume or Expose requires at least one Auto-Bind tag to be defined for the property. Tags on the Consume property must be a subset of the tags on the Expose property. The tag sets do not have to be equal.  
The property type you define affects the Auto-Bind type you can set. |
| **Auto-Bind Tags**         | Adds the Property Descriptor tags to a property.  
**Note**: Tags designated only as Property Descriptor appear in the drop-down menu.  
You can add multiple Auto-Bind tags.  
For example, if the Auto-Bind type for the Apache 2.2.0 service `appsrv_routes` property is assigned as Consume, and the Auto-Bind type for the vFabric tc Server 2.1.0 service `JVM_ROUTE` property is assigned as Expose. The `appsrv_routes` property uses the `JVM_ROUTE` property values to customize itself. To allow Auto-Bind to connect to these service properties, you can add the Servlet Container and Route tags to the `appsrv_routes` and `JVM_ROUTE` properties, respectively. When you create an application that includes the vFabric tc Server and Apache services, the properties Auto-Bind to each other in the blueprint editor. |
| **Required**               | Select the check box for properties that are required to deploy an application.  
For example, for an `http_proxy_port` property using the Apache JServ Protocol (AJP), you must set the value to 8009.  
If any of the required property values are left blank, you are prompted to complete them before an application deployment.  
If the property type is Computed, the **Required** check box is not applicable.  
If the property type is Content, the **Required** check box is selected by default and the Secured check box is not applicable. |
| **Secured**                | Select the check box for passwords you define or to obscure the values of other properties. For example, the `db_password` property in the sample MySQL service is secured.  
If a property is changed from Secured to Unsecured, vFabric Application Director resets the property value, for security purposes. You must set a new password value for the property. |
| **Overridable in Blueprint** | Allows users such as an application architect to override the value for the property in an application blueprint.  
If the property type is Computed, the **Overridable in Blueprint** check box is not applicable.  
If a property is not overridable in the blueprint, you cannot set the Auto-Bind type to Consume.  
For example, the catalog administrator might configure the vFabric tc Server service to have a JVM heap size of 512MB. But for large deployments, the application architect might change the setting to 1024MB. |
| **Delete button**          | Removes the selected row from the Properties section.                                                                                                                                                                                                                   |

After a variable is added and defined, you can create an install, configure, start, or update script for the service version.
7 In the Actions section, create install, configure, start, or update a script.

The catalog administrator must provide an action script for at least one of the life cycle stages. For deploying an application, you can create an install, configure, and start script, or create an update script to update an existing deployment. These scripts are customized to use the component properties.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lifecycle Stage</td>
<td>You are not required to add scripts for all the life cycle stages. If you do not need a particular stage, ignore it.</td>
</tr>
<tr>
<td>Script Type</td>
<td>From the drop-down menu, select a script type for your action script. You can author in bash script for Linux-based applications, or Windows CMD or PowerShell script for Windows-based applications.</td>
</tr>
<tr>
<td>Script</td>
<td>In the Script column, click the Edit icon to open the Edit Script dialog box. You can write the script or copy a script into the dialog box. A catalog administrator can parameterize the installation and configuration of services. The properties that are defined for a service can be used inside the script. Click the down arrow in the script type field to select an action script type for a life cycle stage. To insert the properties you defined, click the down arrow in the Select a property to insert list. You can also select the Reboot check box to restart the virtual machine after the script runs successfully, during an application deployment. Click OK when you are finished.</td>
</tr>
<tr>
<td>Reboot</td>
<td>Select the check box so that the agent bootstrap can restart the virtual machine after an action script runs successfully. After the virtual machine is restarted, the agent proceeds to the next life cycle stage script defined in the service version.</td>
</tr>
<tr>
<td>Delete button</td>
<td>Clears the script but does not remove the row for the life cycle stage.</td>
</tr>
</tbody>
</table>

8 When you are finished creating the service version, click Save and OK.

The service you created is added to the page. If you selected the Pre-install in a Template check box, the service appears in the list of services that you can include when you create or edit a logical template and appears in the OS Templates section of the application blueprint. If you did not select this check box, the service appears in the list of services that you can include when you create a deployment blueprint for an application.

What to do next

Create a copy of an existing service version, edit, or delete the service. See "Maintain Service Versions," on page 76.

Maintain Service Versions

You can copy or edit a service from an existing service version.

Prerequisites

- Verify that your user account has the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.
- If you plan to edit the properties and scripts contained in services that you add to the application blueprint, familiarize yourself with the tasks described in “Add a Service to the Catalog,” on page 73.
- If you plan to use a script that downloads software from an external Web site, verify that the virtual machine you use for deploying the application has access to an external network.
- Familiarize yourself with the basic concepts of defining and configuring component properties and actions. See Chapter 8, “Developing vFabric Application Director Components,” on page 59.
- Create a copy of the service to preserve the original for future reference.
Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Catalog > Services.

2. Copy a service from an existing service version.
   
a. Open a service version and click Copy in the toolbar.
      
      When you copy a service, all of the properties, action scripts, operating systems, a list of descriptive tags, and supported application components are retained. You cannot change the name of the existing service version.
   
b. Change the version number, update the description, and click Save.
      
      The version number and description differentiate the new service from the parent version.

3. To modify the variables or update the action scripts, click Edit in the toolbar, make the changes, and click Save.

What to do next

You can delete the services that you do not use to model an application blueprint. See “Delete a Service from the Catalog,” on page 77.

Delete a Service from the Catalog

You can delete a service from the Services page.

Prerequisites

- Verify that your user account has the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.
- Verify that the service is not used in a blueprint or included as a preinstalled service in a logical template.

Delete the referenced blueprint or logical template that contains the preinstalled service.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Catalog > Services.

2. Open the service.

3. Remove the existing service versions and confirm your selection.

4. Click the arrow next to the service name to return to the Services page.

5. Delete the service and confirm your selection.

Add a Custom Task to the Catalog

With vFabric Application Director, you can create a custom task to perform customized tasks in the application deployment such as run security patches.

The vFabric Application Director catalog contains predefined tasks to configure APT or YUM repositories, a script to register a machine to the Red Hat network, or a script to enable a virtual machine to connect to the designated Windows Active Directory domain. You can add these customized tasks to the execution plan in a deployment profile. In some cases, rather than creating a custom task, you might prefer to edit a predefined task.

Note: Save your changes frequently. The vFabric Application Director user interface session expires after 30 minutes of inactivity. If the session expires, any changes that were not saved are lost.
Prerequisites

- Verify that your user account has the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.
- Verify whether you must add items to the operating systems and descriptive tag lists. See “Add Operating Systems to the Catalog,” on page 80 and “Add Tags to the Catalog,” on page 81.
- If a script in a task requires access to URLs from outside the corporate firewall, configure the applicable services and application components to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- Familiarize yourself with the basic concepts of defining and configuring component properties and actions. See Chapter 8, “Developing vFabric Application Director Components,” on page 59.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Catalog > Tasks.
2. Click New in the toolbar.
3. Set the name of the custom task, add a description, and click Save.
   
   As a best practice, use a name that corresponds to the task the script performs.
4. Click Create Task Version to create a custom task version.

   You can create multiple versions for a custom task.
   
   A page appears for creating a custom task version.
5. Complete the custom task version information.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Task version name stays the same.</td>
</tr>
<tr>
<td>Description</td>
<td>Add detailed information about the custom task. For example, describe what the script does when added to a node in the execution plan.</td>
</tr>
<tr>
<td>Version</td>
<td>You can specify major, minor, or micro releases, with or without qualifiers. For example, you might use unique version numbers such as 1.0 or 1.5.</td>
</tr>
<tr>
<td>Supported OSes</td>
<td>If the scripts used in this task can run only on particular operating systems, select those operating systems here. In the execution plan, vFabric Application Director prevents the custom task from appearing in the Catalog Task Name list unless it contains one of these operating systems. You must add at least one operating system in the text box.</td>
</tr>
<tr>
<td></td>
<td>To create an operating system name that is not in the list, click Cancel, and select Catalog &gt; Operating Systems.</td>
</tr>
</tbody>
</table>
6. In the Properties section, define the variables used for a custom task.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property Name</td>
<td>Click the first row of the Property Name column to define a variable. For example, you might create a custom task to configure a repository or create a custom email task to send a notification email when the deployment task for a service or application component successfully finishes.</td>
</tr>
<tr>
<td>Description</td>
<td>Include details about the custom task.</td>
</tr>
<tr>
<td>Type</td>
<td>To change the type, select a type from the drop-down menu. The available property types are String, Content, and Array. You cannot add types to the menu.</td>
</tr>
<tr>
<td>Value</td>
<td>Type the value to substitute for this property when the custom task runs. For example, for a property called remove_all, you might type either true or false in this text box.</td>
</tr>
</tbody>
</table>
### Option Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Secured     | Select the check box for passwords you define or other properties whose values you want to obscure. For example, the JBOSS_JMX_PWD property in the JBoss service is secured.  
  If a property is changed from Secured to Unsecured, vFabric Application Director resets the property value, for security purposes. You must set a new value for the property. |
| Delete button | Removes the selected row from the Properties section.                                                                                     |

7 In the Script Editor section, add a script for the custom task.

A catalog administrator can parameterize the installation and configuration of services. The properties that are defined for a service can be used inside the script.

a Expand the Edit script dialog box, set the script type from the drop-down menu to write a script or copy a script.

b Select the **Reboot** check box to restart the virtual machine after the script runs successfully, during an application deployment.

c Click the down arrow in the Select a property to insert list to add custom task properties.

d Click **OK** when you are finished.

8 When you finish creating the custom task, click **Save**.

The custom task you created is added to the Tasks page.

**What to do next**

You can add a custom task in the execution plan and deploy it to a deployment environment. See “Add Custom Tasks,” on page 94. Modify the custom task to support your current needs. See “Maintain Custom Task Versions,” on page 79.

**Maintain Custom Task Versions**

You can configure the script or redefine existing properties for a custom task version. To preserve the original custom task version you can create a copy and modify the copied version.

**Note** If a custom task is used in a deployment profile to customize an application deployment, the custom task cannot be deleted.

**Prerequisites**

- Verify that your user account has the **ROLE_CATALOG_ADMIN** catalog administrator role assigned to it.
- Verify whether you must add items to the operating systems and descriptive tag lists. See “Add Operating Systems to the Catalog,” on page 80 and “Add Tags to the Catalog,” on page 81.
- If a script in a task requires access to URLs from outside the corporate firewall, configure the applicable services and application components to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- Familiarize yourself with the basic concepts of defining and configuring component properties and actions. See Chapter 8, “Developing vFabric Application Director Components,” on page 59.

**Procedure**

1 On the vFabric Application Director title bar, click the drop-down menu and select **Catalog > Tasks**.
2 Copy a custom task from an existing custom task version.
   a Open the custom task version and click **Copy** in the toolbar.
      All of the properties, script, and operating systems are retained. You cannot change the name of the
      existing custom task version.
   b Change the version number, update the description, and if needed update the operating system.
      The version number and description differentiates the new custom task from the parent version.
   c Click **Save**.

3 To change existing properties or an existing script, click **Edit** in the toolbar, make your changes, and click **Save**.

### Add Operating Systems to the Catalog

You might have to add operating systems to the list of preinstalled operating systems that
vFabric Application Director includes.

You can create a definition of an operating system in the list and specify which operating system a logical
template uses or which operating systems are supported for a particular service. For
vFabric Application Director 5.0, Linux and Windows operating systems are supported.

---

**Note** You cannot delete an operating system from the catalog.

**Prerequisites**

- Verify that your user account has the **ROLE_CATALOG_ADMIN** catalog administrator role assigned to
  it.
- Familiarize yourself with the preinstalled operating systems on the Operating Systems page.

**Procedure**

1 On the vFabric Application Director title bar, click the drop-down menu and select **Catalog > Operating
   Systems**.

2 Click **New** in the toolbar.

3 Set the operating system name and description.
   As a best practice, use the OS name and version number, and add the bit information in the description.

4 Select the architecture from the drop-down menu.
   For 32-bit operating systems, select **x86_32** and for 64-bit operating systems, select **x86_64**.

5 Select the operating system family from the drop-down menu and click **Save**.
   When you create an action script for a service, the script type depends on the designated operating system
   family. You use a bash script for a Linux-based application and a Windows CMD or PowerShell script for
   a Windows-based application.

6 Click **Create OS Version** to create an operating system version.
   You can create multiple versions for an OS.
   A page opens for creating an OS version.

7 Complete the OS version information and click **Save**.
   The OS name remains the same. You can the specify major and minor versions and add service pack details
   for the OS.
The new operating system is added to the Operating Systems page. You can select the operating system when you create other components to add to the catalog or to an application blueprint.

**Add Tags to the Catalog**

When you create a logical template or a service, you can associate one or more tags with the component.

Setting a tag creates compatibility restriction for the services and components that you can add to a node in the blueprint canvas. For example, you might add a Database Server tag to a service that installs and configures a database component of an application. After you set the tag type and save the tag, if you change the tag type your change might remove the service from the blueprint canvas. You might also receive errors when you configure the service.

**NOTE** You cannot delete a tag from the catalog.

**Prerequisites**

- Verify that your user account has the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.
- Familiarize yourself with the preinstalled tags listed on the Tags page.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Catalog > Tags**.
2. Click **New** in the toolbar.
3. Type the tag name and description in the text box.
4. Select a tag type from the drop-down menu.
   - The Property Descriptor tag type appears as an option in the **Auto-Bind Tags** drop-down menu for a service property. The Server Type tag appears as an option in the drop-down menu when you create a service or logical template.
5. Click **Save** when you are finished.

The newly created tag is listed on the Tags page. You can select the tag when you add a service, task, or logical template to the catalog or to an application blueprint.
Creating Applications

vFabric Application Director provides a drag-and-drop canvas that application architects can use to model an application blueprint. With this blueprint, you can create applications for deployment on a cloud.

You can drag the following items to the blueprint.

- Logical templates from the catalog
  
  You can use the vFabric Application Director sample templates or create custom templates. The cloud templates that contain operating system images and might have services preinstalled and configured are mapped to vFabric Application Director logical templates.

- Application infrastructure components and scripts from the catalog
  
  The included sample services are reusable software components. vFabric Application Director includes installation and configuration scripts that follow best practices for services.

- Application components
  
  These applications operate on top of the services. After you configure an application component, you can deploy it on a compatible service or an operating system image.

**NOTE** For information about deleting an application deployment from vFabric Application Director, see “Delete an Application Deployment from vFabric Application Director,” on page 131.

Familiarize yourself with the key concepts that appear frequently in topics about creating applications. See “Key Concepts,” on page 12.

This chapter includes the following topics:

- “Create an Application,” on page 83
- “Creating an Advanced Blueprint,” on page 86
- “Copy an Application Version,” on page 89
- “Copy an Application,” on page 89
- “Delete an Application Version,” on page 90

Create an Application

With vFabric Application Director, you can compose your application deployment topology, create dependencies, and edit configurations.

The application blueprint provides detailed control over installation dependencies, configuration changes, and editable scripts. vFabric Application Director generates execution plans from the blueprint, which you can revise and use to deploy applications on services based on compatibility.
You model and create an application in the blueprint canvas. On the left side of the blueprint canvas are the logical templates from the vFabric Application Director catalog. On the right is a list of the services available from the catalog and application components. You can select a logical template and drag it to center of the blueprint canvas to start modeling your application.

**Note**  Save your changes frequently. The vFabric Application Director user interface session expires after 30 minutes of inactivity. If the session expires, any changes that were not saved are lost.

**Prerequisites**

- Verify that your user account has the ROLE_APP_ARCHITECT application architect role assigned to it.
- To examine the sample applications and their components, log in to vFabric Application Director with a user account that belongs to the Default group.
- If the application requires access to URLs from outside the corporate firewall, configure the applicable services and application components to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- If you plan to create clustered nodes, create action scripts that contain clustering logic so that you can add these scripts to the blueprint. See “Defining Component Actions,” on page 59.
- Familiarize yourself with the basic concepts of binding to another property if you plan to customize your application. See “Binding to Other Properties,” on page 64.
- For ideas about how to name the application and what text to put in the description text boxes, see the predefined applications already included on the Applications page. To see these applications, you must log in as a user that is a member of the Default group.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click **New** in the toolbar.
3. Set the name of the application, add an optional description, and when you are finished, click **Save**.
4. Click **Create App Version** to create an application version, add the major and minor version of the application, and when you are finished, click **Save**.
   For example, you might also add a qualifier to the major and minor versions such as **1.0.0-SNAPSHOT**.
5. Click **Create Blueprint** to create an application blueprint.
   The canvas for modeling the application blueprint appears.
6. Select and drag one or more logical templates to the canvas to create nodes.
   For example, to create a three-tiered application, you might drag three items from the OS Templates list to the canvas, or drag one template from the OS Templates list, one item from the Database Servers list, and one from the Application Servers list. The names of the lists correspond to the tags associated with a template.
7. (Optional) Select one of the nodes on the canvas and change the name of the node on the **Details** tab below the blueprint.
   The name of the blueprint node must meet the following naming conventions:
   - Limited to 15 characters.
   - If the node name contains a character other than a letter, digit, underscore, or hyphen, that character is replaced with a hyphen.
   For example, for a three-tiered application, you might rename each of the three nodes as **Application_Server**, **Database_Server**, and **Load_Balancer**.
(Optional) Set a host name on the **Details** tab below the blueprint if you plan to deploy the application to vCloud Director.

vCloud Director uses this host name as an identifier for the virtual machine computer name in its network. If the host name text box is left blank, vFabric Application Director generates a host name with random characters.

The host name must meet the following naming conventions:

- Limited to 15 characters.
- Must begin with a letter.
- Can contain a letter, digit, or hyphen, but cannot end with an hyphen.
- Cannot have the same host name as another node in the application blueprint.
- Can include a `{$random}` expression at the end of a host name to generate unique characters.

For example, an Apache{$random} host name might generate characters such as Apache9INOIK3YT after an application is deployed. You can view the host name with unique characters from the deployment summary page.

For a clustered node, the host name cannot exceed 15 characters with the appended node array index. For example, a clustered node called AppServer has the host names of the virtual machines in the cluster as AppServer-1, AppServer-2, and so on.

(Optional) To change the default number of CPUs or amount of memory for a deployed virtual machine, select the relevant node on the canvas and edit the values on the **Details** tab below the blueprint.

The tabs that appear below the blueprint correspond to the node selected.

The CPU and memory values in the blueprint might not match the corresponding values in the virtual machines created in Amazon EC2. Because Amazon EC2 allows virtual machines with specific CPU and memory combinations, it uses the values you assign in a blueprint to determine the closest possible match.

Select and drag one or more services or application components to the nodes.

For example, you might drag the MySQL service onto a database server node, drag the JBoss service onto an application server node, and drag an Apache service onto the Load Balancer node.

If a service or application component is not compatible with a particular node, you cannot drop it on the node. For example, you can drag the application component called SQL SCRIPT onto a MySQL service, but you cannot drag the SQL SCRIPT component onto a JBoss service.

Compatibility restrictions are created when the catalog administrator sets the supported operating systems and components for a catalog service. The catalog administrator can also add to the list of operating systems and tags that are already available in the catalog. For example, the MySQL service in the catalog has the supported components listed as SQL SCRIPT. Only the SQL SCRIPT application component type can be added to the MySQL service.

In addition, MySQL service has the supported operating systems set to CentOS32 5.6.0, CentOS64 5.6.0, and Ubuntu32 10.4.3. The MySQL service can be added to logical templates that include one of the operating systems.

**NOTE** If any preinstalled services are added to a logical template after you create a blueprint, the new preinstalled services are not added to the node. In this case, you must recreate the node and add the preinstalled services.

You can add the application components SCRIPT and Other to a node or any service.
11 (Optional) Select a service or application component and edit the information on the Details and Actions tabs below the blueprint.

Only those properties that the catalog administrator designated as overridable can be changed on the application blueprint.

On the Actions tab, scripts are accessible for all stages of the component’s life cycle, including install, configure, start, and update. A catalog administrator can edit a service script from Catalog > Services.

12 To edit a property, click the property in the table.

If you have a single or clustered node, you must provide a value for the required property in at least one of the life cycle stages of the property. For example, to run an Apache Tomcat server, Java is required and the JAVA_HOME property value must be set.

The Edit Property dialog box appears.

13 (Optional) To bind a property to another property, select the property value from the Blueprint Value drop-down menu in the Edit Property dialog box.

Binding to another property lets you customize a script based on the value of other node’s run time property values such as IP addresses.

14 When you are finished creating the blueprint for the application, click Save.

vFabric Application Director checks the application topology you created and displays a message box listing any errors. For example, you see a message if a property type that you selected is not compatible with a script type, or if a service or component is missing a required script. Some errors require correction before you can save the application.

What to do next
Create a deployment profile. See “Create a Deployment Profile,” on page 91.

Creating an Advanced Blueprint

With vFabric Application Director, you can create an advanced application blueprint with dependencies between components, clustered nodes, and multiple networks.

Figure 10-1. Clustered Windows Application Example
Create a Dependency Between Components on page 87

Dependencies are added in the blueprint to define an order in which the deployment tasks must be performed. Creating a dependency link from one item such as a service or application component to another guarantees that the task of creating the first item finishes successfully before a second task begins.

Specify a Node as a Cluster on page 87

For scaling deployments, you might need to deploy multiple virtual machines or a cluster for a particular node and use a load balancer to manage them.

Define Multiple NICs for a Node on page 88

In most deployments, some servers are deployed to a DMZ zone and some servers are deployed to a network protected by a firewall.

Create a Dependency Between Components

Dependencies are added in the blueprint to define an order in which the deployment tasks must be performed. Creating a dependency link from one item such as a service or application component to another guarantees that the task of creating the first item finishes successfully before a second task begins.

Procedure

1. To create dependencies between services or application components, click the Add Relation button ( ) in the toolbar above the canvas.

2. Select the first component, then the component it is dependent on.

   For example, because a load balancer usually cannot be configured until the application is up and running, you might add a dependency from an Apache service to a WAR component.

   A blue dotted line appears and points to the dependent component.

3. When you are finished, click Save in the toolbar above the canvas.

Example: Create Dependency from JBoss to MySQL

For example, you might draw a dependency line from the JBoss service to the MySQL service.

This line indicates that the application server is dependent on having the database created and configured.

What to do next

Deploy the application. See Chapter 11, “Deploying Applications,” on page 91.

Specify a Node as a Cluster

For scaling deployments, you might need to deploy multiple virtual machines or a cluster for a particular node and use a load balancer to manage them.

Prerequisites

Familiarize yourself with the basic concepts of binding to another property, node array index property, and defining component actions. See Chapter 8, “Developing vFabric Application Director Components,” on page 59.

Procedure

1. To specify a cluster of virtual machines, click the Convert to Node Array icon ( ) in the node.

2. Below the blueprint, set the cluster size.
3 Bind to a property like node_array_index to identify in which virtual machine the current script is running. You can find the IP addresses of all the virtual machines in a cluster by binding a property to all(node:ip).

4 (Optional) If other properties refer to a cluster property, define the component properties to access the array of property values from the clustered nodes.

5 Click Save in the toolbar above the canvas.

Example: Specifying the Load Balancer
To understand how a cluster is used, see the Clustered Duke’s Bank sample application. In the Load Balancer node, click Apache_LB and the http_node_ips property refers to all(appserver:ip).

What to do next
Deploy the application. See Chapter 11, “Deploying Applications,” on page 91.

Define Multiple NICs for a Node
In most deployments, some servers are deployed to a DMZ zone and some servers are deployed to a network protected by a firewall.

In the Clustered Duke’s Bank sample application, the Load Balancer node is the only node that you should access from a public network. The Database and Appserver nodes must be deployed in a private network behind a firewall. The Load Balancer node must also have access to the Database and AppServer nodes.

In vFabric Application Director, to resolve this situation, you can define two NICs on the Load Balancer. Each NIC must specify a logical network name. At deployment time, the logical network is mapped to an actual cloud network. When a virtual machine is created, the number of NICs for the virtual machine are derived from the node.

In the Duke’s Bank sample application, the Load Balancer node has two NICs, NIC0 pointing to ServiceNetwork and NIC1 pointing to MgmtNetwork. Database and Appserver nodes have one NIC pointing to the ServiceNetwork. At deployment time, ServiceNetwork can be mapped to a cloud network protected by firewall and MgmtNetwork can be mapped to a public cloud network.

Prerequisites
Familiarize yourself with the predefined IP address property concept when you have multiple NICs in a virtual machine. See “Predefined IP Address Property,” on page 66.

Procedure
1 To add multiple NICs, select a node and click the NICs tab below the node. You can add up to 10 NICs to a node.

2 To add a NIC, click the Add icon (+) and specify a logical network name.

3 To be part of the same network as another node, pick the network name from the drop-down menu or type a new network.

4 When you are finished, click Save in the toolbar above the canvas.

What to do next
Deploy the application. See Chapter 11, “Deploying Applications,” on page 91.
Copy an Application Version

When you copy an existing application version, all of the blueprint contents are copied except deployment profiles, and the application name remains the same. You can only edit the application version and version description.

Prerequisites

- Verify that your user account has the ROLE_APP_ARCHITECT application architect role assigned to it.
- If you are not familiar with the process of adding components to an application blueprint, creating dependencies between components, or modifying services and scripts, see “Create an Application,” on page 83.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Open an application and select an application version.
3. Click the Copy button in the toolbar to copy the application version.
4. Change the application version, add a description, and click Save.
   The copied application version opens.

What to do next

Create a deployment profile. See “Create a Deployment Profile,” on page 91.

Copy an Application

Copying an application allows you to keep all of the components of an application version, set a new application name, and change the application version into a standalone application.

Prerequisites

- Verify that your user account has the ROLE_APP_ARCHITECT application architect role assigned to it.
- If you are not familiar with the process of adding components to an application blueprint, creating dependencies between components, or modifying services and scripts, see “Create an Application,” on page 83.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. From the Applications page, point to the application card view.
3. Click the Copy button to copy the application.
4. Select an existing application version from the drop-down menu.
5. Set a new name for the application and add a description.
   You might include the details about the configured properties and services, dependencies, node configuration, or custom tasks in the execution plan.
6. Click OK.
   The new application appears on the Applications page.
What to do next
Create a deployment profile. See “Create a Deployment Profile,” on page 91.

Delete an Application Version
If your application version becomes obsolete, you can remove it from the vFabric Application Director appliance.
When you delete an application version, the blueprints are removed. You cannot delete an application version with deployment profiles.

Prerequisites
- Verify that your user account has the ROLE_APP_ARCHITECT application architect role assigned to it.
- If you are not familiar with the process of adding components to an application blueprint, creating dependencies between components, or modifying services and scripts, see “Create an Application,” on page 83.

Procedure
1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Open an application and select an application version.
3. Click Delete and OK to confirm.

The application version is removed and the Application Versions list appears.
vFabric Application Director simplifies and automates deployments of multitier enterprise applications in hybrid cloud environments.

With vFabric Application Director, you can create different deployment profiles for a life cycle, such as development, testing, and production. The deployment profile settings are saved in vFabric Application Director. You can reuse this saved deployment profile to deploy an application version to a supported cloud environment.

If you update a deployed application, vFabric Application Director creates an update profile based on the modified values. This update profile includes an update script generated from the defined update life cycle stage.

Familiarize yourself with the key concepts relating to deploying applications. See “Key Concepts,” on page 12.

This chapter includes the following topics:
- “Create a Deployment Profile,” on page 91
- “Add Custom Tasks,” on page 94
- “Deploy an Application,” on page 96
- “Updating Application Deployments,” on page 97
- “Understanding the Deployment and Update Process,” on page 103
- “Checking the Status of a Deployment,” on page 104

Create a Deployment Profile

Deployment profiles govern settings such as cloud templates, networks, and application configuration values allowed for use in specific deployment environments.

Create a different deployment profile for each deployment environment. In a deployment profile, you can enter or override application properties for a specific deployment if the Override at Deployment option is enabled for the property. For example, for a particular deployment environment, you might change the database port to 3307.

Deployment execution plans are generated by the system based on the blueprint. You can review the execution plan before deploying the application. You can also add custom tasks to the execution plan. Custom tasks help application release teams by communicating with third-party software for approvals and sending notifications. See “Add Custom Tasks,” on page 94.
The deployment profile also contains a network list of vCloud Director and Amazon EC2 networks. For vCloud Director, vFabric Application Director supports external networks and vCloud routed networks with or without DHCP. If the network list is empty, contact your vCloud Director administrator. For Amazon EC2, vFabric Application Director supports the NAT-routed, public, and private networks. NAT-routed and public networks can access the Internet. A private network that is not NAT-routed can access only the vFabric Application Director appliance.

**Prerequisites**

- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Verify that at least one cloud template is mapped to each logical template used in the blueprint. See “Add a Logical Template to the Catalog,” on page 69.
- Verify that at least one application is created in vFabric Application Director. See Chapter 10, “Creating Applications,” on page 83.
- Depending on your cloud environment, you must have at least one vCloud Director or Amazon EC2 network available for the deployment environment.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click the name of the application.
   
   A list of application versions appear.
3. Select an application version and click **Create DP** to create a deployment profile that you can save or deploy immediately.

4. (Optional) To create another deployment profile, click **New**.
5. Set a name for the deployment profile and click **Deploy**.
   
   You might name the profile so that the name indicates which type of environment is used, specific override property, or clustered node configuration.
   
   For example, for an application called MyTimecard, you might name the profile myTimecard-QA to indicate that this profile is for the QA environment rather than for the production environment.
   
   The Deployment Profile wizard appears, with the Deployment Environment page highlighted.
6. Configure the deployment environment.
   
   a. Select a deployment environment from the list and click **Map Details**.
      
      Map details retrieve the list of cloud templates and networks available in the deployment environment. The nodes listed in the VM Templates section correspond to the components of the application, as shown in the application blueprint. The Networking section lists the logical networks defined in the blueprint.
      
      You can view the corresponding list for the Deployment Environment section, click the title bar and select **Clouds > Deployment Environments** from the drop-down menu.
   
   b. In the VM Templates section, map a logical template to a cloud template in the cloud environment.
      
      Cloud templates that belong to the same group as the user appear in the drop-down menu. If the list of cloud templates is empty, the existing cloud templates do not belong to your group or a cloud template was not registered. If you have access, log in to the group that has existing cloud templates, or register at least one cloud template in vFabric Application Director. You can also ask your cloud administrator to register a cloud template for you.
In the Networking section, select a supported cloud network for each logical network in the catalog.

The network you select depends on the deployment environment. To deploy to Amazon EC2, you must carefully determine the virtual machine you put on an external network. Every NIC on an external network gets an Elastic IP address, which puts that interface on the Internet. Put a NIC on an external network only when it is absolutely required.

For example, for a load balancer, if you are deploying the application to a test environment, you might select an internal network for both load balancer network NICs. When you create a deployment profile for the production environment, you might select an internal network for one load balancer NIC and an external network for the other load balancer NIC.

d When you finish making your selections, click Next.

7 Verify that the properties you set in the blueprint appear accurately in the component tabs.

You can define new values for all the node properties. In addition, you can define new values only for application component and service properties that have the Overridable at Deployment check box selected in the application blueprint.

Required node properties should be assigned a value from the catalog, blueprint, or deployment profile for successful deployment. Node properties, such as memory allocation and number of CPUs, are defined by the system but can be overridden. You can also set a host name so that the virtual machine can be easily identified in the vCloud Director deployments.

For example, the vFabric tc Server service might have a JVM heap size of 512MB. But for a large deployment, you can override that setting and change the size to 1024MB.

a (Optional) Click the Service, Application Component, or Node tab.

b Select a specific service, application component, or node property.

c In the table, click the New Value column of a specific row and type the value to use in the deployment profile.

For a deployment to Amazon EC2, you must change the global_conf property value to https://DarwinServerIP:8443/darwin/conf/darwin_global_noproxy.conf.

d To revert to the original value, click the Reset Value button ( ).

e When you finish making your changes, click Next.

8 Review the components and dependencies in the execution plan, click the Expand Cluster button ( ) if the node is clustered, and click Next.

The blue dotted lines in the execution plan define a specific order in which the deployment tasks will run.

For example, because a load balancer usually cannot be configured until the application is running, you might add a dependency in the blueprint.

For services and components that have scripts associated with them, click the down arrow next to the component or service name to view the script or the variable definitions used in the script.

9 Review the settings for the deployment profile, click Save, and click OK to confirm.

The deployment profile is listed for the application version.

What to do next

You can add a custom task to the deployment profile or use the deployment profile to deploy the application. See “Add Custom Tasks,” on page 94 or “Deploy an Application,” on page 96.
Use an Existing Deployment Profile

Instead of creating a deployment profile for deployment, you can reuse an existing deployment profile for an application version.

Prerequisites
- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Verify that at least one cloud template is mapped to each logical template used in the blueprint. See “Add a Logical Template to the Catalog,” on page 69.
- Verify that at least one application is created in vFabric Application Director. See Chapter 10, “Creating Applications,” on page 83.
- Depending on your cloud environment, you must have at least one vCloud Director or Amazon EC2 network available for the deployment environment.

Procedure
1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Click the name of the application.
   A list of application versions appear.
3. Select an application version and click the existing deployment profile.
4. If you created new nodes in the blueprint, map the new nodes to a cloud template.
   Property overrides saved in an existing deployment profile remain overridden in the new value column even if the blueprint value is updated. Deleted properties no longer appear in the deployment profile.

What to do next
Complete the Deployment Profile wizard and deploy the application. See “Create a Deployment Profile,” on page 91 and “Deploy an Application,” on page 96.

Add Custom Tasks

In the execution deployment plan, you can add custom tasks to perform additional customized tasks in the application deployment.

A blueprint helps generate a common execution plan for an application on all of the deployment environments. In some cases, the execution plan must be customized for each deployment environment. For example, when an application is deployed to the production deployment environment, you might need to send an email after deploying, but in the test deployment environment, such checks might not be required. You can create an email custom task to send a notification email when the deployment task for a service or application component successfully finishes and add this task to the execution plan in the deployment profile, which deploys to the production deployment environment.

Verify that no processes are prompting for user interaction when the custom task is running. Any interruption pauses the task, causing it to remain in an idle state indefinitely. You can cancel the application deployment after an hour or vFabric Application Director fails the deployment in an idle state after three days.

Prerequisites
- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.

Verify that at least one cloud template is mapped to each logical template used in the blueprint. See “Add a Logical Template to the Catalog,” on page 69.

Verify that at least one application is created in vFabric Application Director. See Chapter 10, “Creating Applications,” on page 83.

Verify that at least one custom task is created in the vFabric Application Director catalog. See “Add a Custom Task to the Catalog,” on page 77.

Verify that all of the required node properties in the application are assigned a value for successful deployment.

Depending on your cloud environment, you must have at least one vCloud Director or Amazon EC2 network available for the deployment environment.

Familiarize yourself with the basic concepts of defining and configuring component properties and actions. See Chapter 8, “Developing vFabric Application Director Components,” on page 59.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click the name of the application.
   A list of application versions appear.
3. Select an application version and create a deployment profile or use an existing one.
4. In the Deployment Profile wizard, follow the prompts until you get to **Execution Plan**.
5. Click the **Expand Cluster** button, if the node is clustered.
   If the clustered node is not expanded, the custom task is added only to the first virtual machine in the cluster. If an application architect changes a node to a clustered node, the custom task is added to only the first virtual machine in the cluster.
6. Select the **Add Script Task** button and drag the custom task to the node.
   When you drag the **Add Script Task** button, you see anchors that indicate where you can insert the custom task. For a clustered node, add the custom task to each node.
   For example, you can drag one or more custom tasks to the Application Server, Database Server, or Load Balancer node.
   After you drop a custom task onto a node, the Add Custom Task dialog box opens.
7. In the Add Custom Task dialog box, select a task from the **Catalog Task Name** drop-down menu.
   The supported operating systems of the custom task appear. For example, if a custom task is supported on the CentOS 5.6 operating system and the operating system of the node is Ubuntu 10.04, the task is not listed in the menu.
   When you select a custom task, the task, script, and property details appear in the dialog box.
8. (Optional) To override a property value on the **Properties** tab, click the property.
   For example, in a send email custom task, one of the properties is the recipient’s email address. You can set the property email address value to the recipient email address.
   a. In the Edit Property dialog box, type the new value for the property or select an existing property from the drop-down menu to bind the property to one of the properties in the application blueprint.
   b. Click **Save**.
In the Add Custom Task dialog box, review the script and property details of the custom task, and click OK.

Click Next to review the deployment profile settings, and click Deploy.

The deployment summary page has several windows that display the progress of the deployment in various ways.

What to do next

Explore the ways to check the status of the in-progress deployment. See “Checking the Status of a Deployment,” on page 104.

Deploy an Application

After you save a deployment profile, you can deploy the application from the vFabric Application Director user interface.

You can also deploy an application from the command-line interface. See “Deploying and Updating an Application Using CLI,” on page 146.

Prerequisites

- Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.
- Verify that at least one deployment profile is in vFabric Application Director. See “Create a Deployment Profile,” on page 91.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Click the name of the application.
   A list of application versions appear.
3. Select an application version and create a deployment profile or use an existing one.
4. Follow the prompts in the Deployment Profile wizard and make any applicable changes.
5. Review the deployment profile settings.
   You can make necessary changes to the deployment profile before deploying the application.
6. Click Deploy.
   A deployment summary page appears and refreshes in approximately 30 seconds to display the deployment status.
7. (Optional) Click Refresh in the toolbar to update the real-time status of the deployment.

What to do next

Learn about the processes that take place in the background when an application is deployed to the cloud. See “Understanding the Deployment and Update Process,” on page 103.

You can also check the status of an in-progress deployment. See “Checking the Status of a Deployment,” on page 104.
Quick Deploy an Application

When you tear down a deployed application from the cloud environment, you can quickly redeploy the application without configuring the elements in the Deployment Profile wizard.

With quick deploy, you can also update required properties and overridable properties outside the Deployment Profile wizard and deploy the application blueprint.

When you quick deploy an application, the latest application version and associate deployment profiles are available for deployment. To quick deploy an older application version, open the application and quick deploy the specific application version.

Prerequisites

- Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.
- Verify that at least one deployment profile is in vFabric Application Director. See “Create a Deployment Profile,” on page 91.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. From the Applications page, select the application to deploy.
3. Select an application version to Quick Deploy.
4. Click the **Quick Deploy** button to initiate the deployment process.
   - If an application version is not selected, by default the latest application version is deployed.
5. In the **Destination** drop-down menu, select the associated deployment profile.
   - The properties that are required for deployment and overridable at deployment appear.
6. (Optional) Set a new value for a required or overridable property and click **Deploy**.
   - The modified value for a required or overridable property is not saved in the deployment profile for future deployments.

The deployment summary page appears and displays the status of the in-progress deployment.

What to do next

Use the status windows on the deployment summary page to track the deployment status. See “Checking the Status of a Deployment,” on page 104.

Updating Application Deployments

When you update an existing application deployment you create a process that captures new values for the changes required for that update.

**IMPORTANT** To update deployed applications, you must install the vFabric Application Director for Release Automation edition.

You can deploy a saved deployment profile multiple times to create application deployments. With vFabric Application Director 5.0, you can initiate an update process for existing deployments in vCloud Director multiple times by scaling clusters of a node or modifying the configurations and code of existing services and application components.
When you scale node clusters of an existing application make sure that you have enough resources in the cloud to support the additional nodes in the application.

**NOTE** vFabric Application Director 5.0 does not support updating application deployments with Windows-based nodes and updating existing deployments in Amazon EC2.

To help you identify the changed and impacted properties in the application with dependencies, when you update the cluster size of the node, vFabric Application Director highlights the node as changed and the dependent property is highlighted as impacted in the Blueprint window on the wizard and review page. The component that contains the dependent property is highlighted as the impacted component.

![Figure 11-1. Changed and Impacted Components for an Update Process to Scale a Clustered Node](image)

When you update a property to modify a configuration, vFabric Application Director highlights the property as changed and the dependent property is highlighted as impacted in the Blueprint window on the wizard and review page. The component that contains the new property value is highlighted as the changed component. The component that contains the dependent property is highlighted as the impacted component.

**Initiate an Update Process to Scale Deployments**

With vFabric Application Director 5.0, you can initiate an update process for an application deployment multiple times by scaling the clusters of a node or modifying the configurations and code of existing services and application components. When you scale a clustered node of a deployed application, you can configure only the cluster size of the node that were modeled as clusters in the application blueprint.

During an update process, based on the update settings, virtual machines are created and required action scripts run on the new virtual machines. In a multitiered application, if a node depends on the scaled clustered node, an update script must run on the dependent node.

For example, in a deployed Clustered Dukes Bank App, you can scale the AppServer node to handle additional load. During the update deployment, the AppServer install, configuration, and start scripts run on the newly scaled virtual machine. Because the http_node_ips and appsrv_routes properties of the Apache_LB service are dependent on the AppServer node, changes in the AppServer cluster size affect the Apache_LB service and initiate the update script to run.
The update script is the UPDATE life cycle stage you define for a dependent service or application component during the initial application deployment. You can also add or modify the update script during the update process. When you configure the update script during the update process, the script is not saved for future deployments.

**NOTE** You cannot modify the install, configure, or start scripts during an update process. You can configure only the update script.

You can also initiate an update process to scale a deployed application from the command-line interface. See "Deploying and Updating an Application Using CLI," on page 146. With REST APIs in vFabric Application Director, you can automate the scaling of a deployed application. See *VMware vFabric Application Director API Programming* guide.

**Prerequisites**

- Verify that you installed the VMware vFabric Application Director for Release Automation edition to update a deployed application.
- Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.
- Familiarize yourself with the basic concepts of defining and configuring component properties and actions. See Chapter 8, “Developing vFabric Application Director Components,” on page 59.
- The deployed application must include at least one clustered node. See “Specify a Node as a Cluster,” on page 87.
- If you plan to add a custom task, verify that at least one custom task is created in the vFabric Application Director catalog. See “Add a Custom Task to the Catalog,” on page 77.
- Verify that the initial deployment is successfully deployed to a cloud environment. You cannot scale clustered nodes from a failed deployment to modify configuration.
- Contact your cloud administrator to get information about deployment environment storage space limit.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Deployments**.
2. Select an application deployment that has deployed successfully.
3. From the **Update** drop-down menu in the toolbar, select **Scale Out**.
4. Name the scaled update process, add an optional description, and click **OK**. Add information about the changes included in this update.
5. Click the blueprint image to review the highlighted dependencies between services or application components. Note all of the dependent components so that you can create update scripts for all of them.
6. From the **Node** tab, increase the cluster size value for one or more clustered nodes.
7. (Optional) From the **Service** or **Application Component** tab, create an update script for all of the available dependent components with a property bound to the clustered node. If the update script is defined in the UPDATE life cycle stage, then the script appears with the associated service or application component.
   
   For example, if a node has the all(node_array:ip) property bound to a clustered node, then it must run an update script.
8. (Optional) Select the Reboot check box if the agent must restart the virtual machine after the update script runs successfully and click Next.

The scaled clustered node and the update tasks of the dependent components appear in the execution plan. The original execution plan does not appear during an update process.

9. (Optional) Add an APPD_UPDATE_PROPS property in the update script to view a list of all the changed properties.

The update script is not saved and it applies to that particular update process only. You cannot use APPD_UPDATE_PROPS as a property name or as a qualifier for a property name.

For example, the sample MySQL service update script includes the APPD_UPDATE_PROPS property to update the database port and password.

10. (Optional) Add an APPD_PREV property in the update script to view the previous value of a property.

The update script is not saved and it applies to that particular update process only. You cannot use APPD_PREV as a property name or as a qualifier for a property name.

For example, the sample MySQL service update script includes the APPD_PREV property to view the value of the database password.

11. (Optional) Click the Expand Cluster button to expand the clustered node, select a custom task, and drag the task to each node.

You can configure the task properties in the Add Custom Task dialog box and save your changes.

12. Review the scaled clustered node and update script in the execution plan and click Next.

The blue dotted lines in the execution plan define a specific order in which the deployment tasks will run. Click the down arrow next to the service update script to view script details or the variable definitions used in the script.

13. Review the template and modified properties or actions in the update.

The dependent properties with defined update scripts are highlighted.

14. Click Update to deploy the updated application.

The update process deploys the scaled update to the cloud.

What to do next

You can check the status of the deployment from the deployment summary page. See “Checking the Status of a Deployment,” on page 104.

Learn about the various processes that take place in the background when the deployed application is updated in the cloud. See “Understanding the Deployment and Update Process,” on page 103.

Initiate an Update Process to Modify Configurations

You can initiate an update process to modify the configurations and code of existing services in a deployed application such as Tomcat and MYSQL, or application components such as WAR and SQL. When you modify the configurations of a deployed application, you can configure only the application property values.

After the initial deployment, if you modify existing application components or services or modify components that have dependent properties, an update script must run on all of the components. The update script is the UPDATE life cycle stage you define for a service or application component during the initial application deployment or in the update process.
For example, in the Clustered Dukes Bank App, if you modify the db_port property of the MYSQL service, during the deployment update process the update script runs on the MYSQL service. The Dukes_Bank_App application component on the Appserver node also runs the update script because the db_port property is bound to the db_port property of MYSQL service.

**Note** You cannot modify the install, config, or start scripts during an update process. You can configure only the update script.

You can also initiate an update process to modify the configuration and code of services or application components from the command-line interface. See “Deploying and Updating an Application Using CLI,” on page 146. With REST APIs in vFabric Application Director, you can automate the modification of configurations of a deployed application. See *VMware vFabric Application Director API Programming* guide.

**Prerequisites**

- Verify that you installed the VMware vFabric Application Director for Release Automation edition to update a deployed application.
- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
- Familiarize yourself with the basic concepts of defining and configuring component properties and actions. See Chapter 8, “Developing vFabric Application Director Components,” on page 59.
- If you plan to add a custom task, verify that at least one custom task is created in the vFabric Application Director catalog. See “Add a Custom Task to the Catalog,” on page 77.
- Verify that the initial deployment is successfully deployed to a cloud environment.
  You cannot initiate an update process to modify the configuration and code of an existing service from a failed deployment to scale a clustered node.
- The deployed application must have at least one service property or application component property that is Overridable at Deployment.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Deployments**.
2. Select an application deployment that has deployed successfully.
3. From the **Update** drop-down menu in the toolbar, select **Change Configuration**.
4. Name the configuration update process, add a description, and click **OK**.
   Add information about the changes included in this update.
5. Click the blueprint image to review the highlighted dependencies between services or application components.
   Note all of the dependent components so that you can create update scripts for all of them.
6. From the **Service** or **Application Component** tab, modify the applicable properties and update scripts.
   If the update script is defined in the UPDATE life cycle stage, then the script appears with the associated service or application component. You must create an update script for the changed and dependent component.
   In the case of a clustered node, if you modify a property of a service or an application component, the changes are applied to all the nodes of the cluster.
   For example, in the Clustered Dukes Bank App if you change the db_password property in the initialize_db_script application component, the db_password property in the Dukes_Bank_App depends on the initialize_db_script application component. The update scripts run on both because initialize_db_script is the changed component and Dukes_Bank_App is the impacted component. The update script for Dukes_Bank_App runs on all of the nodes of the Appserver cluster.
(Optional) Select the **Reboot** check box if the agent must restart the virtual machine after the update script runs successfully and click **Next**.

The update tasks of the changed and affected components appear in the execution plan. The original execution plan does not appear during an update process.

(Optional) Add an `APPD_UPDATE_PROPS` property in the update script to view a list of all the changed properties.

The update script is not saved and it applies to that particular update process only. You cannot use `APPD_UPDATE_PROPS` as a property name or as a qualifier for a property name.

For example, the sample MySQL service update script includes the `APPD_UPDATE_PROPS` property to update the database port and password.

(Optional) Add an `APPD_PREV` property in the update script to view the previous value of a property.

The update script is not saved and it applies to that particular update process only. You cannot use `APPD_PREV` as a property name or as a qualifier for a property name.

For example, the sample MySQL service update script includes the `APPD_PREV` property to view the value of the database password.

(Optional) Click the **Expand Cluster** button ( ) to expand the clustered node, select a custom task, and drag the task onto each node.

If you modify a service property or an application component property on a clustered node, the changes are applied to all of the nodes. You can configure the task properties in the Add Custom Task dialog box and save your changes.

Review the modified clustered node and update script in the execution plan and click **Next**.

The blue dotted lines in the execution plan define a specific order in which the deployment tasks will run.

Click the down arrow next to the configuration update script to view script details or the variable definitions used in the script.

Review the changed and impacted components in the update.

The modified properties and dependent properties are highlighted.

Click **Update** to deploy the modified configuration for the application.

The update process deploys the configured update to the cloud.

**What to do next**

Review the status of the deployment from the deployment summary page. See “Checking the Status of a Deployment,” on page 104.

Learn about the processes that take place in the background when the deployed application is updated in the cloud. See “Understanding the Deployment and Update Process,” on page 103.

**Troubleshoot Failed Update Process to Scale Deployments**

Known solutions and recommendations can help you when your update process to scale a deployment fails.

**Problem**

An update process to scale deployments has failed.

**Cause**

You might be attempting to initiate an update process to scale a deployment that previously failed to scale.
Solution

1. The cloud administrator must delete the new virtual machines for all the clustered nodes of the deployment.

2. Initiate another update process to scale a clustered node on the previously failed deployment.
   vFabric Application Director prepopulates the previously applied successful property values. The new values are taken from the previous failed deployment.
   
   For example, when you initiate an update process to scale an AppServer1 clustered node from 1 to 2 nodes and scale an AppServer2 clustered node from 1 to 3 nodes and the update process failed. In the second update process, you can scale any node with a value greater than the previous value.

3. Initiate another update process to scale all the failed nodes, so that the update scripts run on all of the dependent components such as load balancer, to make the components valid.

Troubleshoot Failed Update Process to Modify Configuration

You can apply some solutions when your update process to modify configuration fails.

Problem

An update process to modify configuration fails.

Cause

You might be attempting to initiate an update process to modify configuration on a previously failed update by resetting the failed property such as port number, and trying to proceed.

Solution

- Initiate another update process to modify configuration on the previously failed update.
  vFabric Application Director prepopulates the previously applied successful property values. The new values are taken from the previous failed deployment.

  You must update all of the failed properties when you initiate another update process to modify configuration.

- Create an update script to ignore a failed property.

  If you do not want to update all of the failed properties, you can author an update script to ignore the failed properties and allow you to proceed with the update process.

  The modified update script is not saved and it applies to only that particular update process.

Understanding the Deployment and Update Process

When you deploy an application to the cloud or update a deployed application in the cloud, several processes take place in the background. Virtual machines are created and software is provisioned in the virtual machines. It is important to understand the deployment and update process so that you can easily identify and troubleshoot any deployment failures.

The process of deploying an application to the cloud and updating a deployed application includes the following steps:

1. vFabric Application Director provisions the virtual machines by instantiating the cloud templates that were mapped in the deployment profile.
For the update process, vFabric Application Director provisions the scaled clustered virtual machines by instantiating the cloud templates from the vCloud Director catalog that were mapped in the deployment profile. The scaled clustered virtual machines use the same templates that were initially applied when deploying the application to the cloud. You cannot change the templates changed when you update the deployed application.

The rest of the update profile processes are the same as the deployment process.

2 vFabric Application Director requests the cloud service to establish network connections and receive IP addresses for all of the virtual machines in the deployment. After the IP addresses are assigned, the virtual machines restart to make sure the setup process is completed properly. The host names are derived from the application blueprint and assigned as the node names.

If host names are not assigned, then random host names are generated.

3 Bootstrap scripts included in each virtual machine download the agent from the vFabric Application Director server to the virtual machine. The agent is a jar file that runs in a Java virtual machine. Bootstrap scripts must be installed on the physical templates. See “Virtual Machine Requirements for Creating Custom Templates,” on page 39.

4 The bootstrap script starts the agent process.

5 The agent authenticates with the vFabric Application Director server.

6 The agent in each virtual machine downloads the execution plan from vFabric Application Director to the virtual machine.

7 The agent performs the installation and setup tasks for each component in the order specified in the deployment execution plan.

8 For each script, the agent waits for the dependent tasks to finish successfully, and downloads all of the content to the virtual machine directory at /tmp/runid/content/ComponentName/PropertyName and the task scripts to the directory at /tmp/runid/TaskName. The agent runs the tasks according to the parameter values sent by the server. When a task is complete, the agent informs the server about the status of the task.

If a script finishes with a nonzero exit status, the agent marks that task as failed. Otherwise, the agent marks the script as completed and proceeds to the next task. When a task fails, the entire deployment is stopped, marked as Failed Deployment, and no future tasks are executed. The reason for failure is available on the Details tab. When all of the tasks pass, the deployment is marked as Deployed Successfully.

**NOTE** For a script to run without interruptions, the return value must be set to zero. This value allows the agent to capture all of the computed properties and send them to the vFabric Application Director server.

For troubleshooting purposes, you can access the deployment details included in the execution plan. The task information is captured in log files for the install, configure, start, and update scripts used on each component of the deployment. These logs capture all of the information that is sent to the stdout and stderr log files.

### Checking the Status of a Deployment

vFabric Application Director provides a graphical user interface for checking the status of an application deployment in real time on the deployment summary page.

You can also use the vCloud Director user interface to check the status of the deployed virtual machines. To view the status of a vCloud Director deployment in an Amazon VPC and associated Availability Zone, see [Amazon AWS Documentation](https://aws.amazon.com/documentation).
On the deployment summary page, when a deployment is running, the overall deployment status of the deployment, update, or teardown process appears in the toolbar. After the deployment finishes, the status bar turns red or green depending on the success or failure of the tasks in the deployment. Above the task status windows, a task timeline contains a time stamp that shows when the application deployment was initiated, any subsequent update deployments, such as two scaled deployments or four updates to modify configuration, or if a teardown process was run.

**Figure 11-2. Deployment Summary Page**

To view the status of a task, a dependant task, or virtual machines acquiring IP addresses, click the **Expand** icon ( ) to open the status window in the summary page. Click the **Minimize** icon ( ) to view the entire deployment status summary.

After you deploy an update profile, you can check the status from the deployment summary page.

**Task Details Status Window**

The Task Summary section lists details about the deployment profile, user role, start and end time of the deployment, the last time the application was updated, run ID number of the subfolder containing logs, and the vApp name that is assigned to the application. You can also check the progress of a deployment in vCloud Director. Locate the vApp name and in vCloud Director, click the **My Cloud** tab and select **vApps** in the **My Cloud** tree on the left side of the screen. Search for the vApp name of the application you deployed in vFabric Application Director. To check the deployment status of an Amazon EC2 instance, see *Amazon AWS Documentation*. 
The Deployment Location section incorporates deployment environment, cloud provider name, cloud provider description, host IP address, vCloud Director Org name or Amazon VPC and associated Availability Zone, and User name details. The information shown in this section is a snapshot of the application blueprint and deployment profile at the time of the deployment. If you make changes to the actual application blueprint, those changes do not affect the blueprint shown in a specific deployment. The cloud provider and deployment information reflects the cloud provider mapping and deployment environment details that were created for the application deployment.

The Application Details section includes name and version information about the application blueprint.

Blueprint Status Window

The Blueprint is a snapshot that reflects the settings and definitions the blueprint contained at the time of the deployment. If you make changes to the actual application blueprint, those changes do not affect the blueprint shown in a specific deployment. A specific deployment can be deleted without affecting components or services in the application.

To view the property overrides in the blueprint and deployment profile, select a service or application component and click the Properties tab. For an update process, in the Properties tab, the Previous Value column shows the catalog, blueprint, or deployment profile values of a previous update process. On the same tab, the New Value column shows the values added in the current update process.

If a task fails on a service or application component, the task failure icon appears on the service or application component. When a task fails, the entire deployment is stopped, marked as Failed Deployment, and no future tasks are run. The task did not run icon appears on all the rest of the tasks.

In case of an update deployment, the updated node application components and properties are highlighted. If the update deployment fails, the application component is marked as failed and the impacted properties are highlighted in red.

VM Details Status Window

The VM Details shows status updates when virtual machines are created. IP addresses for each virtual machine appear when they are assigned. Virtual machine-related information, such as instance name, logical template, cloud template, host name in vCloud Director or Amazon EC2, memory allocation, number of CPUs, and network connection details are available.

After the IP address is acquired, the check mark under the Console column indicates that the virtual machine in the cloud is powered on. For vCloud Director, you can click the virtual machine to open the console in a Web browser that supports the VMRC plug-in and log in the virtual machine. For a deployed Web application, the IP address column displays the address you can use to access the application. For example, in a three-tiered application that includes a load balancer node, you might find the IP address of the load balancer and give that IP address to your end users. Agent bootstrap logs for each virtual machine is updated in the VM Log column. If the bootstrap process fails, the deployment is labeled Failed Deployment and the reason appears in the Task Details window. Click the ellipses button (…) in the columns to retrieve the virtual machine-specific cloud template and network information from vCloud Director.

Execution Plan Status Window

After the IP addresses are acquired and agents are bootstrapped, the components are deployed, installed, and configured. These processes occur according to the dependencies of the application shown by the arrows between the tasks in the execution plan. You can view the status of each task when the screen refreshes in approximately 30 seconds. Click Refresh in the upper-right corner of the page to update the status.

The following icons indicate the status for each component.

- Task has not yet begun or did not run ( ).
- Task is in progress ( ).
- Task is finished successfully (✅).
- Task requires reboot (⚠️).
- Task has failed (❌).
- Task is waiting for one of the dependencies to finish running (⏳).

When updating a deployed application, the execution plan displays only the clustered nodes that were modified with new values in the application and the update script that is generated from the defined update life cycle stage. Expand the clustered node to view the status on each node and the order in which the deployment tasks will run.

You can examine the log files for an action script after the task finishes running. To access the log files, click the gear icon in the upper-right corner of the task and select **View Virtual Machine Logs**. To view the property values for the script, click the gear icon in the upper-right corner of the task and select **View Component Properties**.
vFabric Application Director catalog includes predefined sample services, logical templates, application components, and tasks.

A sample application consists of either a Windows or Linux-based logical template, services, and application components configured specifically for that application. You must configure only the action scripts of the services in a predefined application and deploy it to the supported cloud environment. When you create a deployment profile to deploy a sample application, you can add a compatible customized task to the execution plan.

**IMPORTANT** Use the predefined sample templates, application components, services, and tasks only in a test environment. If you plan to use the sample templates in a production environment, make sure that you apply the latest security patches to the operating system in the template before deployment.

All user accounts that are assigned to the vFabric Application Director Default group can access and deploy the sample applications.

All of the icons rendered for the sample templates and services in the catalog are based on the name of the template or service. For example, the official MySQL icon appears next to the MySQL service in the catalog. If the name of the service or template is unique, then a generic icon appears.

Familiarize yourself with the key concepts that appear frequently in topics about deploying sample catalog components. See “Key Concepts,” on page 12.

This chapter includes the following topics:

- “Using the Sample Templates,” on page 109
- “Available Sample Applications for Deployment,” on page 110
- “Sample vFabric Application Director Catalog Tasks,” on page 122
- “Sample vFabric Application Director Catalog Services,” on page 128

**Using the Sample Templates**

vFabric Application Director provides the CentOS and Ubuntu sample templates with the operating system installed and all of the necessary libraries to deploy an application.

For example, to use the CentOS sample template, download the following files from the VMware product download site.

- CentOS 32-bit OVF
- CentOS 32-bit VMDK

For a list of the available sample templates, see the vFabric Application Director Readme file.
Upload the CentOS 5.6 32-bit OVF package to the vCloud Director catalog. For more information on the vCloud Director catalog, see the vCloud Director documentation.

**Note** For CentOS 32-bit sample templates, Physical Address Extension (PAE) is not enabled, so you can allocate up to 3.25GB of RAM for a virtual machine. PAE is enabled for Ubuntu 32-bit sample templates, so you can allocate more than 4GB of RAM for a virtual machine.

For the sample Ubuntu template to work properly, add the libpython XML package.

**Enable an SSH Connection for Ubuntu Virtual Machines**

Ubuntu virtual machines must have SSH host keys generated to work properly.

**Prerequisites**

Verify that you have the VMRC plug-in installed to work with your browser. For information about compatible Web browsers, see “vFabric Application Director System Requirements,” on page 18.

**Procedure**

1. Upload the Ubuntu OVF package to the vCloud Director catalog.
2. Click in the VMRC console.
3. Log in to the Ubuntu virtual machine.
4. Enable the SSH connection.
   ```
   ssh-keygen -t dsa -f /etc/ssh/ssh_host_dsa_key
   ssh-keygen -t rsa -f /etc/ssh/ssh_host_rsa_key
   
   service ssh restart
   ```
5. Restart the SSH server.

The SSH connection to the Ubuntu virtual machine is enabled.

**Available Sample Applications for Deployment**

You can use the predefined single-tier or three-tier applications to deploy either a simple Web application or a complex email suite. The application blueprint of a sample application includes logical templates, services, and application components that are fully functional when deployed to the cloud.

An application architect can use the vFabric Application Director sample applications as a base to model and create custom applications.

**Deploy Clustered WebSphere Application Server Application**

The Clustered WebSphere Application Server is a two-tier Linux-based application that provides automated installation and configuration for a clustered WebSphere service. The sample application deploys a Websphere admin server that manages a cluster of managed server nodes.

**Prerequisites**

- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 32bit 6GB Disk logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Verify that your user account has the ROLE_APP_ARCHITECT application architect role and ROLE_DEPLOYER deployer role assigned to it.
Log in to vFabric Application Director with a user account that belongs to the Default group.

Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.

Verify that the IBM Installation Manager file, `agent.installer.linux.gtk.x86_1.5.0.20110909_1200.zip`, is downloaded to an HTTPD server.

Set up an IBM download center account.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Select **Clustered Websphere Application Server**.
3. Click the **Blueprint** screenshot at the top of the screen.
4. In the blueprint editor, select the **Clustered_WAS-AppServer** service and set the values for the following properties.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>im_download_url</td>
<td>Add the URL to download the IBM Installation Manager file, <code>agent.installer.linux.gtk.x86_1.5.0.20110909_1200.zip</code>. The service uses this property to <code>wget</code> the IBM Installation Manager.</td>
</tr>
<tr>
<td>ibm_username</td>
<td>Set user name for the IBM download center.</td>
</tr>
<tr>
<td>ibm_password</td>
<td>Designate a password for the IBM download center.</td>
</tr>
<tr>
<td>admin_password</td>
<td>Set an admin password for the WebSphere Application Server.</td>
</tr>
<tr>
<td>proxy_host</td>
<td>If the vFabric Application Director server is configured to use a proxy, configure the proxy. For example, <code>proxy.setting.com</code>.</td>
</tr>
<tr>
<td>proxy_port</td>
<td>If the vFabric Application Director server is configured to use a proxy, set the proxy port.</td>
</tr>
<tr>
<td>proxy_username</td>
<td>If the vFabric Application Director server is configured to use a proxy, add the proxy user name.</td>
</tr>
<tr>
<td>proxy_password</td>
<td>If the vFabric Application Director server is configured to use a proxy, designate the proxy password.</td>
</tr>
</tbody>
</table>

5. In the blueprint editor, select the **Clustered_WAS-Dvlp_Manager** service and set the values for the following properties.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>im_download_url</td>
<td>Add the URL to download the IBM Installation Manager file, <code>agent.installer.linux.gtk.x86_1.5.0.20110909_1200.zip</code>. The service uses this property to <code>wget</code> the IBM Installation Manager.</td>
</tr>
<tr>
<td>ibm_username</td>
<td>Set user name for the IBM download center.</td>
</tr>
<tr>
<td>ibm_password</td>
<td>Designate a password for the IBM download center.</td>
</tr>
<tr>
<td>admin_password</td>
<td>Set an admin password for the WebSphere Application Server.</td>
</tr>
<tr>
<td>proxy_host</td>
<td>If the vFabric Application Director server is configured to use a proxy, configure the proxy. For example, <code>proxy.setting.com</code>.</td>
</tr>
<tr>
<td>proxy_port</td>
<td>If the vFabric Application Director server is configured to use a proxy, set the proxy port.</td>
</tr>
<tr>
<td>proxy_username</td>
<td>If the vFabric Application Director server is configured to use a proxy, add the proxy user name.</td>
</tr>
<tr>
<td>proxy_password</td>
<td>If the vFabric Application Director server is configured to use a proxy, designate the proxy password.</td>
</tr>
</tbody>
</table>

6. Create a deployment profile and deploy the application.
To access the deployed application, add the Websphere Application Server virtual machine IP address in a supported Web browser at https://DMGR_Manager_Node_IP:50001/ibm/console/logon.jsp.

Use the log in wasadmin and password that was set for the admin_password property in the Clustered_WAS-AppServer service to access the administration console.

Deploy Nanotrader Application

The sample Nanotrader application is a three-tier Web application that deploys the NanoTrader application using the vFabric Web Server, vFabric tc Server, vFabric RabbitMQ, and vFabric SQLFire components.

Prerequisites

- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 32bit 6GB Disk logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Verify that your user account has the ROLE_APP_ARCHITECT application architect role and ROLE_DEPLOYER deployer role assigned to it.
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.
- Verify that the spring-nanotrader-asynch-services.war, spring-nanotrader-data.war, spring-nanotrader-web.tgz, and nanotrader.tgz files are downloaded to an HTTPD server.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Select Nanotrader.
3. Click the Blueprint screenshot at the top of the screen.
4. In the blueprint editor, select the vFabric_Web_Server service and add a URL value to download the spring-nanotrader-web.tgz file for the deployment_archive property.
5. Select the vFabric_tc_Server service and add a URL value to download the nanotrader.tgz file for the external_template property.
6. Select the spring-nanotrader-services SCRIPT component and add a URL value to download the spring-nanotrader-data.war file for the war_file property.
7. Select the spring-nanotrader-asynch-services SCRIPT component and add a URL value to download the spring-nanotrader-asynch-services.war file for the war_file property.
8. Create a deployment profile and deploy the application.
9. To access the deployed application, open a supported Web browser and type the http://LoadBalancer_IP/#login URL.
   The LoadBalancer_IP is the IP address of the deployed load balancer.
10. Create a NanoTrader account to set a user name and password.
Deploy WebLogic Clustered Deployment Application

The sample WebLogic Clustered Deployment application has a multitier clustered configuration with an Apache load balancer running Oracle WebLogic Server Web Server Plugins 11g. The Apache load balancer also acts as the application server for the Clustered Dukes Bank and showcases the features of the application.

Prerequisites

- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 32bit 6GB Disk logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Verify that your user account has the ROLE_APP_ARCHITECT application architect role and ROLE_DEPLOYER deployer role assigned to it.
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.
- Verify that the wls1211_generic.jar installer file is downloaded to an HTTPD server.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Select Weblogic Clustered Deployment.
3. Click the Blueprint screenshot at the top of the screen.
4. In the blueprint editor, select the ASWebLogic service and add a URL value to download the generic wls1211_generic.jar installer file for the weblogic_installer property.
5. Select the MSWebLogic service and add a URL value to download the generic wls1211_generic.jar installer file for the weblogic_installer property.
6. Select the SetupWLClusterLB SCRIPT component and add a URL value to download the Oracle WebLogic Server Web Server Plugins 11g for the mod_wl_so property. You can download the plug-in from the Oracle Web site.
7. Create a deployment profile and deploy the application.
8. To access the Admin console of the deployed WebLogic application, open a supported Web browser and type the http://AdminServer_IP:7001/console URL.
9. To access the deployed Clustered Dukes Bank application, open a supported Web browser and type the http://Apache_Server_IP URL.
   The Apache_Server_IP is the IP address of the deployed Apache server.
10. Use the log in 200 and password foobar123 to access the application.
Deploy Clustered DotShoppingCart Application

The Clustered DotShoppingCart is a sample three-tier Windows-based application that uses Microsoft SQL Server as the database server, Microsoft IIS Server as the application server, and Apache HTTP Server as the load balancer.

The Create_DB script creates and initializes the DotShoppingCart database in the SQL Server database. The DotShoppingCart_App script installs the DotShoppingCart application in the Microsoft IIS server and uses the DotShoppingCart database. The Microsoft IIS AppServer node is defined as a cluster that allows scaling out to multiple nodes for handling a larger load. The Apache HTTP server handles the load balancing.

**NOTE** Use the predefined sample application only in a test environment.

**Prerequisites**
- Register the Windows Server 2008 R2 SP1 vApp template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the Windows Server 2008 R2 SP1 logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Verify that your user account has the **ROLE_APP_ARCHITECT** application architect role and **ROLE_DEPLOYER** deployer role assigned to it.
- Verify that the DotShoppingCart 3.0 (Open Source Edition) package **OpenSourceEdition.V3.zip** file is downloaded. The Web server must be configured to transfer the file using the HTTP protocol. Identify the URL used to access the file in a Web browser.
- Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.

**Procedure**
1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Select **Clustered DotShoppingCart**.
3. Click the Blueprint screenshot at the top of the screen.
4. In the blueprint editor, select DotShoppingCart_App application component and type a valid URL for the DSC_ZIP property value.
   
   During deployment, the vFabric Application Director agent in the AppServer Windows virtual machine must be able to access the valid URL for the OpenSourceEdition.V3.zip file.
5. (Optional) If you plan to have more than one AppServer node, select **AppServer** and increase the cluster size.
6. Create a deployment profile and deploy the application.
7. To access the deployed application, open a supported Web browser and type the http://Load_Balancer_IP:8081 URL.
   
   The Load_Balancer_IP is the IP address of the deployed load balancer.
Deploy Clustered Dukes Bank Application

The Clustered Dukes Bank application is a sample three-tier vFabric Application Director application that uses MySQL or Microsoft SQL Server as its database, JBoss Server as its application server, and Apache HTTP server as its load balancer that you can deploy to your test environment.

The initialize_db_script script creates and initializes the dukes_db database in the MySQL or SQL Server. The Dukes_Bank_App EAR application component uses the dukes_db database, which is deployed in the JBoss server. The JBoss appserver node is defined as a cluster that allows scaling out to multiple nodes to handle a higher load. The Apache HTTP server handles the load balancing.

**NOTE** Use the predefined sample application only in a test environment.

Prerequisites

- Register the uploaded CentOS 5.6 32-bit Linux template or Windows Server 2008 R2 SP1 vApp template to a cloud provider. See “Using the Sample Templates,” on page 109 and “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 32bit or Windows Server 2008 R2 SP1 logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the ROLE_APP_ARCHITECT application architect role and ROLE_DEPLOYER deployer role assigned to it.
- For the Linux-based Clustered Dukes Bank, if your system requires a proxy to access the Internet, verify that your proxy connections are properly configured. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- For the Windows-based Clustered Dukes Bank, verify that the Windows virtual machine has a valid JRE installation path that can be used in the template.
- Understand the basic concepts of creating a deployment profile, deploying an application, and updating a deployed application. See Chapter 11, “Deploying Applications,” on page 91.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click **Clustered Dukes Bank App**.
3. Select the application version.
   - For the Linux-based Clustered Dukes Bank, select the application version 2.1.0.
   - For the Windows-based Clustered Dukes Bank, select the application version 3.0.0.
4. Click the **Blueprint** screenshot at the top of the screen.
5. In the blueprint editor for the Windows-based Clustered Dukes Bank, select the JBossAppServer service and type the appropriate value for the JAVA_INSTALL_DIR property.
   During deployment, vFabric Application Director agent in the JBossAppServer service should be able to access the installed JRE.
6. For the Linux-based Clustered Dukes Bank, if the cloud template is in a private network without a proxy, when you deploy the application, add a YUM configuration task to each virtual machine in the deployment profile, and set the repository_url property to use a CentOS 5.6 32-bit repository hosted in the private network.
7. Create a deployment profile and deploy the application.
To access the deployed application, open a supported Web browser and type the http://Load_Balancer_IP:8081/bank/main.faces URL. The Load_Balancer_IP is the IP address of the deployed load balancer.

Use the log in **200** and password **foobar** to access the application.

(Optional) Update the configuration of a deployed Linux-based Clustered Dukes Bank application.

You can modify several component properties in the deployed application.

<table>
<thead>
<tr>
<th>Component</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td>http_port</td>
</tr>
<tr>
<td></td>
<td>http_node_port</td>
</tr>
<tr>
<td></td>
<td>http_proxy_port</td>
</tr>
<tr>
<td>JBOSS for Linux</td>
<td>JBOSS_JMX_USER</td>
</tr>
<tr>
<td></td>
<td>JBOSS_JMX_PWD</td>
</tr>
<tr>
<td>MySQL</td>
<td>db_port</td>
</tr>
<tr>
<td></td>
<td>db_root_username</td>
</tr>
<tr>
<td>initialize_db_script</td>
<td>db_username</td>
</tr>
<tr>
<td></td>
<td>db_password</td>
</tr>
<tr>
<td>Dukes_Bank_App</td>
<td>JAR_FILE</td>
</tr>
<tr>
<td></td>
<td>EAR_FILE</td>
</tr>
<tr>
<td></td>
<td>db_port</td>
</tr>
<tr>
<td></td>
<td>db_password</td>
</tr>
<tr>
<td></td>
<td>db_user</td>
</tr>
</tbody>
</table>

**Deploy Apache Hadoop Cluster**

Apache Hadoop cluster is a sample application that runs on Hadoop. Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using a programming model.

The software is designed to scale up from single servers to a large number of machines, each offering local computation and storage capacity. Apache Hadoop detects and handles failures at the application layer and delivers a highly available service on a cluster of computers that might be prone to failures.

**NOTE** Use the predefined sample application only in a test environment.

**Prerequisites**

- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 32bit 6GB Disk logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the ROLE_APP_ARCHITECT application architect role and ROLE_DEPLOYER deployer role assigned to it.
- If your system requires a proxy to access the Internet, verify that your proxy connections are properly configured. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.
Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Click Apache Hadoop Cluster.
3. Click the Blueprint screenshot at the top of the screen.
4. On each node, set a Hadoop user password for the Hadoop_NameNode, Hadoop_DataNode, Hadoop_JobTracker, and Hadoop_TaskTracker services.
5. Create a deployment profile and deploy the application.
6. To access the application running on Hadoop, open a supported Web browser and type the JT virtual machine IP address `http://JT_VM_IP:8080/HadoopDemoApp/Login.jsp` URL.
7. Use the log in admin and password admin to access the application.

Deploy Microsoft SharePoint Server 2010

The single-tier sample application deploys a SharePoint server in a Windows 2008 R2 SP1 64-bit virtual machine. With SharePoint 2010 you can set up Web sites to share information with others, manage documents from start to finish, and publish reports.

**Note** Use the predefined sample application only in a test environment.

Prerequisites

- Register the Windows Server 2008 R2 SP1 vApp template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the Windows Server 2008 R2 SP1 logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Verify that your user account has the ROLE_APP_ARCHITECT application architect role and ROLE_DEPLOYER deployer role assigned to it.
- Verify that the SharePointServer.exe installer file is downloaded and hosted on a Web server configured to transfer the installer file using the HTTP protocol. Identify the URL used to access the file in a Web browser.
- Verify that the Microsoft SharePoint Server 2010 volume license key is readily available.
- Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Click Microsoft Sharepoint Server 2010.
3. Click the Blueprint screenshot at the top of the screen.
4. Select the Sharepoint_2010 service and set the download_url and pid_key property values.
5. Create a deployment profile and deploy the application.
6. To access the SharePoint application, open a supported Internet Explorer Web browser and type the `http://VM_IP:7001/` URL.
7. Use a Windows administrator user account to log in to the application.
**Deploy jPetStore Application**

vFabric Application Director includes two versions of the jPetStore sample application that you can deploy. jPetStore 1.0.1 is a three-tier application that deploys jPetStore with vFabric SQLFire Server and Locator as the database layer, vFabric tc Server as the application server, and vFabric Web Server as the load balancer. jPetStore 1.0.0 is a single-tier sample Web store application that deploys the jPetStore App WAR file on vFabric tc Server using SQLFire as a database that you can deploy to your test environment.

*NOTE* Use the predefined sample application only in a test environment.

**Prerequisites**

- Register the uploaded CentOS 5.6 64-bit template for jPetStore 1.0.1 or CentOS 5.6 32-bit template for jPetStore 1.0.0 to a cloud provider. See “Using the Sample Templates,” on page 109 and “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 64bit disk logical template for jPetStore 1.0.1 or CentOS56 32bit 6GB disk logical template 1.0.0 for jPetStore 1.0.0. See “Add a Logical Template to the Catalog,” on page 69.
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
- If your system requires a proxy to access the Internet, verify that your proxy connections are properly configured. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click the **jPetStore** version.
   
   All the services and application components for this application are predefined and do not need additional configuration.
3. Click the **Blueprint** screenshot at the top of the screen.
4. Create a deployment profile and deploy the application.
5. To access the deployed application, complete the task depending on the application version.
   
   - For jPetStore 1.0.1, open a supported Web browser and type the http://vFabric_Web_Server_IP/jpetstore-1.0.1 URL.
   - For jPetStore 1.0.0, open a supported Web browser and type the http://jPetStore_VM_IP:8080/jpetstore-1.0.0 URL.

**Deploy Spring Travel Application**

vFabric Application Director includes two versions of the Spring Travel sample application that you can deploy.

Spring Travel 3.0.0 is a multitier Web application that deploys Spring Travel App with vFabric tc Server as the application server and vFabric Web Server as the load balancer. Spring Travel 1.0.0 is a single-tier sample Web application that deploys the Spring Travel App WAR file on tc Server.

*NOTE* Use the predefined sample application only in a test environment.
Prerequisites

- Register the uploaded CentOS 5.6 64-bit template for Spring Travel 3.0.0 or CentOS 5.6 32-bit template for Spring Travel 1.0.0 to a cloud provider. See “Using the Sample Templates,” on page 109 and “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 64bit disk logical template for Spring Travel 3.0.0 or CentOS56 32bit 6GB disk logical template 1.0.0 for Spring Travel 1.0.0. See “Add a Logical Template to the Catalog,” on page 69.
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.
- If your system requires a proxy to access the Internet, verify that your proxy connections are properly configured. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Click the Spring Travel version.
   All the services and application components for this application are predefined and do not need additional configuration.
3. Click the Blueprint screenshot at the top of the screen.
4. Create a deployment profile and deploy the application.
5. To access the deployed application, complete the task depending on the application version.
   - For Spring Travel 3.0.0, open a supported Web browser and type the http://LoadBalancer_IP/swf-booking-mvc-2.0.3.RELEASE URL.
   - For Spring Travel 1.0.0, open a supported Web browser and type the http://SpringTravel_VM_IP:8080/swf-booking-mvc-2.0.3.RELEASE URL.

Deploy Zimbra Application

Zimbra is an email and collaboration suite. The Zimbra App is divided into the LDAP server node, the Mailbox (mbox) node, and the Mail Transport Agent (MTA) node.

**Note** Use the predefined sample application only in a test environment.

**Prerequisites**

- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 32bit 6GB Disk logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.
- Verify that you have a software and license key available.
  The license file is available on the VMware Zimbra Web site. You must host the file on an HTTP server.
- Verify that vFabric Application Director is configured to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- Verify that the Zimbra virtual machine IP addresses have an associated DNS record that supports forward and reverse IP address lookup.

- Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click **Zimbra App**.
3. Click the **Blueprint** screenshot at the top of the screen.
4. Set up an NFS server.
5. Download the Zimbra 7.1.1 Network Edition installation file for Red Hat Enterprise Linux 6. You can download the file from the VMware Zimbra Web site **Downloads** tab.
6. Untar the installation files in the NFS server you created.
7. Select each Zimbra service in the blueprint and configure the property value.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zimbra_nfs_path</td>
<td>Add the NFS server path. For example, nfsserver.com:/path/to/export.</td>
</tr>
<tr>
<td>zimbra_installer_root_relpath</td>
<td>Set to the directory that contains the untarred installation files.</td>
</tr>
<tr>
<td>zimbra_licencefile_url</td>
<td>Add the URL of the hosted license file.</td>
</tr>
<tr>
<td>ldap_ip</td>
<td>From the drop-down menu, set the property to bind to ldap:ip.</td>
</tr>
<tr>
<td>mbox_ip</td>
<td>From the drop-down menu, set the property to bind to mbox:ip.</td>
</tr>
<tr>
<td>mta_ip</td>
<td>From the drop-down menu, set the property to bind to mta:ip.</td>
</tr>
<tr>
<td>self_ip</td>
<td>From the drop-down menu, set the property to bind to self:ip.</td>
</tr>
</tbody>
</table>

For example, when you configure Zimbra MTA service, the property value for mta_ip is self:ip.

8. Create a deployment profile and deploy the application.

9. To access the application, add the IP address of the deployed mbox virtual machine in a supported Web browser and log in using the username **admin** and default password **VirtualMachine**.

You can change the default password value in the ADMIN_PWD property for all virtual machines in a blueprint.

**Deploy the Zimbra Clustered Application**

Zimbra is an email and collaboration suite. The Zimbra Clustered App is divided into the LDAP server node, the primary mailbox (mbox) node, the clustered subordinate mailbox (mbox) nodes, the Mail Transport Agent (MTA) node, and the proxy node.

**Note** Use the predefined sample application only in a test environment.

**Prerequisites**

- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
- Map the cloud template to the CentOS56 32bit 6GB Disk logical template. See “Add a Logical Template to the Catalog,” on page 69.
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
Verify that you have a software and license key available.
The license file is available on the VMware Zimbra Web site. You must host the file on an HTTP server.

Verify that vFabric Application Director is configured to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.

Verify that the Zimbra virtual machine IP addresses have an associated DNS record that supports forward and reverse IP address lookup.

Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.

Procedure
1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Click Clustered Zimbra App.
3. Click the Blueprint screenshot at the top of the screen.
4. Set up an NFS server.
   You can download the file from the VMware Zimbra Web site Downloads tab.
6. Untar the installation files in the NFS server you created.
7. Select each Zimbra service in the blueprint and configure the property value.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zimbra_nfs_path</td>
<td>Add the NFS server path. For example, nfsserver.com:/path/to/export.</td>
</tr>
<tr>
<td>zimbra_installer_root_relpath</td>
<td>Set to the directory that contains the untarred installation files.</td>
</tr>
<tr>
<td>zimbra_licencefile_url</td>
<td>Add the URL of the hosted license file.</td>
</tr>
<tr>
<td>ldap_ip</td>
<td>From the drop-down menu, set the property to bind to ldap:ip.</td>
</tr>
<tr>
<td>mbox_ip</td>
<td>From the drop-down menu, set the property to bind to mbox:ip.</td>
</tr>
<tr>
<td>mta_ip</td>
<td>From the drop-down menu, set the property to bind to mta:ip.</td>
</tr>
<tr>
<td>self_ip</td>
<td>From the drop-down menu, set the property to bind to self:ip.</td>
</tr>
<tr>
<td>proxy_ip</td>
<td>From the drop-down menu, set the property to bind to proxy:ip.</td>
</tr>
</tbody>
</table>

For example, when you configure Zimbra MTA service, the property value for mta_ip is self:ip.
8. Create a deployment profile and deploy the application.
9. To access the application, add the IP address of the deployed proxy virtual machine in a supported Web browser and log in using username admin and default password VirtualMachine.
   You can change the default password value in the ADMIN_PWD property for all virtual machines in a blueprint.

Deploy Radiant CMS Application

Radiant CMS is a single-node application that deploys a Ruby On Rails service and a MySQL database, and installs the Radiant Application configured to use the MySQL database.

**NOTE** Use the predefined sample application only in a test environment.

Prerequisites
- Register the uploaded CentOS 5.6 32-bit template to a cloud provider. See “Register the vCloud Director Cloud Provider and Template,” on page 52.
Map the cloud template to the CentOS56 32bit 6GB Disk logical template. See “Add a Logical Template to the Catalog,” on page 69.

Log in to vFabric Application Director with a user account that belongs to the Default group.

Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.

Verify that vFabric Application Director is configured to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.

Understand the basic concepts of creating a deployment profile and deploying an application. See Chapter 11, “Deploying Applications,” on page 91.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select Applications.
2. Click Radiant CMS.
   
   All the services and application components for this application are predefined and do not need additional configuration.
3. Click the Blueprint screenshot at the top of the screen.
4. Create a deployment profile and deploy the application.
5. To access the application, add the Radiant CMS virtual machine IP address http://VM_IP/ in a supported Web browser and log in using the Radiant default admin username and Radiant password.

**Sample vFabric Application Director Catalog Tasks**

You can add predefined tasks in an execution plan during deployment. These tasks can be required or optional during the application deployment process.

For example, you must add and configure the RHN Repository predefined task in the execution plan to install or update the YUM repositories, for a Red Hat Linux-based application to run properly.

You can also create a custom task, add it to the predefined application, and deploy it to the cloud. For the custom task to be available in the Tasks page, you must add it to the catalog. See “Add a Custom Task to the Catalog,” on page 77.

**Add Join Domain Predefined Task**

The Join Domain predefined task allows a Windows virtual machine to join an Active Directory domain as part of the deployment process. The predefined task starts the Windows Domain Manager Netdom.exe utility, to automate this operation.

With the predefined task, you do not need to manually configure a Windows virtual machine template with static domain settings and you can customize the task. The predefined task requires an extra restart cycle to complete the process of joining an Active Directory domain.

**Prerequisites**

- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.
- Verify that vFabric Application Director is configured to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.
- For multiple deployments, manually enable the Change SID and Domain Join options for a Windows virtual machine template before you deploy an application. See “Enable SID Change and Domain Join for Windows Virtual Machine Templates,” on page 43.
Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click the name of a Windows-based application.
   
   A list of application versions appear.
3. Select an application version.
4. Create a deployment profile.
5. In the Deployment Profile wizard, follow the prompts until you get to **Execution Plan**.
6. Click the **Expand Cluster** button ( ), if the node is clustered.
   
   If the clustered node is not expanded, the predefined task is added only to the first virtual machine in the cluster.
7. Click the **Add Script Task** button ( ) and drag a predefined task in the blueprint.
   
   When you drag a predefined task, you see anchors (§) that indicate where you can drop the predefined task.
   
   After you drag and drop a predefined task to a node, the Add Custom Task dialog box opens.
8. Select the predefined task from the **Catalog Task Name** drop-down menu.
   
   The supported operating systems, predefined task details, script, and property details appear in the dialog box.
9. On the **Properties** tab, configure the properties.
   
   a. Select the domain_name property, type a new name for the Windows domain in the Edit Property dialog box, and click **Save**.
   
   b. Select the domain_user property, specify the name of the domain user who can join the Active Directory in the Edit Property dialog box, and click **Save**.
   
   c. Select the domain_password property, type the domain user password in the Edit Property dialog box, and click **Save**.
   
   d. (Optional) Select the apply_ou property, change the new value to **Yes** in the Edit Property dialog box to allow a specific organization unit in the Active Directory domain to join a Windows virtual machine, and click **Save**.
   
   e. (Optional) Select the domain_ou property, specify the organization unit in the Active Directory domain in the Edit Property dialog box, and click **Save**.
10. Click **OK** to close the Edit Property dialog box.

   The Join Domain predefined task is added to the execution plan.
11. Review the deployment profile settings and deploy the application.

What to do next

Explore whether to add a customized task to the vFabric Application Director catalog. See “Add a Custom Task to the Catalog,” on page 77.
Add APT Repository Config Predefined Task

The APT Repository Config predefined task is a script used to update the APT repositories to install or update software on Ubuntu or other DEB-based operating systems.

You can configure the APT Repository Config properties to add a new repository or remove all of the existing repositories. If you need more than one repository, you can create multiple tasks and link them in the execution plan by adding one task next to the other.

**Note** Use the predefined sample task only in a test environment.

**Prerequisites**
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
- Verify that the predefined task is added to the execution deployment plan before you add any services or application components that require APT for installing or updating software packages.
- Verify that vFabric Application Director is configured to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click the name of the application.
   
   A list of application versions appear.
3. Select an application version.
4. Create a deployment profile.
5. In the Deployment Profile wizard, follow the prompts until you get to **Execution Plan**.
6. Click the **Expand Cluster** button ( ), if the node is clustered.
   
   If the clustered node is not expanded, the predefined task is added only to the first virtual machine in the cluster.
7. Click the **Add Script Task** button ( ) and drag a predefined task in the blueprint.
   
   When you drag a predefined task, you see anchors ( # ) that indicate where you can drop the predefined task.
   
   After you drag and drop a predefined task to a node, the Add Custom Task dialog box opens.
8. Select the predefined task from the **Catalog Task Name** drop-down menu.
   
   The supported operating systems, predefined task details, script, and property details appear in the dialog box.
9 On the **Properties** tab, configure the properties.
   a Select the repository_name property, type a new unique value identifying the repository in the Edit Property dialog box, and click **Save**.
   b Select the source_str property, type `http://site.example.com/debian distribution component1 component2 ...` in the **URL value** text box of the Edit Property dialog box, and click **Save**.
   A sample Ubuntu URL is `deb http://us.archive.ubuntu.com/ubuntu/ lucid main`.
   c Select the remove_all_repos property and define an appropriate value in the Edit Property dialog box.
   Set the value to **true** to remove all other repositories before you add the new configuration. You can also accept the default **false** value to add a new repository.

10 Click **Save** and **OK** to close the Edit Property dialog box.
   The APT Repository Config predefined task is added to the execution plan.

11 Review the deployment profile settings and deploy the application.

**What to do next**
Determine whether to add a customized task to the vFabric Application Director catalog. See “Add a Custom Task to the Catalog,” on page 77.

**Add a YUM Repository Config Predefined Task**

The YUM Repository Config predefined task is a script used for updating the YUM repositories to install or update software on CentOS or other RPM-based operating systems.

You can configure the YUM Repository Config properties to add a new repository or remove all of the existing repositories. If you need more than one repository, you can create multiple tasks and link them in the execution plan by adding one task next to the other.

**NOTE** Use the predefined sample task only in a test environment.

**Prerequisites**
- Log in to vFabric Application Director with a user account that belongs to the Default group.
- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
- Verify that the predefined task is added to the execution deployment plan before you add any services or application components that require YUM for installing or updating software packages.
- Verify that vFabric Application Director is configured to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.

**Procedure**
1 On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2 Click the name of the application.
   A list of application versions appear.
3 Select an application version.
4 Create a deployment profile.
5 In the Deployment Profile wizard, follow the prompts until you get to **Execution Plan**.
6 Click the **Expand Cluster** button ( ), if the node is clustered.

If the clustered node is not expanded, the predefined task is added only to the first virtual machine in the cluster.

7 Click the **Add Script Task** button ( ) and drag a predefined task in the blueprint.

When you drag a predefined task, you see anchors ( ) that indicate where you can drop the predefined task.

After you drag and drop a predefined task to a node, the Add Custom Task dialog box opens.

8 Select the predefined task from the **Catalog Task Name** drop-down menu.

The supported operating systems, predefined task details, script, and property details appear in the dialog box.

9 On the **Properties** tab, configure the properties.

   a Select the repository_name property, type a new unique value identifying the repository in the Edit Property dialog box, and click **Save**.

   b Select the repository_url property, type a URL value in the Edit Property dialog box, and click **Save**.

      An example of a CentOS 5.6 32-bit URL is http://vault.centos.org/5.6/os/i386/.

   c Select the remove_all_repos property and define an appropriate value in the Edit Property dialog box.

      Set the value to **true** to remove all other repositories before adding the new configuration. You can also accept the default **false** value to add a new repository.

10 Click **Save** and **OK** to close the Edit Property dialog box.

    - Set the value to **true** to remove all other repositories before adding the new configuration.

    - Accept the default **false** value to add a new repository and click **Save**.

The YUM Repository Config predefined task is added to the execution plan.

11 Review the deployment profile settings and deploy the application.

**What to do next**

Consider adding a customized task to the vFabric Application Director catalog. See “Add a Custom Task to the Catalog,” on page 77.

**Add a RHN Registration Predefined Task**

The RHN Registration predefined task is a script used to update the YUM repositories to install Red Hat Enterprise Linux or other Red Hat operating systems that have the Red Hat Network with YUM.

The predefined task registers the virtual machine with the Red Hat Network using the credentials provided with a machine name VMware_AppDirector_$_RANDOM, where $RANDOM is a short string that makes the virtual machine registration unique.

**NOTE** Use the predefined sample task only in a test environment.

**Prerequisites**

- Log in to vFabric Application Director with a user account that belongs to the Default group.

- Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.
- Verify that the predefined task is added to the execution deployment plan before you add any services or application components that require YUM for installing or updating software packages.

- Verify that vFabric Application Director is configured to use a proxy. See “Configure vFabric Application Director to Use a Proxy for External URLs,” on page 25.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Applications**.
2. Click the name of the application.
   
   A list of application versions appear.
3. Select an application version.
4. Create a deployment profile.
5. In the Deployment Profile wizard, follow the prompts until you get to **Execution Plan**.
6. Click the **Expand Cluster** button ( ), if the node is clustered.
   
   If the clustered node is not expanded, the predefined task is added only to the first virtual machine in the cluster.
7. Click the **Add Script Task** button ( ) and drag a predefined task in the blueprint.
   
   When you drag a predefined task, you see anchors ( # ) that indicate where you can drop the predefined task.
   
   After you drag and drop a predefined task to a node, the Add Custom Task dialog box opens.
8. Select the predefined task from the **Catalog Task Name** drop-down menu.
   
   The supported operating systems, predefined task details, script, and property details appear in the dialog box.
9. On the **Properties** tab, configure the properties.
   
   a. Select the rhn_username property, type the username value used to register the virtual machine with the Red Hat Network, and click **Save**.
   b. Select the rhn_password property, type the password value used to register the virtual machine with the Red Hat Network, and click **Save**.
10. Click **OK** to close the Edit Property dialog box.
   
    The RHN Registration predefined task is added to the execution plan.
11. Review the deployment profile settings and deploy the application.

**What to do next**

If you have a customized task, you can add it to the vFabric Application Director catalog. See “Add a Custom Task to the Catalog,” on page 77.
Sample vFabric Application Director Catalog Services

vFabric Application Director includes predefined components, such as services, in its catalog that are reusable components in several applications. These services are available to all user groups in vFabric Application Director. Logical templates must be added for each group outside the Default group.

Catalog Services

On the vFabric Application Director title bar, click the drop-down menu and select Catalog > Services to view the available sample services. The Catalog menu also includes standard logical templates, task scripts, operating systems, and tags.

An application architect can create an application blueprint and add the sample services to the applicable nodes and configure them. The sample services can also be configured when deploying a predefined application.

In the application blueprint, these sample services are grouped into Application Servers, Database Servers, Web Servers, Windows Services, Zimbra Services, Monitoring, Performance Management, Other, and Windows.

For recommended property configurations, supported operation systems, and application components information, see the VMware vFabric Application Director Catalog Services guide.

Note: Use the predefined sample catalog services only in a test environment.
Managing Deployments

When you deploy an application, an item is added to the Deployments page in vFabric Application Director. You can use this page to view the following items:

- List of all the deployments
- Deployment details of an individual deployment
- A scaled update process or configured update process

You can also teardown a deployed application from the cloud and remove an application deployment record from vFabric Application Director.

This chapter includes the following topics:

- “View Deployment Details for an Application,” on page 129
- “Teardown an Application from the Cloud,” on page 131
- “Delete an Application Deployment from vFabric Application Director,” on page 131
- “Cancel a Deployment or an Update Process,” on page 132
- “Collect Logs to Troubleshoot Deployment Failures,” on page 133
- “Troubleshooting Common Errors During Deployment,” on page 134
- “Troubleshooting Common Errors During an Update Process,” on page 139

View Deployment Details for an Application

You can view details about the success or failure of a particular application deployment or an update process. You can view such details as the IP addresses that were assigned, the cloud networks chosen, and the logs for each installation, configuration, and startup or update scripts that were run.

All of the users roles can view deployments that are associated with their user's group.

The deployment summary page shows the overall status of the deployment. Each deployment listed on the summary page has a snapshot of the application blueprint and deployment profile at the time the deployment was started. If you make changes to the actual application blueprint or deployment profile, those changes do not affect the blueprint or deployment shown in a specific deployment. See “Checking the Status of a Deployment,” on page 104.

Prerequisites

Verify that you installed the supported VMRC plug-in and Web browser. See “vFabric Application Director System Requirements,” on page 18.
Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Deployments.

2. Click the name of the deployment.

The deployment summary page appears with four status windows. The VM Details status window provides quick access to each virtual machine, and lets you log in to the virtual machine deployed on vCloud Director.

3. Use the deployment summary page to find application deployment information.

<table>
<thead>
<tr>
<th>Type of Information</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reason for deployment failure</td>
<td>Look at the status bar at the top of the deployment summary page for the overall deployment status.</td>
</tr>
<tr>
<td></td>
<td>The deployment details are in the Task Details status window. For action script failure, expand the Execution Plan status window and review the individual tasks.</td>
</tr>
<tr>
<td>Start time, end time, and last updated</td>
<td>For the overall deployment status, look at the Task Details status window. For specific task details, expand the Execution Plan window.</td>
</tr>
<tr>
<td>time</td>
<td></td>
</tr>
<tr>
<td>Order in which components were created</td>
<td>Look at the Execution Plan status window.</td>
</tr>
<tr>
<td>vApp Name of the application</td>
<td>Look at the Task Details status window. You need the vApp or instance name to find information about the deployment in vCloud Director or Amazon EC2.</td>
</tr>
<tr>
<td>IP Addresses of the virtual machines</td>
<td>Look at the VM Details window.</td>
</tr>
<tr>
<td>Logs</td>
<td>Virtual machine-specific agent bootstrap logs are in the table of virtual machines of the VM Details window. Logs for action scripts are in the Execution Plan window.</td>
</tr>
<tr>
<td>Details about a component or action</td>
<td>Expand the Execution Plan status window and click the View Task Information button ( ) next to the component. The details listed there are the settings and definitions that the blueprint contained at the time of deployment.</td>
</tr>
<tr>
<td>script and its properties</td>
<td></td>
</tr>
<tr>
<td>Details about a deployed virtual</td>
<td>Look at the table of virtual machines in the VM Details status window. This information includes disk size. Expand the window and click the ellipses (…) in the Cloud Template column to retrieve this information from either vCloud Director or Amazon EC2.</td>
</tr>
<tr>
<td>machine</td>
<td></td>
</tr>
<tr>
<td>Details about the networks used for a</td>
<td>Look at the table of virtual machines in the VM Details status window. Expand the window and click the ellipses (…) in the Network Information column to retrieve the network details from either vCloud Director or Amazon EC2.</td>
</tr>
<tr>
<td>virtual machine</td>
<td></td>
</tr>
<tr>
<td>Console to access a virtual machine</td>
<td>Look at the table of virtual machines deployed to vCloud Director in the VM Details status window. You can view the status when the screen refreshes in approximately 30 seconds. The green check mark under the Console column indicates that the virtual machine is powered on, the agent bootstrap and install processes have succeeded.</td>
</tr>
</tbody>
</table>

What to do next

To teardown a deployed application from the cloud, see “Teardown an Application from the Cloud,” on page 131.

To delete a deployment record from vFabric Application Director, see “Delete an Application Deployment from vFabric Application Director,” on page 131.
Teardown an Application from the Cloud

In vFabric Application Director, you can start the teardown process to remove the vCloud Director vApp and associated virtual machines or Amazon EC2 instances from the cloud environment.

You can teardown a deployed application from the vFabric Application Director user interface or the command-line interface. See “Using CLI to Teardown a Deployment,” on page 149.

If a deployment fails after installing one or more virtual machines that are part of an application, or if an application is not used, you can use vFabric Application Director to teardown the application. All virtual machines in the application are removed from their hosts in the cloud.

Deployment teardown from the cloud does not remove the deployment record from the Deployments page in vFabric Application Director. To remove the deployment record from the Deployments page, see “Delete an Application Deployment from vFabric Application Director,” on page 131.

Prerequisites

- Verify that your user account has the ROLE_DEPLOYER deployer role assigned to it.
- Verify that the virtual machines that are part of the application still exist in the cloud. If, for example, your environment has a policy of deleting virtual machines after a certain number of days, the virtual machines might already be deleted.

Procedure

1. On the vFabric Application Director title bar, click the drop-down menu and select Deployments.
2. From the list of deployed applications, select the application to teardown.
3. Click Teardown in the toolbar above the deployment summary.
4. Confirm the teardown.
5. (Optional) If the teardown process fails, repeat the process.

Above the task status windows, the task timeline shows the status of the teardown process. After the virtual machines are deleted successfully from their hosts in the cloud, the status appears in the Task Summary section and in the overall deployment status.

After a teardown process is initiated, you cannot update the deployed virtual machines even if the teardown process fails and the virtual machines exist in the cloud environment. Ask your cloud administrator to teardown the deployment from the cloud.

What to do next

To delete the record of a deployment from the Deployments page, see “Delete an Application Deployment from vFabric Application Director,” on page 131.

Delete an Application Deployment from vFabric Application Director

After you remove an application from the cloud, or if you do not need details for a particular deployment, you can delete the deployment from the Deployments page in vFabric Application Director.

Deleting a deployment from the Deployments page in vFabric Application Director does not delete the deployed application, vApp and its virtual machines, or Amazon EC2 instance from the cloud. To delete an application from the cloud, see “Teardown an Application from the Cloud,” on page 131.

IMPORTANT: If you delete a deployment from the Deployments page without first tearing down the deployed application from the cloud, you must use vCloud Director or Amazon EC2 to remove the residual components in the cloud.
For vCloud Director, delete the vApp and associated virtual machines. For Amazon EC2, stop the instances of the deployment, delete the Security Group corresponding to the deployment, and release the Elastic IP address assigned to the instances of the deployment.

**Prerequisites**

Your user account must have the **ROLE_DEPLOYER** deployer role assigned to it.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Deployments**.
2. From the list of deployed applications, select the deployment to delete.
3. Click **Delete** in the toolbar above the deployment summary.
4. Confirm the deletion.

The deployment record is removed from the Deployments page.

**Cancel a Deployment or an Update Process**

In some cases, if a deployment or update process is in progress indefinitely and does not show either a pass or fail deployment status, vFabric Application Director allows you to mark the deployment or update process as failed without stopping provisioning in the cloud environment.

**Note** When you cancel a deployment or an update process, vFabric Application Director does not delete the deployment or update process.

**Prerequisites**

- Verify that your user account has the **ROLE_DEPLOYER** deployer role assigned to it.
- Verify that you have a deployment or update process in progress indefinitely for more than an hour.

**Procedure**

1. On the vFabric Application Director title bar, click the drop-down menu and select **Deployments**.
2. Select the deployment or updates process that has been in progress for more than an hour.
   - The deployment summary page opens.
3. Click **Cancel** in the toolbar above the deployment summary.
4. Confirm your selection.

**What to do next**

When you mark a deployment as failed, you can teardown the deployment from the cloud or delete the application deployment record from vFabric Application Director. See Chapter 13, “Managing Deployments,” on page 129.

For an update process, you can continue to interact with the deployment by starting another update process. See “Updating Application Deployments,” on page 97.
Collect Logs to Troubleshoot Deployment Failures

vFabric Application Director creates virtual machine-specific logs and an overall deployment log to aid in troubleshooting.

You can use the log pages in the vFabric Application Director user interface to find and correct some problems on your own. If a technical support representative requests more logs, you can retrieve them from the file system of the vFabric Application Director virtual appliance or the virtual machines that were created as part of an application deployment.

Prerequisites

- Verify that you have access to the virtual machine where vFabric Application Director is installed and have the password for logging in with the darwin_user user account. This password was set during installation. See “Start Up the vFabric Application Director Appliance,” on page 22.
- Verify that you have credentials for logging in to the Linux-based virtual machine with root privileges or a Windows-based virtual machine with Administrator privileges.

Procedure

- For failed tasks, use the vFabric Application Director user interface to copy the action script logs.
  a. On the vFabric Application Director title bar, click the drop-down menu and select Deployments.
  b. Click the name of the deployment and expand the Execution Plan status window.
  c. (Optional) If the node is clustered, click the Expand Cluster button first.
  d. On the failed node, click the View Task Information button.
  e. From the drop-down menu, select View Virtual Machine Logs and copy all of the text in the log window.
  
You can paste the log into a text file or email, or create a bug report to send it to a technical support engineer.

- To view failed tasks on a specific virtual machine, use the vFabric Application Director user interface.
  a. On the vFabric Application Director title bar, click the drop-down menu and select Deployments.
  b. Click the name of the deployment and expand the VM Details status window.
  c. Locate the virtual machine and click the icon in the VM Log column.
  
You can copy and paste the virtual machine log into a text file or email, or create a bug report to send it to a technical support engineer.

- To collect logs from the vFabric Application Director appliance, log in to the virtual machine and send catalina.out or localhost.$(date).log output from the /home/darwin/tcserver/darwin/logs directory.

- To collect logs from a virtual machine that was created as part of an application deployment, perform the following tasks:
  a. Copy all of the log files on the deployed virtual machine.
     For a Linux-based virtual machine, navigate to the /opt/vmware-appdirector/agent/log directory.
     For a Windows-based virtual machine, navigate to the \opt\vmware-appdirector\bootstrap.log and \windows\system32\darwin-agent\Date.log files.
  b. In the vFabric Application Director user interface, on the Deployments page, click the name of the deployment, and in the Task Details status window, note the run ID number.
c On the Windows-based virtual machine, navigate to the /Users/darwin/AppData/Local/Temp subdirectory.
This directory contains many logs relating to application components.

d On the Linux-based virtual machine, navigate to the /tmp/runid subdirectory and tar the subdirectory.
This directory contains many logs relating to application components.

To collect detailed logs for API calls made to the vCloud Director and Amazon EC2 back end from vFabric Application Director, perform the following tasks:

a Log in to vFabric Application Director and open the /home/darwin/tcserver/darwin/webapps/darwin/WEB-INF/classes/logback.xml file.
b Locate the comment line <!-- Begin - Cloud Provider Config --> and navigate to the <logger> ... </logger> section.
c Change the value attribute for the <level> tag to DEBUG or INFO.
d Restart the vFabric Application Director server.
sudo service vmware-darwintcserver restart

What to do next
Send the logs to a technical support representative.

Troubleshooting Common Errors During Deployment

If an application deployment fails, the deployment summary page shows a reason for the failure. For the most common errors, you can use these solutions and redeploy the application.

See “Troubleshooting Common Errors During an Update Process,” on page 139.

Table 13-1. Common Errors During Application Deployment

<table>
<thead>
<tr>
<th>Error</th>
<th>Possible Cause</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>An error occurred in the cloud: com.vmware.darwin.cal.api.exceptions.CALOperationException: createVapp: Unable to perform this action. Contact your cloud administrator.</td>
<td>The cloud template used for deployment is not correct. The virtual machines cannot be created in the cloud or the password in the virtual machines cannot be set.</td>
<td>Follow the instructions for creating a custom virtual machine template in “Virtual Machine Requirements for Creating Custom Templates,” on page 39. If you are using a predefined template, contact your vCloud Director administrator to verify that the template is correctly uploaded to the cloud.</td>
</tr>
<tr>
<td>Error in vCloud: The EULA of the entity must be accepted for it to be instantiated.</td>
<td>The cloud template requires a EULA to be accepted.</td>
<td>Disable the Create EULA check box because cloud templates should not have EULAs. For more information, see “Virtual Machine Requirements for Creating Custom Templates,” on page 39.</td>
</tr>
<tr>
<td>com.vmware.darwin.exceptions.CloudException: com.vmware.darwin.cal.api.exceptions.CALOperationException: Unable to compose vapp 'opdpd-xxx-1.0.0-admin-1928-0b37d9cf-1b8d-42a2-8212-a048e81bcb'</td>
<td>Typically, this error is caused by insufficient resources, such as IP addresses or storage, in the cloud. Check virtual machine logs or the vFabric tc Server log in the vFabric Application Director appliance for more detailed error messages from the cloud.</td>
<td>Verify that enough IP addresses are available in the network to which you are deploying. Also, check if the organization vDC has enough storage.</td>
</tr>
<tr>
<td>Error</td>
<td>Possible Cause</td>
<td>Possible Solution</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Error in vCloud: The operation was aborted because you would exceed your stored virtual machine quota. 1 new virtual machine would have been created, and you are already using 100 of a limit of 100.</td>
<td>The deployment error occurred because the cloud user has exceeded the available virtual machine quota.</td>
<td>Stop any unwanted virtual machines in vCloud Director.</td>
</tr>
<tr>
<td>Timed out while connecting to the cloud</td>
<td>The DHCP server might have become unresponsive.</td>
<td>Verify that the DHCP server is running properly.</td>
</tr>
<tr>
<td>An error occurred in the cloud: sendPowerOn: There are insufficient CPU or memory resources to complete the operation.</td>
<td>The deployment error occurred because the virtual machine has exceeded the available CPU or memory.</td>
<td>Reconfigure CPU or memory allocation in the vCloud Director. Delete any unwanted virtual machines that are consuming the same pool of resources in vCloud Director.</td>
</tr>
<tr>
<td>Error in vCloud: There are insufficient IP addresses to complete the operation. You need to add IP addresses to the network that is associated with the object being created or deployed.</td>
<td>Insufficient IP addresses in the network.</td>
<td>Assign additional IP addresses to the network where the application is being deployed. Delete unwanted deployments from vFabric Application Director to free some IP addresses and storage space.</td>
</tr>
<tr>
<td>An error occurred in the cloud: sendPowerOn: Unable to perform this action. Contact your cloud administrator.</td>
<td>The vSphere Distributed Resource Scheduler (DRS) failed to move a virtual machine from one ESX host to another.</td>
<td>Contact your cloud administrator.</td>
</tr>
<tr>
<td>Error in vCloud: The parameter is not supported in the current context: AdminPassword</td>
<td>When the Windows virtual machine instantiated, the administrator password might have been replaced with a randomly generated password.</td>
<td>Remove the randomly generated password.</td>
</tr>
<tr>
<td></td>
<td>1 Shut down Windows virtual machine and open the properties.</td>
<td>1 Shut down Windows virtual machine and open the properties.</td>
</tr>
<tr>
<td></td>
<td>2 Click the Guest Customization tab.</td>
<td>2 Click the Guest Customization tab.</td>
</tr>
<tr>
<td></td>
<td>3 In the Password Reset section, deselect Allow local administrator password and click OK to save your changes.</td>
<td>3 In the Password Reset section, deselect Allow local administrator password and click OK to save your changes.</td>
</tr>
<tr>
<td>Error</td>
<td>Possible Cause</td>
<td>Possible Solution</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Run failed due to failure of task \( node name, agent\_bootstrap \). | The vFabric Application Director agent cannot bootstrap and contact the Darwin server within the set time limit due to several reasons.  
- Agent bootstrap script is not present on the virtual machine template. See “Virtual Machine Requirements for Creating Custom Templates,” on page 39.  
- Agent does not have network connectivity with the Darwin server appliance.  
- vFabric Application Director does not limit the amount of memory that you can allocate for virtual machines during deployment. Insufficient memory allocation might disrupt the virtual machine operating system boot sequence.  
- The NAT networking rules are not configured properly on the routed organization network in vCloud Director.  

Detailed agent logs are available in the deployed Linux-based virtual machine in the \( /opt/vmware-appdirector/agent/logs \) directory. For a Windows-based virtual machine, navigate to the \( /opt/vmware-appdirector/\) \( \text{bootstrap.log} \) and \( \text{windows}\backslash system32\text{darwin-agentDate.log} \) files.  
Run network diagnostics to verify that you can ping the Darwin server appliance from the virtual machine. For the specific hardware and system requirements, see the operating system documentation. For the routed organization network, see the vCloud Director documentation to configure SNAT rules. |                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

| Run failed due to failure of task \( node name,task name \).        | One of the execution plan tasks failed. The following problems are some of the reasons that an execution plan task might fail:  
- A property of type content is not set to a valid URL. The agent log displays the following message: Exception while running task \( \langle node name\rangle\langle task name\rangle\rangle\), message Cannot fetch content, url http://192.0.2.255:8443/darwin/api/file/download/123 is not accessible or invalid. cause IOException: Server returned HTTP response code: 500 for URL: http://192.0.2.255:8443/darwin/api/file/download/123  
Run failed due to failure of task \( \langle node name\rangle\langle task name\rangle\rangle\)  
- A property name has hyphens and other characters that are not valid for shell scripts.  
- The repository URL is not set to the correct operating system version.  
- Action scripts might need Java installed on the cloud template and Java is not installed in the cloud template. | Expand the Execution Plan status window on the deployment summary page and identify the task that failed. If the node is clustered, click the Expand Cluster button ( ) first. Click the View Task Information button ( ) and click View Virtual Machine Logs from the drop-down menu.  
If the task log does not indicate the failure, examine the agent logs in the deployed virtual machine. Agent logs are in the logs directory under the \( /opt/vmware-appdirector/agent \) folder. |
<table>
<thead>
<tr>
<th>Error</th>
<th>Possible Cause</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>An error occurred in the cloud: createVapp: The operation failed because VirtualCenter &quot;Darwin-vCenter-5.0&quot; is not connected.</td>
<td>The vCenter instance is not connected to vCloud Director.</td>
<td>Request your cloud administrator to connect the virtual center to the vCloud Director instance.</td>
</tr>
<tr>
<td>An application deployment has failed but a task is still running in the execution deployment summary. This problem does not generate an error message.</td>
<td>In some cases, one of the tasks in the application deployment is running. At the same time, another task fails to deploy. vFabric Application Director immediately marks the entire deployment as failed. However, the task that is in progress continues to run until it finishes.</td>
<td>Expand the Execution Plan status window on the deployment summary page. Diagnose the cause of the long running task and fix blueprint or network connectivity problems. If the problem is intermittent, you can teardown the failed deployment from the cloud. See “Teardown an Application from the Cloud,” on page 131.</td>
</tr>
<tr>
<td>A deployment is in progress indefinitely and does not show either a pass or fail deployments status. This problem does not generate an error message.</td>
<td>Intermittent loss of connection with the Tomcat service, the server restarts during a deployment process, or the agent bootstrap fails. <strong>Note</strong> This problem does not surface for all connections failures. It happens based on the state of the deployment when the connection failure occurred.</td>
<td>Expand the Execution Plan status window on the deployment summary page. Diagnose the cause of the long running task and fix blueprint or network connectivity problems. If the problem persists, cancel the deployment. This action marks the deployment as failed without stopping provisioning so that you can interact with the application. See “Cancel a Deployment or an Update Process,” on page 132. If the problem is intermittent, you can teardown the failed deployment from the cloud. See “Teardown an Application from the Cloud,” on page 131.</td>
</tr>
<tr>
<td>A deployment with one or more custom task is in progress indefinitely and the deployment cannot be stopped from vFabric Application Director. This problem does not generate an error message.</td>
<td>One or more custom tasks in the deployment might not have been given names. A custom task has an infinite loop or is running a long process.</td>
<td>In vCloud Director, stop the vApp corresponding to the deployment. You can also reclaim the cloud resources from the vCloud Director application, if you have the appropriate privileges and delete the vApp corresponding to the deployment. In the Deployment Profile wizard, verify that the custom tasks have names and redeploy the application. See “Deploy an Application,” on page 96. If the problem persists, cancel the deployment. This action marks the deployment as failed without stopping tasks so that you can interact with the application. See “Cancel a Deployment or an Update Process,” on page 132.</td>
</tr>
<tr>
<td>One or more custom tasks or action scripts run indefinitely and the deployment fails with a timeout error.</td>
<td>Processes prompting for user interaction might pause the custom task or action script.</td>
<td>Close all of the processes that prompt for user interaction before running a custom task or action script.</td>
</tr>
<tr>
<td>Error</td>
<td>Possible Cause</td>
<td>Possible Solution</td>
</tr>
<tr>
<td>-------</td>
<td>---------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Exception while running task <em>(node name, task name)</em>, message Cannot fetch content, url <a href="https://192.0.2.255:8443/darwin/api/file/download/">https://192.0.2.255:8443/darwin/api/file/download/</a> is not a accessible or invalid. cause SunCertPathBuilderException: unable to find valid certification path to requested target</td>
<td>The deployment error occurred because the content property might not have a valid value assigned to it.</td>
<td>Add a URL in the content property value for the deployment to successfully complete. See “Types of Properties,” on page 61.</td>
</tr>
<tr>
<td>Unable to automatically setup: Invalid username/password The Hyperic Agent does not start but the deployment status was successful.</td>
<td>The Hyperic Agent server username and password might be incorrect, causing the hq-agent.sh script to fail.</td>
<td>Type the correct Hyperic Agent server username and password in the configure script and start the application deployment.</td>
</tr>
<tr>
<td>Overriding the blueprint property value in the Deployment Profile wizard and resetting the blueprint value to the original version causes the value to be converted to a string as opposed to expressions.</td>
<td>Blueprint properties reset to the original values in the Deployment Profile wizard are not evaluated as expressions.</td>
<td>Reload the deployment profile and deploy the application without resetting the blueprint value.</td>
</tr>
<tr>
<td>Deploying the Microsoft SQL Server 2008 R2 Express service on a Windows virtual machine configured to join a domain during deployment, the installation life cycle stage fails.</td>
<td>The domain name begins with darwin. This is also a known issue in the Microsoft SQL Server installation program.</td>
<td>Rename the domain without using darwin as the prefix. You can also create a Windows template, name the bootstrap service account something other than darwin, and configure the service to run as that account.</td>
</tr>
<tr>
<td>Windows deployment with a PowerShell script that includes a Start-Job command for running programs in the background is unresponsive.</td>
<td>The PowerShell script exits because the Start-Job command is not running the program in the background successfully.</td>
<td>Use the Start-process command in the script with the appropriate parameters to start the program in a separate process.</td>
</tr>
<tr>
<td>The Media application component does not extract files to the C:\Windows\System32 folder for a Windows-based application deployment and the deployment is not marked as failed. This problem does not generate an error message.</td>
<td>The C:\Windows\System32 folder is a Windows protected folder that prohibits unauthorized file creation. The deployment is not marked as failed because the file extraction utility is not exiting with an error status.</td>
<td>Set full Administrator privileges for the Windows system directory to allow files to be extracted to the C:\Windows\System32 folder.</td>
</tr>
<tr>
<td>Deploying an application to the Amazon EC2 environment fails.</td>
<td>The deployment error occurred because the IP address for the cloud tunnel in the deployment environment might be inaccurate.</td>
<td>Verify that the IP address for the cloud tunnel is valid and the Endpoint VM is correctly set up. See “Create an Amazon EC2 Deployment Environment,” on page 57 and “Create a Cloud Tunnel to Connect to Amazon EC2,” on page 55.</td>
</tr>
</tbody>
</table>
Table 13-1. Common Errors During Application Deployment (Continued)

<table>
<thead>
<tr>
<th>Error</th>
<th>Possible Cause</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>When you deploy several applications to Amazon EC2 continuously, the deployments might fail and a request limit exceeded error appears. When you attempt to teardown the deployment, the process seems to be successful in vFabric Application Director but the applications still exist in the Amazon EC2 environment.</td>
<td>You might have exceeded the allocated Elastic IP address limit, the number of Amazon EC2 instances, or the number of API requests allowed in an hour.</td>
<td>Open the AWS management console, remove the instances, and release the Elastic IP addresses. Contact Amazon support to request an increase in the instance or Elastic IP address limit.</td>
</tr>
<tr>
<td>Deployments to Amazon EC2 from within a corporate network might fail.</td>
<td>▪ A network problem might cause the cloud tunnel connection to be lost. &lt;br&gt;▪ The Endpoint VM is selected in the wrong VPC. &lt;br&gt;▪ Security group or internal IP address settings for the Endpoint VM is incorrect.</td>
<td>Reestablish the lost cloud tunnel network connection. Assign the Endpoint VM to the correct VPC in the Amazon Region. Determine whether the Endpoint VM has the correct security group or internal IP address settings. See “Configure Amazon EC2 Environment for vFabric Application Director,” on page 50.</td>
</tr>
</tbody>
</table>

Troubleshooting Common Errors During an Update Process

If an update process fails, the deployment summary page shows a reason for the failure. For the most common errors, you can use these solutions and initiate another update process.


Table 13-2. Common Errors During an Update Process to Scale a Deployment or Modify Configuration

<table>
<thead>
<tr>
<th>Error</th>
<th>Possible Cause</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>When initiating an update process on a failed update deployment, an Error communicating with the server. Please contact the administrator error appears.</td>
<td>The vFabric Application Director server times out during the update process.</td>
<td>Use the CLI to initiate an update process of the failed deployment. See “Deploying and Updating an Application Using CLI,” on page 146.</td>
</tr>
<tr>
<td>The update configuration CLI command fails and a No properties are specified for this update. error appears.</td>
<td>The vFabric Application Director appliance is running a license edition that does not support updating the configuration of a deployed application.</td>
<td>Create a vFabric Application Director appliance and install the VMware vFabric Application Director for Release Automation edition. See “Start Up the vFabric Application Director Appliance,” on page 22 and “Deploying and Updating an Application Using CLI,” on page 146.</td>
</tr>
<tr>
<td>Error</td>
<td>Possible Cause</td>
<td>Possible Solution</td>
</tr>
<tr>
<td>-------</td>
<td>---------------</td>
<td>-------------------</td>
</tr>
</tbody>
</table>
| During an application deployment or an update process, a service is set to reboot so that the agent bootstrap can restart the virtual machine after an action script runs successfully. The scripts that take more than 15 minutes to provision and reboot might cause the deployment or an update process to fail. | The task scheduler that pings the server times out after 15 minutes. | Increase the timeout property value to more than 15 minutes.  
1. Open the `/etc/init.d/vmware-darwin-tcserv` file.  
2. In the CATALINA_OPTS section, change the java system nodetask.time.out property to more than 15 minutes.  
3. Restart the vFabric Application Director server.  
   `sudo service vmware-darwin-tcserv restart` |
| Could not connect to messaging server | vFabric Application Director is not able to connect to the RabbitMQ server. | Type `service rabbitmq-server status` to verify that the RabbitMQ server is running. Troubleshoot any RabbitMQ server connection problems. |
| The deployment no longer exists on the cloud | The application deployment might have been deleted from the cloud. | In the cloud environment, verify whether the deployment is deleted. If it is deleted, you cannot initiate an update process. |
| An error occurred when provisioning the cloud: Not all VMs in deployment 'apppd-xxx-3.0.0-admin-9-9a7bd508-da44e4-98f9-7c862758587f' are on. 4 are powered off | Some of the virtual machines in the vApp might be powered off. | Locate the deployment in vCloud Director and power on all of the virtual machines in the vApp. See the vCloud Director documentation. |
| An error occurred while connecting to the data storage | The redis server might be down. | Type `service redis status` to verify that the redis server is running. Troubleshoot any redis server connection problems. |
| An error occurred when provisioning the cloud: Virtual Machine 'vmName_2_' already exists | You might be attempting to initiate an update process to scale a deployment that has previously failed to scale. | Locate the vApp in vCloud Director and delete the failed virtual machine. |
| A value must be provided for property 'PropertyName' of component 'PropertyName' because the previous update task was unsuccessful in the update wizard. | You might be attempting to initiate an update process to modify configuration on a previously failed update by resetting the failed property such as port number, and trying to proceed.  
You might be trying to initiate an update process to modify configuration by changing a property that has a dependent property. The task on the changed component succeeded, but the task on the dependent component failed. When you initiate another update process to modify configuration, the dependent property is highlighted as failed as you try to proceed with the update process. | Add new values to all of the failed properties. If you do not want to change the properties, modify the action script to ignore the failed properties.  
Add new value to the property of the failed update. If you do not want to change the properties, modify the action script to ignore the failed properties.
Using the vFabric Application Director CLI

The vFabric Application Director CLI is a Spring Roo-based client that communicates to the vFabric Application Director server over HTTPS using REST APIs.

User accounts with the system administrator role can use the vFabric Application Director CLI to create and manage users, groups, and LDAP configurations. User accounts with the deployer roles can use the CLI to deploy or teardown applications from the cloud. See “Predefined Users, Groups, and Roles,” on page 30.

**Note** The predefined user accounts are disabled by default. If you do not enable these accounts, you can use only the admin user. The password for the admin user is the admin password that was set the first time the appliance was started.

This chapter includes the following topics:

- “General CLI Options,” on page 141
- “Managing Users and Groups,” on page 142
- “Managing LDAP Configurations,” on page 143
- “Managing Cloud Tunnels,” on page 145
- “Deploying and Updating an Application Using CLI,” on page 146
- “Using CLI to Teardown a Deployment,” on page 149

**General CLI Options**

With the vFabric Application Director command-line interface options, you can create and manage users and groups, deploy an application, update a deployed application, or teardown an application from the cloud.

After you log in to the CLI program, in the roo shell prompt, press the Tab key to display the list of available command options in vFabric Application Director. If you use multiple words with spaces between the words, enclose the words in quotation marks. Often after you enter a command, the system displays many details, in addition to indicating whether the command was successful.

**Note** Do not use command options that are not available in CLI.

The general CLI options are available to all user groups in vFabric Application Director.

**Table 14-1. General CLI Commands**

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>help</td>
<td>Lists use information.</td>
</tr>
<tr>
<td>cliversion</td>
<td>Displays the CLI version information.</td>
</tr>
</tbody>
</table>
### Table 14-1. General CLI Commands (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>login</td>
<td>Logs in the current user to the <em>roo</em> shell.</td>
</tr>
<tr>
<td>logout</td>
<td>Logs out the current user without closing the <em>roo</em> shell prompt. You can log out and log in as a different user.</td>
</tr>
<tr>
<td>status</td>
<td>Indicates whether a user is logged in. If a user is logged in, user name is displayed.</td>
</tr>
<tr>
<td>exit</td>
<td>Exits the CLI program.</td>
</tr>
</tbody>
</table>

### Managing Users and Groups

To manage users and groups from the CLI, you must create users and groups.

Verify that your user account has the **ROLE_SYSTEM_ADMIN** system administrator role assigned to it.

See “Use the vFabric Application Director CLI to Create Users and Groups,” on page 32.

**Note** Unknown CLI options are ignored by the *roo* shell.

### Table 14-2. Manage Users and Groups in the CLI

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>change-password</td>
<td>Changes the password of the current user.</td>
</tr>
<tr>
<td>change-user-password <strong>--username</strong> UserName</td>
<td>Changes a user's password. If you run the <strong>--password</strong> parameter with the log in command, your password is saved as plain text in the <em>log.roo</em> file. For added security, delete this log file.</td>
</tr>
<tr>
<td>create-group <strong>--name</strong> GroupName <strong>--description</strong> Description</td>
<td>Creates a group. For example, to create a group called Test Group, use the command <code>create-group --name &quot;Test Group&quot;</code>.</td>
</tr>
<tr>
<td>create-user <strong>--username</strong> UserName <strong>--firstName</strong> FirstName <strong>--lastName</strong> LastName <strong>--enabled</strong> true <strong>--roles</strong> ROLE_SYSTEM_ADMIN, ROLE_CATALOG_ADMIN, ROLE_CLOUD_ADMIN, ROLE_SYSTEM_INTEGRATOR, ROLE_DEPLOYER, ROLE_APP_ARCHITECT <strong>--group</strong> GroupName <strong>--email</strong> EmailAddress</td>
<td>Creates a user. If you specify all the roles shown in this example, the user is the equivalent of the admin user. Specify only the roles the specific user must have. The valid values for the <strong>enabled</strong> option are true, yes, 1, false, no, and 0. If you run the <strong>--password</strong> parameter with the log in command, your password is saved as plain text in the <em>log.roo</em> file. For added security, delete this log file.</td>
</tr>
</tbody>
</table>
Table 14-2. Manage Users and Groups in the CLI (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>update-user --username UserName --roles ROLE_SYSTEM_ADMIN, ROLE_CLOUD_ADMIN</code></td>
<td>Changes the roles assigned to a user. In this example, the roles <code>ROLE_SYSTEM_ADMIN</code>, <code>ROLE_CLOUD_ADMIN</code> are used. Substitute a comma-separated list of the roles you want the user to have. Do not add a space between the comma-separated list.</td>
</tr>
<tr>
<td><code>update-user --username UserName --group GroupName</code></td>
<td>Change a user's group.</td>
</tr>
<tr>
<td><code>enable-user --username UserName</code></td>
<td>Enables a user account.</td>
</tr>
<tr>
<td><code>disable-user --username UserName</code></td>
<td>Disables a user account. <strong>CAUTION</strong> The disable option also allows you to disable your user account. If you are the only user with system administrator privileges, you cannot reenable your user account.</td>
</tr>
<tr>
<td><code>list-user</code></td>
<td>Lists information about all users.</td>
</tr>
<tr>
<td><code>list-user --username UserName</code></td>
<td>Displays information about a specific user.</td>
</tr>
<tr>
<td><code>list-roles</code></td>
<td>Lists information about all the roles.</td>
</tr>
<tr>
<td><code>list-group</code></td>
<td>Shows information about all the groups.</td>
</tr>
</tbody>
</table>

Managing LDAP Configurations

With vFabric Application Director you can perform various operations such as activating, importing, updating, or deleting an LDAP configuration.

When you run the import command, LDAP completes the following processes.

1. Verifies the existence of an LDAP user or group name in the system.
2. Creates an entry in the vFabric Application Director database for that LDAP user or group.
3. Adds the user or group SID information in the database.
4. Associates the LDAP user or group in the database with the specified vFabric Application Director group and roles.

**Note** While importing existing LDAP users and groups, LDAP does not collect or transfer any secure information from the directory to vFabric Application Director.

The existing LDAP users or groups that are imported to vFabric Application Director and assigned roles can log in to the vFabric Application Director Web interface using their LDAP credentials.

Verify that your user account has the `ROLE_SYSTEM_ADMIN` system administrator role assigned to it.

See “Create and Activate an LDAP Configuration,” on page 33.
### Table 14-3. Using CLI to Manage LDAP Configurations

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>create-ldap-config</code></td>
<td>Creates an LDAP configuration and saves the configuration in the vFabric Application Director database.</td>
</tr>
<tr>
<td><code>activate-ldap-config --configname LDAPName</code></td>
<td>Activates an LDAP configuration in the vFabric Application Director server to authenticate against an LDAP configuration. <strong>Note</strong>: The LDAP configuration name should not include any periods.</td>
</tr>
<tr>
<td><code>update-ldap-config --configname LDAPName</code></td>
<td>Updates an existing LDAP configuration in the system. <strong>Note</strong>: The LDAP configuration name should not include any periods.</td>
</tr>
<tr>
<td><code>import-ldap-user --configname LDAPName --group GroupName --roles ROLE_DEPLOYER</code></td>
<td>Imports an existing LDAP user and assigns the user the vFabric Application Director deployer role. In this command, the <code>GroupName</code> is the name of the vFabric Application Director group that the LDAP user will be associated with. <strong>Note</strong>: You must have an active LDAP configuration before you can import LDAP users.</td>
</tr>
<tr>
<td><code>import-ldap-group --configname LDAPName --group GroupName --roles ROLE_CLOUD_ADMIN</code></td>
<td>Imports and existing group and assigns the group the vFabric Application Director cloud administrator role. <strong>Note</strong>: The LDAP configuration name should not include any periods. For example, to import a group called Admin Group, use the command <code>import-ldap-group --name Admin --group &quot;Admin Group&quot;</code>.</td>
</tr>
<tr>
<td><code>update-ldap-user --name UserName --group GroupName --roles ROLE_APP_ARCHITECT</code></td>
<td>Updates an existing user that was imported from the LDAP directory and assigns the user the vFabric Application Director application architect role.</td>
</tr>
<tr>
<td><code>update-ldap-group --name UserName --group GroupName --roles ROLE_CATALOG_ADMIN,ROLE_CLOUD_ADMIN</code></td>
<td>Updates an existing group that was imported from the LDAP directory. In this example, the group has the <code>ROLE_CATALOG_ADMIN</code> and <code>ROLE_CLOUD_ADMIN</code> roles assigned.</td>
</tr>
<tr>
<td><code>list-ldap-configs</code></td>
<td>Lists existing LDAP configurations in the system.</td>
</tr>
<tr>
<td><code>list-ldap-principals</code></td>
<td>Lists all of the users and groups imported from the LDAP directory.</td>
</tr>
<tr>
<td><code>print-active-ldap-config</code></td>
<td>Lists the details of the active LDAP configurations.</td>
</tr>
<tr>
<td><code>print-named-ldap-config --configname LDAPName</code></td>
<td>Displays the details of the named LDAP configuration. <strong>Note</strong>: The LDAP configuration name should not include any periods.</td>
</tr>
<tr>
<td><code>delete-ldap-user --name UserName --configname LDAPName</code></td>
<td>Removes a user from the local database. <strong>Note</strong>: The LDAP configuration name should not include any periods.</td>
</tr>
</tbody>
</table>
### Table 14-3. Using CLI to Manage LDAP Configurations (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>delete-ldap-group --name GroupName --configname LDAPName</code></td>
<td>Removes a group from the local database.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>The LDAP configuration name should not include any periods.</td>
</tr>
<tr>
<td><code>disable-ldap</code></td>
<td>Deactivates the current LDAP configuration and removes the LDAP authentication from the authentication chain in the system, so that only the local authentication is available. The deactivated LDAP configurations are available on the system and can be reactivated.</td>
</tr>
</tbody>
</table>

### Managing Cloud Tunnels

You can perform operations such as creating, updating, enabling, or deleting a secure cloud tunnel connection between the vFabric Application Director appliance and an Endpoint VM in an Amazon EC2 VPC.

Verify that your user account has the `ROLE_CLOUD_ADMIN` cloud administrator role assigned to it.

See “Create a Cloud Tunnel to Connect to Amazon EC2,” on page 55.

### Table 14-4. Using the CLI to Manage Cloud Tunnels

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>create-cloud-tunnel --name TunnelName --description &quot;TunnelDescription&quot; --enabled false --externalAddress EndpointVMElasticIP --sshPort 22 --internalAddress EndpointVMPrivateIP --proxyUrl ProxyURL --username UserName --privateKeyPath PrivateKeyFilePath</code></td>
<td>Creates a secure cloud tunnel. For example, you can use the following command to create a cloud tunnel called EC2 Tunnel. <code>create-cloud-tunnel --name EC2 Tunnel --description &quot;Test EC2 tunnel connection&quot; --enabled false -- externalAddress 50.18.100.100 --sshPort 22 --internalAddress 192.0.2.255 --proxyUrl http://proxy.vmware.com:3128 -- username root --privateKeyPath /tmp/private_key_file</code></td>
</tr>
<tr>
<td><code>update-cloud-tunnel --name PreviousTunnelName --new-name NewTunnelName --description &quot;TunnelDescription&quot; --externalAddress EndpointVMElasticIP --sshPort 22 --internalAddress EndpointVMPrivateIP --proxyUrl ProxyURL --username UserName --privateKeyPath PrivateKeyFilePath</code></td>
<td>Changes the values of various parameters of an existing cloud tunnel. For most of the parameters, this command works only if the cloud tunnel is disabled.</td>
</tr>
<tr>
<td><code>list-cloud-tunnels --name TunnelName</code></td>
<td>Retrieves a list of all the available cloud tunnels or a particular cloud tunnel, when specified. To view the details of a particular cloud tunnel, use the command <code>list-cloud-tunnels --name EC2 Tunnel</code>.</td>
</tr>
<tr>
<td><code>enable-cloud-tunnel --name TunnelName</code></td>
<td>Enables an existing cloud tunnel. This command initiates a background operation that establishes the secure cloud tunnel connection.</td>
</tr>
<tr>
<td><code>disable-cloud-tunnel --name TunnelName</code></td>
<td>Disables an existing cloud tunnel. This command initiates a background operation that disconnects the secure cloud tunnel connection.</td>
</tr>
</tbody>
</table>
### Table 14-4. Using the CLI to Manage Cloud Tunnels (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>delete-cloud-tunnel --name TunnelName</td>
<td>Deletes an existing cloud tunnel. The cloud tunnel must be disabled for it to be deleted. Removes the cloud tunnel connection between the vFabric Application Director appliance and the Amazon EC2 environment.</td>
</tr>
<tr>
<td>test-cloud-tunnel --name TunnelName</td>
<td>Verifies a cloud tunnel secure SSH connection and reports the current status. Enabling or disabling a cloud tunnel initiates the connection or disconnection operation in the background. This command reports the status of that background operation. In addition, the command checks whether the connection will function properly with vFabric Application Director during a deployment. Use this command after you enable or disable a cloud tunnel to confirm that your request succeeded.</td>
</tr>
</tbody>
</table>

### Deploying and Updating an Application Using CLI

To deploy an application, your user account must have the deployer role (ROLE_DEPLOYER) assigned to it.

Before you deploy an application, verify that you have a deployment profile that is complete, saved, and free of any validation errors. You also need the name of the application. Verify that the deployed application includes a clustered node before you initiate an update process to scale a clustered node.

You can perform a quick deploy of an application, check the status of an application deployment, create an update profile to scale out an application, or modify service and application component configurations of deployed applications using the CLI. You can also track the application deployment status from the vCloud Director and Amazon EC2 user interfaces.

You can use the vFabric Application Director user interface to perform all of these tasks. See “Quick Deploy an Application,” on page 97, “Checking the Status of a Deployment,” on page 104, “Initiate an Update Process to Scale Deployments,” on page 98, or “Initiate an Update Process to Modify Configurations,” on page 100.

**IMPORTANT** To update deployed applications, install the vFabric Application Director for Release Automation edition.

### Table 14-5. Deploy or Update an Application in the CLI

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>deploy-application --destination ApplicationName-ApplicationVersion-DeploymentProfileName</td>
<td>Deploys an application blueprint. For example, to deploy the Clustered Dukes Bank application, version 2.1.0, with the prod-dep deployment profile, use the command deploy-application --destination &quot;Clustered Dukes Bank App-2.1.0-prod-dep&quot;. Type deploy and press the Tab key to display the possible options for the command. If the list does not display quotation marks for multiple words with spaces between the words, enclose the words in quotation marks.</td>
</tr>
<tr>
<td>deployment-status --deploymentName DeploymentName</td>
<td>Shows the status of the latest deployment. If an application deployment fails, you can collect the virtual machine-specific log files to troubleshoot the problem. See “Collect Logs to Troubleshoot Deployment Failures,” on page 133.</td>
</tr>
</tbody>
</table>
Table 14-5. Deploy or Update an Application in the CLI (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| `deploy-application --destination ApplicationNameApplicationVersion-DeploymentProfileName --propertiesFile FileName.xml` | Deploys the latest application version with new property values. You can create a properties file that defines new property values for required properties and properties that are overridable at deployment. The following code example defines new values for the global_conf, http_node_port, and http_server_port properties in the Apache_LB and AppServer nodes.  

```xml
<?xml version="1.0" encoding="UTF-8"?>
<ns1:profile-properties xmlns:ns1="http://www.vmware.com/darwin/schema/beans/api">
  <node>
    <node-component>
      <name>Apache_LB</name>
      <property>
        <key>global_conf</key>
        <value><![CDATA[http://$darwin.server.ip:8080/darwin/conf/darwin_global.conf]]></value>
      </property>
    </node-component>
    <node-component>
      <name>AppServer</name>
      <property>
        <key>http_server_port</key>
        <value>newValue3</value>
      </property>
    </node-component>
  </node>
</ns1:profile-properties>
```

For example, to deploy the Spring Travel application version 1.0.5, with the staging-dep deployment profile, and new.props properties file, use the command `deploy-application --destination "Spring Travel-1.0.5-staging-dep" --propertiesFile ~/new.props.xml` Type `deploy` and press the Tab key to display the possible options for the command. If the list does not display quotation marks for multiple words with spaces between the words, enclose the words in quotation marks.
### Table 14-5. Deploy or Update an Application in the CLI (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| `update-scaleout --deploymentName DeploymentName --propertiesFile FileName.xml` | Creates an update profile to scale a deployed application. You must create a properties file that defines a new cluster size value for a node. The following code example defines cluster size values for the AppServer1 and DBServer1 nodes.  
```xml
<?xml version="1.0" encoding="UTF-8"?>
<ns1:scaleout-properties xmlns:ns1="http://www.vmware.com/darwin/schema/beans/api">
  <cluster-node>
    <name>AppServer1</name>
    <scale-out-by>2</scale-out-by>
  </cluster-node>
  <cluster-node>
    <name>DBServer1</name>
    <scale-out-by>3</scale-out-by>
  </cluster-node>
</ns1:scaleout-properties>
```
For example, to scale out the Clustered Dukes Bank application, version 3.0.0 with scaleout properties file, use the command `update-scaleout --deploymentName "appd-Clustered Dukes Bank App-3.0.0-admin-6-cc0a3b20-43a0-4a22-be3e-49d4f31ab8e8" --propertiesFile ~/scaleout.xml`
Type `update` and press the Tab key to display the possible options for the command. If the list does not display quotation marks for multiple words with spaces between the words, enclose the words in quotation marks. |
| `update-config --deploymentName DeploymentName --configUpdatePropertiesFile ConfigUpdatePropertiesFileName.xml` | Creates an update profile to modify configurations of existing services or application components in a deployed application. You must create a config update properties file to configure an existing service. The following code example modifies the global_conf and db_port properties in the vFabric tc Server service of the AppServer node.
```xml
<?xml version="1.0" encoding="UTF-8"?>
<ns1:config-update-properties xmlns:ns1="http://www.vmware.com/darwin/schema/beans/api">
  <node>
    <name>AppServer</name>
    <node-component>
      <name>tcServer</name>
      <property>
        <key>global_conf</key>
        <value><![CDATA[http://$ {darwin.server.ip}: 8080/darwin/conf/darwin_global.conf]]></value>
      </property>
      <property>
        <key>db_port</key>
        <value>1111</value>
      </property>
    </node-component>
  </node>
</ns1:config-update-properties>
```
Table 14-5. Deploy or Update an Application in the CLI (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>For example, to modify a configuration in the Clustered Dukes Bank application, version 3.0.0 with configupdate properties file, use the command <code>update-config --deploymentName &quot;appd-Clustered Dukes Bank App-3.0.0-admin-2-19d63535-673e-4766-b380-de4e6e3676a&quot; --configUpdatePropertiesFile ~/configupdate.xml</code></td>
</tr>
<tr>
<td></td>
<td>Type <code>update</code> and press the Tab key to display the possible options for the command. If the list does not display quotation marks for multiple words with spaces between the words, enclose the words in quotation marks.</td>
</tr>
</tbody>
</table>

Using CLI to Teardown a Deployment

To teardown a deployment, your user account must have the deployer role (ROLE_DEPLOYER) assigned to it.

Before you teardown a deployed application from the cloud, verify that you have the deployment name of the deployed application. You can monitor the status of the teardown process from the vFabric Application Director user interface. See "Teardown an Application from the Cloud," on page 131.

Table 14-6. Remove Deployment in the CLI

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>teardown --deploymentName DeploymentName</td>
<td>Teardown a deployed application from the vCloud Director. For example, the <code>teardown --name appd-TestApp-1.0.0-admin-3-a99389aa-a05b-45c2-b6a8-bac4a421178e</code> command removes the application from the cloud.</td>
</tr>
<tr>
<td></td>
<td>Type <code>teardown</code> and press the Tab key to display the possible options for the command. If the list does not display quotation marks for multiple words with spaces between the words, enclose the words in quotation marks.</td>
</tr>
</tbody>
</table>
Using the CLI Import and Export Functionality

You can import and export preconfigured application blueprints and their associated deployment profiles, services, and available custom tasks across vFabric Application Director instances to help you further customize your applications.

**NOTE** You can import and export packages created only in vFabric Application Director 5.0.

For example, if you decide to switch from an instance with vFabric Application Director for Provisioning to a new instance with vFabric Application Director for Release Automation edition, you can transfer your preconfigured application blueprints and associated components to the new instance by using the import and export features.

You can start the CLI from a vFabric Application Director appliance or from a remote machine. See “Start the CLI in an Appliance,” on page 30 or “Start the CLI Remotely,” on page 31.

Verify that your user account has the ROLE_APP_ARCHITECT application architect role and the ROLE_CATALOG_ADMIN catalog administrator role assigned to it.

- **Using the CLI Export Command** on page 151
  
The CLI export command creates a package that consists of applications and their associated blueprints and deployment profiles, services, logical templates, and available custom tasks to export between different vFabric Application Director instances.

- **Using the CLI Import Command** on page 153
  
The CLI import command imports a package that includes applications and their associated blueprints and deployment profiles, services, logical templates, and available custom tasks between different vFabric Application Director instances.

**Using the CLI Export Command**

The CLI export command creates a package that consists of applications and their associated blueprints and deployment profiles, services, logical templates, and available custom tasks to export between different vFabric Application Director instances.

You cannot export application and service version names that contain characters other than space, tab, underscore, new line, or carriage return.

**NOTE** The export package includes the applications or services you selected to import. When you run the import command, the entire package is imported to the designated server. You cannot selectively import elements from the export package.

When you export an application or service, all of the secure properties are removed by default to avoid exporting sensitive information such as passwords to another vFabric Application Director instance.
You can also export a package using the vFabric Application Director REST APIs. Packages exported using REST APIs are encoded in the Base64 format and cannot be imported using the CLI. To import these packages using the CLI, you must use an external tool to convert the Base64 encoded package into a plain text package file.

To import packages using REST APIs that were previously exported with the CLI, you must use an external tool to convert the plain text package file into a Base64 encoded package. See VMware vFabric Application Director API Programming guide.

Table 15-1. CLI Export Commands

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>export-package --exportFilePath ExportFilePath</code> --applicationVersion ApplicationVersionSpecification --serviceVersion ServiceVersionSpecification</td>
<td>Exports an application and service version. For example, you can specify the exportFilePath as <code>/Users/UserName/tmp/test.xml</code>. Note: If the specified directory in the ExportFilePath is not available when you run the export command, the operation fails. The ApplicationVersionSpecification command option includes <code>ApplicationName: major . minor . micro { qualifier}</code>. Specifying ServiceVersionSpecification is optional because, during the export process all of the application dependencies are exported. The command option includes, <code>ServiceName: major . minor . micro { qualifier}</code>. You can export multiple application and service versions at the same time. For example, <code>export-package --exportFilePath /Users/test_user/tmp/test.xml --applicationVersion &quot;Clustered Dukes Bank App:2.1.0, Clustered Zimbra App:7.1.0, App_with_Qualifier:11.22.33-RELEASE&quot; --serviceVersion &quot;Hyperic HQ Agent: 4.6.0, Hyperic HQ Server:4.6.0, Preinstalled vFabric GemFire Server for Ubuntu:6.6.8&quot;</code> Note: If the application version contains spaces, you must put the application version in double quotation marks.</td>
</tr>
<tr>
<td><code>export-package --exportFilePath allApps.pkg</code> --applicationVersion ALL</td>
<td>Exports all of the applications and associated versions.</td>
</tr>
<tr>
<td><code>export-package --exportFilePath allServiceVersions.pkg</code> --serviceVersion ALL</td>
<td>Exports all of the services and associated versions.</td>
</tr>
<tr>
<td><code>export-package --exportFilePath allAVSV.pkg</code> --applicationVersion ALL --serviceVersion ALL</td>
<td>Exports all of the application and service versions.</td>
</tr>
</tbody>
</table>
Using the CLI Import Command

The CLI import command imports a package that includes applications and their associated blueprints and deployment profiles, services, logical templates, and available custom tasks between different vFabric Application Director instances.

Before you import a deployment profile, save the custom task you added to the execution deployment plan. See “Add Custom Tasks,” on page 94.

NOTE The export package includes the applications or services you selected to import. When you run the import command, the entire package is imported to the designated server. You cannot selectively import elements from the export package.

You can also import a package using the vFabric Application Director REST APIs. Packages exported using REST APIs are encoded in the Base64 format and cannot be imported using the CLI. To import these packages using the CLI, you must use an external tool to convert the Base64 encoded package to a plain text package file.

To import packages using REST APIs that were previously exported with the CLI, you must use an external tool to convert the plain text package file to a Base64 encoded package. See the VMware vFabric Application Director API Programming guide.
Table 15-2. CLI Import Commands

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>import-package --importFilePath ImportFilePath --conflictResolutionAction CHECK</code></td>
<td>Compares the package and repository elements, lists the package elements that match and do not match the repository. Elements are not imported to the repository when you run this command. The match is based on the name and version qualifiers such as major, minor, micro, and optional. You must set one of the following settings for the <code>conflictResolutionAction</code> command option.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><code>import-package --importFilePath ImportFilePath --conflictResolutionAction SKIP</code></td>
<td>Compares the package and repository elements, reuses the elements that match from the repository elements, and imports the package. The <code>SKIP</code> option compares only the application, service, and task name and version. This option does not compare properties or action scripts in an application. For example, if you change a property in an existing application and use this option to import the package, the revised property is not imported. In this case, you must use the <code>OVERWRITE</code> or <code>IMPORTASNEW</code> option.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 15-2. CLI Import Commands (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>import-package --importFilePath ImportFilePath</td>
<td>Compares the package and repository elements, overwrites the repository elements that match with the package elements, and imports the package. If a package element does not match the repository element, then the OVERWRITE option creates an element.</td>
</tr>
<tr>
<td>--conflictResolutionAction OVERWRITE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>When an existing logical template in the repository is overwritten by the OVERWRITE option, all of the cloud templates that were assigned to the logical template are removed. After the OVERWRITE operation successfully finishes, you must to remap cloud templates to this logical template.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> The OVERWRITE option does not work if an application, service, or task name in the repository differs only by an upper or lower case letter. For example, if the repository contains an Apache 2.0.0 service, you cannot import a package called APACHE 2.0.0, apache 2.0.0, or ApAche 2.0.0. In this case, change the name of the import package or use the IMPORTASNEW option.</td>
</tr>
</tbody>
</table>
Table 15-2. CLI Import Commands (Continued)

<table>
<thead>
<tr>
<th>CLI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>import-package --importFilePath ImportFilePath --conflictResolutionAction IMPORTASNEW --suffix NewSuffix</td>
<td>Imports the package and repository elements as new elements. The --suffix option renames the element with the specified suffix. It must be used with the IMPORTASNEW option. <strong>Note</strong> If the NewSuffix contains spaces, you must enclose the suffix in double quotation marks. When you import an application version using the IMPORTASNEW option, an application and associated application version is created based on the specified suffix. This option does not create an application version in an existing application. As part of the import process, it also creates a logical template version that you must map to a cloud provider to deploy the newly imported application. For example, to import the Clustered DotShoppingCart application as a new application called test Windows app, use the command import-package --importFilePath ./ClusteredDotShoppingCart 1.0.0.xml --conflictResolutionAction IMPORTASNEW --suffix &quot;test Windows app&quot;. <strong>Note</strong> If you attempt to import a package element twice with the same suffix, the operation fails.</td>
</tr>
<tr>
<td>import-package --importFilePath ImportFilePath --withDeploymentProfilesUsingDeploymentEnvironment DeploymentEnvironmentSpecification</td>
<td><strong>Important</strong> Before you use this command, use the import with CHECK option to review the existing logical templates in the repository. Then use the import with SKIP option to import the package. From the vFabric Application Director user interface, map the logical template to a cloud template. The SKIP option is valid only with this command. When the import command is set to SKIP, all the deployment profiles associated with the exported application versions are imported to the repository. Imports a deployment profile. The DeploymentEnvironmentSpecification includes CloudProviderName: DeploymentEnvironmentName For example, you can use the --withDeploymentProfilesUsingDeploymentEnvironment myCloudProvider:myDeploymentEnv command to import a deployment profile. <strong>Note</strong> When a deployment profile is imported, existing cloud network mappings are removed. In the Deployment Profile wizard, click <strong>Map Details</strong> to map to a cloud network and verify cloud template mapping before you can use the imported deployment profile to deploy an application blueprint.</td>
</tr>
</tbody>
</table>

Complete the tasks provided in the description section, before using this command.

**Important** Before you use this command, use the import with CHECK option to review the existing logical templates in the repository. Then use the import with SKIP option to import the package. From the vFabric Application Director user interface, map the logical template to a cloud template. The SKIP option is valid only with this command. When the import command is set to SKIP, all the deployment profiles associated with the exported application versions are imported to the repository. Imports a deployment profile. The DeploymentEnvironmentSpecification includes CloudProviderName: DeploymentEnvironmentName For example, you can use the --withDeploymentProfilesUsingDeploymentEnvironment myCloudProvider:myDeploymentEnv command to import a deployment profile. **Note** When a deployment profile is imported, existing cloud network mappings are removed. In the Deployment Profile wizard, click **Map Details** to map to a cloud network and verify cloud template mapping before you can use the imported deployment profile to deploy an application blueprint.
Known vFabric Application Director troubleshooting information can assist you in solving common problems.


Table 16-1. Troubleshooting Common vFabric Application Director Errors

<table>
<thead>
<tr>
<th>Error</th>
<th>Possible Cause</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent did not respond while running task agent_bootstrap on the node CentOS_x32_5.6. Please check agent logs. For CentOS logical templates, guest customization does not successfully complete, causing a failure in the agent bootstrap script and the overall deployment fails.</td>
<td>Having more than five NICs on a node in a CentOS virtual machine might cause the problem. The network used for the application deployment does not have connectivity to the vFabric Application Director appliance. VMware Tools is not installed. The vFabric Application Director agent bootstrap service or JRE is not installed properly.</td>
<td>Reduce the number of NICs for an individual node on the CentOS virtual machines. See “Define Multiple NICs for a Node,” on page 88. Check the application deployment network and infrastructure settings. Install VMware Tools on your vCloud template. Verify that the agent bootstrap service or JRE is installed properly on the vCloud template or Amazon AMI.</td>
</tr>
<tr>
<td>An invalid URL error in the Web browser.</td>
<td>The Adobe Flash Player plug-in might not be supported.</td>
<td>Verify that you are running the supported version of the Adobe Flash Player plug-in, in your Web browser. See “vFabric Application Director System Requirements,” on page 18.</td>
</tr>
<tr>
<td>ERROR in ch.qos.logback.core.joran.spi.ConfigurationWatchList@158f9d3 - URL [jar:file:/opt/vmware-appdirector/agent/nobel-agent.jar!/logback.xml] is not of type file</td>
<td>If you deploy a vFabric tc Server on a node with a Linux operating system, an error message mistakenly appears in the log file /opt/vmware-appdirector/agent/logs/agent_bootstrap.log.</td>
<td>You can safely ignore the error message.</td>
</tr>
<tr>
<td>The Zimbra Clustered application does not load but the deployment status was successful.</td>
<td>The zimbra_nfs_path property value might point to an unavailable NFS server.</td>
<td>Add a zimbra_nfs_path to an NFS server that is working and deploy the Zimbra Clustered application. See “Deploy the Zimbra Clustered Application,” on page 120.</td>
</tr>
<tr>
<td>The application name includes special characters causing the deployment to fail.</td>
<td></td>
<td>Rename the application without using special characters. You can add spaces between the letters or numbers.</td>
</tr>
</tbody>
</table>
Table 16-1. Troubleshooting Common vFabric Application Director Errors (Continued)

<table>
<thead>
<tr>
<th>Error</th>
<th>Possible Cause</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>/usr/lib/python2.4/site-packages/Cheetah/Compiler.py: 1508: UserWarning: You don't have the C version of NameMapper installed! I'm disabling Cheetah's useStackFrames option as it is slow with the Python version of NameMapper. You should get a copy of Cheetah with the compiled C version of NameMapper. warnings.warn(</td>
<td>After you deploy the Clustered Dukes Bank application, in the JBoss install and configure log file an error message mistakenly appears.</td>
<td>You can safely ignore the error message.</td>
</tr>
<tr>
<td>[Warning] option 'max_join_size': unsigned value 18446744073709551615 adjusted to 4294967295 [Warning] option 'max_join_size': unsigned value 18446744073709551615 adjusted to 4294967295 OK Filling help tables... [Warning] option 'max_join_size': unsigned value 18446744073709551615 adjusted to 4294967295 [Warning] option 'max_join_size': unsigned value 18446744073709551615 adjusted to 4294967295 OK</td>
<td>If you deploy a MySQL predefined service in a CentOS5.6 64-bit logical template, in the MySQL configure log file an error message mistakenly appears.</td>
<td>You can safely ignore the error message.</td>
</tr>
<tr>
<td>Hyperic Agent fails with a java.io.EOFException error.</td>
<td>The error might appear intermittently in the Execution Plan log file.</td>
<td>Relaunch your application deployment.</td>
</tr>
<tr>
<td>Could not save Application version because another session has modified it.</td>
<td>While an application architect is saving an application blueprint or a deployer is deploying an application, if another application architect attempts to access the same blueprint, the error message appears in the browser of the second application architect.</td>
<td>Click the Refresh button to reload the application blueprint. This might cause you to lose the current changes made to the blueprint.</td>
</tr>
<tr>
<td>Error</td>
<td>Possible Cause</td>
<td>Possible Solution</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| For some users, when registering a new cloud provider a Could not connect to the cloud provider at 192.0.2.255: An error occurred with the cloud provider: peer not authenticated error appears. | The certificate of the cloud provider is signed by a certificate authority that is not in the openssl trusted list of the vFabric Application Director server. | Export and save the certificate file of the vCloud Director server from a supported Web browser. If you are using the Firefox browser, save the top level certificate authority and all of the intermediary certificate authorities.  
1 Import the certificate to the vFabric Application Director appliance.  
2 From the command prompt, log in as root and add the certificate file to the vFabric Application Director appliance trusted list.  
   keytool -importcert -trustcacerts -alias UniqueAlias -file CertFilePath.crt -storepass "" -keystore /home/darwin/keystore/appd.truststore  
3 Restart the vFabric Application Director server.  
sudo service vmware-darwin-tcserver restart |
| An unknown error has occurred: 500 Internal Server Error. | The LDAP configuration name might include a period. For example, in ldap.test.com, the period in the configuration name is not supported.  
Duplicate LDAP user or group names might exist among different users or groups.  
The LDAP user or group name has the same UID in two different LDAP configurations. | Rename the LDAP configuration without any periods and delete the LDAP user.  
Log in to the LDAP server and check whether the group name or user name are unique.  
Determine whether your LDAP configuration is created correctly. See “Create and Activate an LDAP Configuration,” on page 33. |
| Retrieval of user information from the LDAP server failed. Error messages follow: [ERROR] Incorrect result size: expected 1, actual 2 | When you run the command to check the existence of a user name in the LDAP directory to confirm server connectivity, you might not have specified the User Search Base DN. | Specify the User Search Base DN in the LDAP command. In the sample command,  
test-named-ldap-config --configname test-slave.test.com --name user1 user1 specifies the User Search Base DN. |
<p>| Retrieval of user information from the LDAP server failed. Error messages follow: [ERROR] Root DNs must be the same when using multiple URLs | When you run the command to check the existence of a user name in the LDAP directory to confirm server connectivity, you might have added a space in the Server Base DN. | Remove the space in the Server Base DN. |
| While initiating an update process to scale out a clustered node, in the execution plan page of the wizard, when you attempt to add a task to a clustered node, you are prompted to expand the node. If you resize the information panel, you might not be able to add a custom task to the execution plan. | The information panel which is in HTML 5 code and the Flex code are not properly interacting. | Click the X icon to close the information panel and add a custom task to the clustered node. |</p>
<table>
<thead>
<tr>
<th>Error</th>
<th>Possible Cause</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typing the vFabric Application Director Web interface URL without HTTPS renders a blank page in the Web browser.</td>
<td>The cause of this problem is unknown.</td>
<td>Change HTTP to HTTPS in the URL. (Optional) Change the port number in the URL to 8080: <a href="https://ApplicationDirectorIP">https://ApplicationDirectorIP</a>: 8080/darwin.</td>
</tr>
<tr>
<td>When you type the status command, it shows that you are logged in. However, you receive an error, Your session has expired or been invalidated, please login again, when you use the CLI.</td>
<td>The CLI session has timed out.</td>
<td>Log out of the CLI and log in to resume.</td>
</tr>
<tr>
<td>When you run a Batch file using a PowerShell script in vFabric Application Director, the script might not run but the task completes successfully.</td>
<td>The PowerShell script might need an expression to run successfully.</td>
<td>Add an <code>invoke-expression</code> expression to the PowerShell script. For example, to start and stop a Windows vFabric tc Server add <code>invoke-expression $service_stop</code> and <code>invoke-expression $service_start</code> expressions to the script.</td>
</tr>
</tbody>
</table>
Index

Symbols
$(random) 83

A
access key 50
actions
component 59
life cycle stages 59
activate LDAP configuration 33
Active Directory domain 43, 122
add, custom task 77, 94
add Windows template 44
advanced application blueprint 86
agent bootstrap service requirements for custom templates 39
Amazon EC2
cloud providers 54
deployment profile 91
linking to as a cloud provider 49
network 91
Amazon EC2 AMI 46
Amazon EC2 templates 54
AMI 46
Application Director
disk space requirements 18
memory requirements 18
ports opened 18
See also vFabric Application Director
Application Director shuts down 25
applications
creating 83
deleting virtual machines from the cloud 131
deploying 91, 96
deployment details 129
deployment profile 91, 94, 97
quick deploy 97
APT Repository Config 124
architectural principles 10
Auto-Bind consume 65, 73
Auto-Bind expose 65, 73
Auto-Bind tag 65, 73
Auto-Bind type 65, 73
Availability Zone 54, 57

B
basic application blueprint 83
binding properties 64
blueprint for an application 83, 86
boot for an application 22

C
cancel
deployment 132
update process 132
catalog, Application Director 11, 69
certificate
import 34
LDAP 34
local 34
SSL 34
check list to set up vFabric Application Director 15
CLI (command-line interface)
creating groups with 32
creating users with 32
deploy 146
deployment status 146
export 151
general commands 141
import 151, 153
managing cloud tunnels 145
managing LDAP configurations with 143
managing users with 142
quick deploy 146
tear down deployment 149
update profile 146
cloud providers, register 52, 54
cloud templates 69
cloud tunnel 49
cluster node 64
command-line interface 141
configure
Amazon EC2 50
component properties 61
DHCP 43
Linux template 44
vFabric Application Director 55
Windows template 41
configuring network options 20
configuring storage options 20
configuring vCloud Director 19
connect Amazon EC2 55
connection problems 24
copy
  application 89
  application version 89
create
cloud tunnel 55
Linux template 44
user groups 141
Windows template 41
create dependency 87
create LDAP configuration 33
creating groups with the CLI 32
creating users with the CLI 32
custom task
  copy 79
  delete 79
  edit 79
custom virtual machine templates 39

D
Default group 35
define, property values 63
define multiple NICs 88
delete
  application deployments from Application Director 131
  application version 90
  applications from the cloud 131
deploy
Apache Hadoop 116
Clustered DotShoppingCart 114
Clustered WebSphere Application Server 110
jPetStore 118
Nanotrader 112
Radiant CMS 121
SharePoint Server 2010 117
Spring Travel 118
WebLogic Clustered Deployment 113
Zimbra 119
Zimbra Clustered 120
deploy Clustered Dukes Bank App 115
deploy sample application 110, 112–121
deployment environment 56, 57
deployment of applications 91
deployment process 103
deployment profile 91
Deployment Profile wizard 96
deployment status 104
deployment summary page 104
deployments 129, 131
develop components, catalog 59
developing components, best practice 68
download agent bootstrap 41, 44

e
Elastic IP address 50
enable
  Join Domain 43
  SID 43
Endpoint VM 49, 57
environment variables 61
execution plan 96, 133
existing deployment profile 94
export
  application 151
  custom tasks 151
  service version 151
templates 48

F
firewalls, accessing URLs outside the firewall 25

G
groups 27

H
host name 83, 91

I
import
  CHECK option 153
  IMPORTASNEW option 153
  SKIP option 153
import deployment profile 153
install agent bootstrap 41, 44
installation prerequisites 17
installing vFabric Application Director 17

J
Join Domain 122
JRE requirements for custom templates 39

L
licensing 22
locked account 24
log files for troubleshooting 133
logging in 35
logical templates
  copy 71
  delete 72
  edit 71
  virtual machine 69
Index

M
manage, user groups 141
manage deployments 129
manage LDAP configuration 143
manage users with the CLI 142
managing cloud tunnels 142
modify configuration 100
multicloud 10
multiple hosts in a cluster 20

N
network bandwidth 55

O
OOB (out-of-the-box), services 11, 73
operating systems, for custom templates 39
operating systems in the vFabric Application Director catalog 80
out-of-the-box users, groups, and roles 30

P
passing property values 66
predefined
   IP Address property 66
   node array property 67
predefined properties 66
predefined service 73
predefined users, groups, and roles 30
predefined users and roles 27
profile, application deployment 91, 94
   properties
      array 61
      computed 61
      content 61
      required 67
      secured component 67
      string 61
proxy configuration 25

R
REST API 141
restart Application Director 25
RHN Registration 126
roles 27
run ID for a deployment 104

S
sample
   applications 109, 110
   catalog services 128
   CentOS template 109
   services 109
tasks 109, 122, 124–126
template 109
Ubuntu template 109
   virtual machine templates 109
scale deployments 98
script
   Bash 60
   PowerShell 60
   Windows CMD 60
search criteria 36
security settings 11
service
   copy 76
   delete 77
   edit 76
services, adding 73
set host name 83, 91
set up, cloud providers 49
shut down appliance 22
single host in a cluster 20
single node 64
single-tier application 110
special update property 98–100
specify clustered node 87
SSH connection, Ubuntu VM 110
start appliance 22
start CLI program appliance 30
   remote 31
supply the correct password 24
supported action script 60
supported custom task script 60
system requirements for Application Director 18

tags in the vFabric Application Director catalog 81
task status 104
task status window 104
task timeline 104
tear down an application 131
templates
   OVF format 48
   register 52
virtual machine 10, 46, 54
termology 12
three-tier application 110
troubleshoot
   update errors 139
   update process 102, 103
troubleshooting, common deployment errors 134
troubleshooting common errors 157

U
understand deployment process 103
update process 103
unlock your darwin_user account 24
update application deployment 97
change code 100
change configuration 100
scale out 98
update deployment 104
update existing VM template 48
update Linux VM template 48
update process 103, 129
update Windows VM template 48
upgrading vFabric Application Director 26
user accounts 27
user interface
buttons 37
icons 38
users, groups and roles 30

V
vCloud Director
cloud providers 52
linking to as a cloud provider 49
vCloud Director catalog 44
vCloud Director templates 52
verify vCloud Director setup 20
vFabric Application Director
applications 83, 86
catalog 69
custom task 77
deploying the Application Director vApp 21
deployment profiles 91
logical templates 69
overview of 9
services 73
setup 15
terminology 12
Web interface 35
virtual datacenters (vDCs) 56
virtual machine
Endpoint 55
Linux 50
NAT 50
virtual machine template, Amazon EC2 46
virtual private cloud (VPC) 54, 57
VM, Endpoint 51

VMware Tools 41, 44
VMware Tools requirements for custom templates 39
VMware vFabric Application Director for Provisioning, edition 22
VMware vFabric Application Director for Release Automation, edition 22

W
Web interface for vFabric Application Director 35
Web user interface 36
Windows, virtual machine template 41, 43, 44

Y
YUM Repository Config 125