

HP Continuous Delivery Automation

Software Version 1.00

User Guide

Document Release Date: June 2012

Software Release Date: June 2012



Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 2012 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

RED HAT READY™ Logo and RED HAT CERTIFIED PARTNER™ Logo are trademarks of Red Hat, Inc.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

<http://h20230.www2.hp.com/selfsolve/manuals>

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

<http://h20229.www2.hp.com/passport-registration.html>

Or click the **New users - please register** link on the HP Passport login page. You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: www.hp.com/go/hpsoftwaresupport. This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

Search for knowledge documents of interest

Submit and track support cases and enhancement requests

Download software patches

Manage support contracts

Look up HP support contacts

Review information about available services

Enter into discussions with other software customers

Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport user ID, go to: **<http://h20229.www2.hp.com/passport-registration.html>** To find more information about access levels, go to: **http://h20230.www2.hp.com/new_access_levels.jsp**

Contents

About this Guide	8
1 Introduction	9
HP Continuous Delivery Automation Overview	9
The Application Release Process	9
Model Driven Approach	9
Extensible, Pluggable Architecture	10
2 HP CDA General Information and Tasks.....	12
Parameters and Parameterization	12
Parameters.....	12
Adding Parameters	12
Stored Parameter Sets	15
Saving Parameter Values to a Stored Parameter Set.....	15
Creating or Editing a Stored Parameter Set.....	15
Using a Stored Parameter Set during Provisioning or Deployment.....	16
Substitution Variables.....	16
Substitution Variables in Workflow Actions	16
Substitution Variables in Fields	17
Parameter Referencing	18
Referencing Parameters via Reference Variables.....	21
Provisioning Platforms	23
Provisioning a Platform.....	23
Viewing Provisioned Platforms.....	25
Deprovisioning a Platform	25
Deploying Applications	26
Deploying an Application.....	26
Viewing Deployed Applications	28
Undeploying an Application	29
Cancel, Backout, and Forced Cleanup Operations.....	30
Cancelling Provisioning or Deployment Operations	30
Backing Out of Failed Operations	31
Forced Cleanup of Failed Operations	33
Archive Management.....	34
Exporting Single Models to a ZIP Archive File	34
Exporting Multiple Models to a ZIP Archive File	35
Importing Model Archives.....	36
3 Application Development.....	38
Application Creation & Management	38
Model Driven Approach.....	38
Application Model.....	38

Application Lifecycle Workflow	38
Application, Application Version, & Application Model	39
Managing the Application Model.....	39
Viewing Applications.....	40
Creating an Application and Application Version.....	42
Software Artifacts	43
Defining Software Artifacts	44
Adding Software Artifacts.....	46
Using the Wizard to Add Artifacts	47
Using the Drop-down List to Add Artifacts	48
Application Model Components	49
Application Model Tabs	50
“Overview” Application Model Tab	50
“Layers” Application Model Tab.....	51
“Deployment Topologies” Application Model Tab	52
“Deployments” Application Model Tab	53
“Parameters” Application Model Tab	53
“More” Application Model Tab	54
Application Model Actions	54
Editing an Application Model	55
Cloning an Application Model	55
Changing the Domain for an Application Model.....	56
Defining an Application Layer.....	57
Application Workflow	58
Workflow Actions & Programming Operations.....	58
Accessing an Application Workflow	59
Configuring a Workflow Layer Action.....	59
Configuring a Workflow Backout	61
Input Component Parameters	63
Adding an Input Component Parameter to a Workflow	63
Adding an Input Component Parameter to a Layer	65
Adding a Parameter Definitions Group to an Application Model.....	66
Connections & Endpoints.....	66
Creating, Modifying, & Deleting Endpoints.....	66
Defining and Deleting Connections.....	67
Deployment Topology	67
Modeled Relationships	67
Deployment Topology and Load Balancing.....	68
Editing a Deployment Topology.....	69
Adding a Deployment Topology.....	70
Cloning a Deployment Topology	72
Deleting a Deployment Topology.....	73

4	Platform Development	74
	Platform Modeling Basics	74
	Infrastructure Template Management	75
	Infrastructure Templates – Important Information	75
	Manually Synchronizing IO Infrastructure Templates with HP CDA	75
	Editing an IO Infrastructure Template from HP CDA	76
	Creating Infrastructure Templates Using the Existing Infrastructure Provider	77
	Creating a Pet Clinic Infrastructure Template Using the Existing Infrastructure CLI Example	78
	Platform Software Management	79
	Understanding Platform Software Models (Overview)	79
	Workflows and Programming Actions	80
	Displaying Platform Software	91
	Creating Platform Software	91
	Create the Platform Software Object	91
	Platform Model Management	96
	Creating a Platform Model from a Template	96
	Managing Platform Properties Using the Platform Designer	98
	Displaying the Platform Designer	100
	Adding Connections	103
5	Monitoring Administration	105
	Monitoring Administration Overview	105
	Step 1: Identify the monitoring tools in your environment and create Monitor Servers	106
	Step 2: Define Monitors.	115
	Importing SiteScope Monitors	115
	Manually configuring monitors	117
	Step 3: Define monitoring policies	118
	Step 4: Assign Monitoring Policies to Application Topologies.	119
	Remonitoring	120
	Deploying and Undeploying Applications with Monitors	121
	Step 5: Review the state of monitored applications.	121
	Monitoring Applications in Amazon Elastic Compute Cloud	122
	HP Diagnostics and Amazon EC2	122
	HP SiteScope and Amazon EC2	123
	Establishing SSH Tunnels	123
	Using SSH Tunnels with HP Diagnostics	124
	Using SSH Tunnels with HP SiteScope	124
	Using SSH Tunnels with Nagios	125
6	HP CDA System Administration	127
	Administration Overview	127
	Security and Access Control	128
	Tenant Management	129

Domain Management	129
Domains Overview	130
Creating and Deleting Domains	132
Managing Default Access Rights	134
Exporting Domain Content	136
Disabling Default Domain Sign-In	137
User Management	138
Disabling and Enabling Users	138
Assigning Users to Groups and Roles	139
Assigning Users to Roles:	139
Setting Default Domains for Users	139
Setting New Artifact Ownership	140
Managing Groups	141
Creating Groups	141
Managing Group Membership	141
Assigning Groups to Additional Domain Roles	142
Setting Default Domains for Groups	142
Retiring and Deleting Groups	142
Managing Roles	143
Roles Overview	143
User Interface Roles	144
Lifecycle Roles	145
Managing Roles	146
Changing the Sharing Principal	146
Create Role Page	147
Policy and Assertion Management	147
Managing Technical Policies	148
Default Technical Policies	149
Tag Management	150
Adding, Deleting, or Modifying a Tag	150
Provider Integration	151
Defining Servers	151
Integrating Providers Using Plugins	152
Testing a Plugin Prior to Updating	153
Predefined Deployer Types for Plugins	153
Updating Plugins	153
Uploading New Plugins	154
Defining a Deployer Plugin	155
Lifecycle Process Management	157
Creating Lifecycle Processes	157
Defining Lifecycle Stages	158
Defining Lifecycle Transitions	159
Defining Lifecycle Tasks	159
Defining Lifecycle Policies	160
Defining Lifecycle Approvers	160
Defining Lifecycle Automatic Actions	161
Defining Lifecycle Permissions	161
Publishing a Lifecycle Process	162
Exporting a Lifecycle Processes	162

Lifecycle Best Practices	162
Configuration Management	163
Managing Basic Configuration Options	163
Create Empty Data Artifacts	164
SSL Customization	164
License Management	164
Changing the License	164
HP CDA Self-Test	165
Managing System Settings	166
Managing Artifact Form Validation	166
Administration Task Management	167
Running Tasks	168
Adding Change Management Tasks	168
Adding Custom Tasks	169
External System Integration Using the Command Line Interface	169
CLI Parameters and Options	170
Model Commands	170
Job Commands	170
Existing Infrastructure Commands	170
A The HP CDA Command Line Interface	172
B HP Diagnostics Probe Deployment	213
Import the platform software for installing Diagnostics probes	213
Configure the platform software to point to the HP Diagnostics distribution	213
Configure the logical platform to attach the platform software	214
C For More Information	215

About this Guide

This guide explains how to install and configure the HP Continuous Delivery Automation (HP CDA) software. HP highly recommends that each chapter be read and all procedures be performed in the order presented in this guide, for successful results. This guide contains the following chapters:

- Chapter 1: This chapter provides an overview of HP CDA.
- Chapter 2: This chapter presents general information and tasks that are important for using HP CDA.
- Chapter 3: This chapter contains information for developing application models.
- Chapter 4: This chapter contains information for developing platform models.
- Chapter 5: This chapter contains information related to HP CDA's monitoring features.
- Chapter 6: This chapter contains HP CDA system administration information and tasks.

This guide also contains these additional sections:

- Appendix A: This appendix explains the available HP CDA CLI commands.
- Appendix B: This appendix contains information necessary to deploy HP Diagnostics probe software onto target servers so that HP Diagnostics can be used to monitor the servers.
- Appendix C: This appendix lists other HP publications that are available for the HP CDA product.

1 Introduction

HP Continuous Delivery Automation Overview

HP CDA provides a model driven approach to DevOps collaboration, automation of application deployment, and monitoring. HP CDA enables customers to focus on their core applications and to drive business values while reducing costs, risks, and time to perform provisioning and deployment tasks. Core HP CDA features include:

- Modeling the application and infrastructure configuration to deliver infrastructure-as-code for deployment and provisioning.
- Full artifact version control, role-based access, application life cycle management, and Definitive Software Library (DSL)
- Application deployment management using configurable tool options
- Infrastructure provisioning management across hybrid environments
- Embedded monitoring deployment in conjunction with application deployment

The Application Release Process

To bridge the gap between dev and ops environments, a new level of organizational coordination and communication is needed. The requirement is for a defined pipeline of release stages and application candidates moving regularly from planning to development and from release to operation. This ensures application visibility to the developers when their applications are pushed into production.

Continuous delivery results in a dynamic set of data and actions to turn the data into realized deliverables. Data can be contained in documents, checklists, and wikis, but the realization of this data is susceptible to human error or interpretation. There are many enterprise tools and products available to enact the process – from provisioning systems to deploying applications and monitoring, but without automation, standardization of tools and environments becomes hard to control. Delivery into a hybrid environment can magnify these inconsistencies when tools and processes must be managed both inside and outside the organization. HP CDA creates a centralized structure for implementing a DevOps agenda and standardizes the integration of tools that are best suited to drive the continuous delivery processes. HP CDA delivers this capability through the following methods:

- Model driven approach
- Extensible, pluggable architecture

Model Driven Approach

To provide flexibility and portability of applications across platforms through continuous delivery, HP CDA provides a data model that represents the artifacts and relationships required to define how composite applications are delivered to a dynamic set of target environments. This model driven approach enables HP CDA to centrally store and version artifacts in a definitive software library and then orchestrate the requisite operations, which leverage the models for automating tasks such as platform provisioning and application deployment. HP CDA defines several key models and relationships:

- **Platform:** the compute, storage, network, and additional software capability required to fully define the target environment (infrastructure) for an application. HP CDA integrates with HP Matrix Operating Environment (HP MOE) to provide the capability to provision Infrastructure as a Service (IaaS).

- **Application:** the build artifacts, scripts, and commands per layer required to deploy the application, including the basic operations (start, stop, and so on) that can be performed on the application.
- **Policy:** the monitoring providers and monitor templates as well as the values and parameters to automate the configuration of system and application monitoring during deployment
- **Topology:** the dynamic relationship between composite application layers and compute resources within a platform, which defines a unique deployment mapping.

This model driven approach provides flexibility and portability of applications across platforms. It also allows for dynamic reuse of artifacts to meet the delivery requirements. This translates into value as follows:

- Improved Time-To-Delivery (TTD) of application ecosystems, and associated reduction in costs, errors, rework, and risks by having model driven automation.
- Improved collaboration between organization teams (dev, test, operations) by centralizing information about setup, operations, and monitoring across the release pipeline. The CDA model becomes a collaborative library, which reduces rework of processes and reduces errors.
- Reduced vendor lock-in by separating the model that defines the delivery artifacts from the tools that operate on the model to realize the automation.

Extensible, Pluggable Architecture

HP CDA provides the flexibility to design continuous delivery around the tools and environments prescribed by an organization, as there is no one delivery environment that fits all businesses. This allows you to rapidly adopt a DevOps paradigm, as you do not need to re-tool the organization and, at the same time, you can maintain the organizational knowledge during the transformation. By allowing tools such as software deployers and application monitors to be configured and changed dynamically, HP CDA provides a future proof method for continuous delivery processes that you require.

The architecture of HP CDA provides configuration of delivery providers in several key areas:

- Software deployment:
 - HP CDA models for applications can be translated into sequences of steps for leading products used for software deployment such as HP Server Automation and Chef. This allows the same application model to be deployed to target platforms through different deployers depending on the user configuration.
 - Many leading software deployment technologies have evolved significant libraries from off-the-shelf content either within organizations or in public communities. HP CDA can leverage existing deployment content directly as extensions to both application and platform models.
- Monitoring:
 - After infrastructure is in place and applications are deployed, monitoring becomes the focus. With many available tools, customers may use any of a number of monitoring tools such as HP SiteScope, HP Diagnostics, or Nagios. HP CDA monitoring policy models abstract vendor specifics, but provide the plug-in layer to configure specific tools based in the model.
 - By normalizing and centralizing monitor policy definitions in a central library, monitoring is no longer an activity confined to production. The portability of policies across release pipeline environments provides a well-defined life cycle of performance management.
- Hybrid infrastructure:
 - IT organizations use an array of private and public cloud technology and providers as well as traditional physical infrastructure. This heterogeneity creates the demand to make applications

easily portable across technologies, and may also be driven by organizational boundaries or investment decisions.

HP CDA provides an extensible infrastructure layer for provisioning systems in environments such as HP Cloud Services, Amazon Elastic Compute Cloud, and VMWare.

2 HP CDA General Information and Tasks

Parameters and Parameterization

Parameters and parameterization is a feature of HP CDA that allows platform software, logical platform, application, and monitoring model developers the ability to design flexibility into their models.

Note: This section refers to such concepts as platform, monitoring, and application models and modeling. Subsequent chapters in this User Guide present those concepts in detail.

An important aspect of modeling platforms and applications using HP CDA is the ability to provision and deploy such models into different environments. This is done by specifying parameter values that are specific (customized) to the environment to which the platform is to be provisioned or the application is to be deployed.

Once the model developer determines the information that needs to be customizable for different environments, parameters can be added to the models and *substitution variables* can be placed at different points in the model, such as in files to be placed or scripts to be executed on the target machine. Such substitution variables then will receive their values from the parameters with which they are associated. Customized values for parameters are typically set right before a provisioning or deployment operation is started, and parameter substitution occurs either right at the beginning or during the provisioning or deployment operation, depending on the *parameter type* used.

Parameter value customization can be accomplished either manually, or through the use of previously-saved *stored parameter sets*.

Parameters

Parameters are named objects, created by a model designer/developer, to hold a value which can then be substituted into various points in the model via substitution variables. A parameter's value can come from the following:

- A default value that had been assigned to the parameter when it was created.
- A value manually entered by a user before a provision or deployment operation is started.
- A value from a *stored parameter set*.
- The value of another parameter that has been referenced from the parameter via a *reference variable*. Such references to reference variables are resolved automatically by HP CDA when a workflow is initiated.
- Via automatic resolution during runtime. For example, a reference to a certain reference variable type called a *per-server variable* is resolved at the point in time when a placed file is deployed or an executed script is run on a target host.

Adding Parameters

Model designers add parameters to customize model components.

To add a parameter, complete the following steps:

1. Navigate to the pertinent location in the HP CDA interface where you will be adding the parameter. The location to which you will navigate depends upon the portion of the model for which you need to

define parameters. Table 1 lists the locations in the HP CDA interface to where you need to navigate in order to add parameters for various models or portions of models.

Table 1 Parameter Creation Interface Locations

Model/Portion	Parameter Creation Location
Application model	"Application Model" screen ⇒ Parameters tab.
Application layer	"Application Model" screen ⇒ Layers tab ⇒ pertinent layer section ⇒ "Parameters" tab.
Platform software	"Software" screen ⇒ Parameters tab.
Platform Software and Application workflow actions	"Input Component Parameters" section of the pertinent workflow action screen.
Monitors	"Monitor" screen ⇒ Overview tab ⇒ "Additional Parameters" section.
SiteScope templates	"SiteScope Template" screen ⇒ Overview tab ⇒ "Additional Parameters" section.
Tags	"Input Component Parameters" section of the pertinent "Tag" screen.

2. Click the **+Add** icon in the parameter creation location

The "Add" parameter dialog box displays.

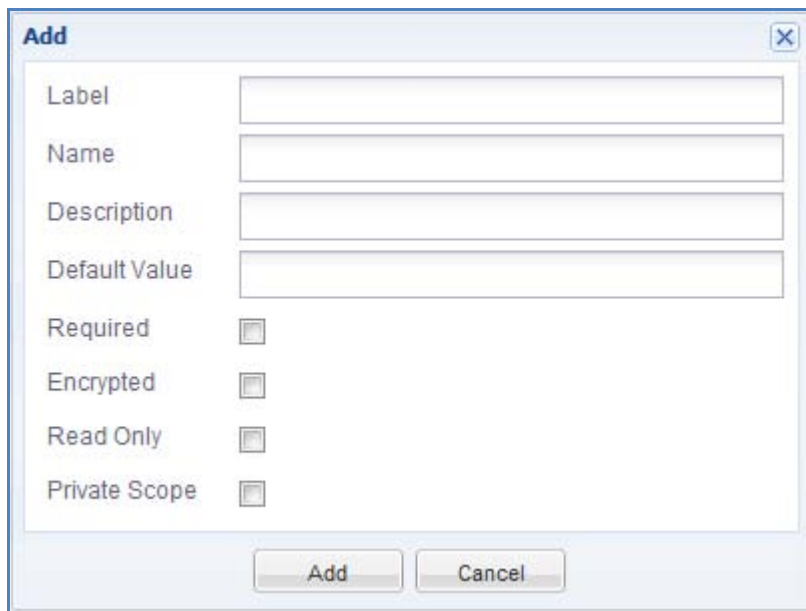


Table 2 provides a description of each field in the "Add" parameter dialog box.

Table 2 Add Parameter Dialog Box Field Descriptions

Field	Description
Label	The label for the parameter. This label is used in the HP CDA user interface only to identify it and is not the actual name of the parameter used in references and substitution variables. When a label is entered here, the "Name" field is filled out automatically, with upper-case letters being replaced by lower-case characters, and spaces replaced by periods. Unlike the parameter name ("Name" field), this field can be changed after the parameter has been saved.

Field	Description
Name	<p>The name of the parameter. It is the entry into this field that becomes the character-for-character string that is used in references and substitution variables. As mentioned under "Label," this field is filled in automatically when the "Label" field is filled in, but it can be changed <i>before</i> the parameter is saved. After the parameter has been saved, you can no longer change the name directly, so you will need to delete the old parameter and re-create one with the new name.</p> <p>Note: If you do re-create a parameter with a new name in order to rename it, you will also need to change the parameter's name in places where it is referenced, such as in substitution variables.</p> <p>Parameters within the same group of parameters cannot have the exact same name; however, parameter names are case-sensitive. For example, two parameters within a group of parameters cannot have the name "my.parameter," however, a parameter named "my.parameter" and another named "my.Parameter" can exist in the same group. Such restrictions do not apply to parameters having the same name that are used in different groups of parameters.</p>
Description	<p>An optional description for the parameter.</p>
Default Value	<p>The default value of the parameter, which can be one of the following:</p> <ul style="list-style-type: none"> • Blank, if no default value if desired. • A textual value. • A reference to another parameter. <p>A textual value can be entered directly into the field. To reference another parameter, click the ellipsis icon (...) that appears to the right of the field, and select the appropriate reference variable from the "Reference Variables" dialog box that appears.</p> <p>Note: Refer to the section entitled "Parameter Referencing" for more information of parameter referencing and reference variables.</p> <p>If the parameter to be referenced has not been created yet when you create a parameter that will reference it, you will need to subsequently edit the referencing parameter to include the reference once the parameter to be referenced has been created.</p>
checkboxes	<ul style="list-style-type: none"> • Required – Select this checkbox if a value for the parameter will be required to be entered or included when the parameter value is called for, for example, when the "Deploy Application" Wizard runs. A required parameter cannot be left blank, but its value can be entered manually, via reference, or via a default value. • Encrypted – Select this checkbox to mask the parameter value where it appears or is entered into user interface screens. • Read Only – Select this checkbox if the parameter value is to be un-editable in user interface screens other than the "Add" or "Edit" parameter dialog box. • Private Scope – Select this checkbox to limit the scope of the parameter to within the area where it was created. A parameter with a private scope might be listed, for example, on the "Parameters" tab of the "Application Model" screen, but not appear in the "Deploy Application" Wizard when it runs.

3. Enter information in the Add parameter dialog box.
4. Click the **Add** button in the "Add" parameter dialog box to save the parameter and close the dialog box.

Stored Parameter Sets

The values of “user-exposed” parameters can be customized either manually or through the use of a stored parameter set.

When a provision or deploy operation is initiated via the HP CDA interface, either the “Provision Platform” dialog box or the “Deploy Application” wizard will open. Within those interface elements, the user is presented with the “user-exposed” parameters that have been designed into the model. Those are the parameters that allow customization of the platform or application for different environments.

Such parameter values can be saved to a stored parameter set so that they can be recalled at a subsequent time when deployment to the same environment is needed. Stored parameter sets are also useful for non-interactive provisioning and deployment, such as when HP CDA is integrated and used with HP Application Lifecycle Management (HP ALM). Stored parameter sets can also be created without opening the “Provision Platform” dialog box or the “Deploy Application” wizard. These methods are explained below.

Saving Parameter Values to a Stored Parameter Set

To save parameter values to a stored parameter set, complete the following steps:

1. Initiate the provisioning or deployment operation, as is desired, to invoke the “Provision Platform” dialog box or “Deploy Application” wizard, respectively.
2. Customize the values of the parameters as is necessary for the environment that the model is to be provisioned or deployed into.

Note: The provisioning and deployment processes are explained in detail elsewhere in this chapter.

3. Click the **Save Parameters** button in the dialog box or wizard.

Note: The **Save Parameters** button is in the last screen of the “Deploy Application” wizard.

4. In the **Enter stored parameters entry name** field that pops up, enter a name for the stored parameter set and then click the **Save** button.

The message “New parameters entry ‘<name>’ was saved successfully” appears to indicate that the stored parameter set has been saved.

5. At this point, you can either follow through with the provisioning or deployment operation by clicking the **OK** or **Finish** button as is applicable, or click the **Cancel** button to not follow through. Either way, the stored parameter set has been saved.

Creating or Editing a Stored Parameter Set

To create or edit a stored parameter set, complete the following steps:

1. Navigate to the “Platform” or “Application Model” screen in the interface, as is applicable, for the model for which you want to create a stored parameter set.
2. Click the **Stored Parameters** tab in the screen to invoke the listing of stored parameter sets.
3. To create a new stored parameter set, click the **New** button that is above the list. To edit an existing stored parameter set, click the [Edit](#) link that is to the far right of the stored parameter set in the list.
4. Fill in the fields in the “New Configuration” or “Edit Configuration” dialog box as is necessary, and then click the **OK** button to save the values and close the dialog box.

Using a Stored Parameter Set during Provisioning or Deployment

To use a stored parameter set during provisioning or deployment, complete the following steps:

1. Initiate the provisioning or deployment operation, as is desired, to invoke the “Provision Platform” dialog box or “Deploy Application” wizard, respectively.
2. Click the **Load Parameters** button, and select the name of the stored parameter set to load from the drop-down list that appears.

Note: The **Load Parameters** button is in the second screen of the “Deploy Application” wizard.

3. Finish the provisioning or deployment operation, as planned.

Substitution Variables

A substitution variable allows the value of a given parameter to be substituted into a certain point or points in a model during provisioning or deployment.

Substitution variables may be used in executed scripts and placed files. Such substitution variables are replaced with the values of their referenced parameters just prior to, for example, the execution of a workflow action. Certain fields in some of the interface screens also support the use of substitution variables.

Substitution variables have the following pattern or format:

@{name.of.parameter}

Where “name.of.parameter” is the name of the parameter (not the parameter’s label) for which the value will be substituted.

Substitution Variables in Workflow Actions

You can use substitution variables in an Executed Script workflow action. When you create those variables, they are shown in the Input Component Parameters portion of the window. The names of those parameters (as they appear within substitution variables in the script) are highlighted in yellow in the following graphic. When you run the Executed Script workflow action, the substitution variables will be replaced by parameter values.

The screenshot shows a configuration window for a workflow action. The top section, titled "Content", contains a shell script with several substitution variables highlighted in yellow. The script is as follows:

```
#!/bin/sh
mysql -e "CREATE DATABASE IF NOT EXISTS @{mysql.dbname};" -u @{mysql.localuser.name} -p @{mysql.localuser.password}"
mysql -e "GRANT ALL ON @{mysql.dbname}.* TO @{mysql.remoteuser.name}@% IDENTIFIED BY '{mysql.remoteuser.password}';" -u @{mysql.localuser.name} -p @{mysql.localuser.password}"
exit 0
```

The bottom section, titled "Input Component Parameters", contains a table with the following data:

Name	Label	Default Value	Required	Encrypted	Read Only	Prints Scope	
mysql.dbname	mysql.dbname	Ref->\${END...}	Yes	No	No	Yes	
mysql.localuser.name	mysql.localuser.name	Ref->\${LAYR...}	Yes	No	No	Yes	
mysql.localuser.password	mysql.localuser.password	*****	Yes	Yes	No	Yes	
mysql.remoteuser.name	mysql.remoteuser.name	Ref->\${MOD...}	Yes	No	No	Yes	
mysql.remoteuser.password	mysql.remoteuser.password	*****	Yes	Yes	No	Yes	

You can use substitution variables in a Placed File workflow action in a similar way. For example, you can design a file to include the necessary substitution variables, name it “my.cnf” and save it to DSL. When a workflow action executes, the substitution variables in the file will be replaced by parameter values when the file is placed on a target server.

Source:
 my.cnf Upload Change Delete

Destination: *

Content: Parameterized

Input Component Parameters

+ Add Remove Save as CSV

<input type="checkbox"/>	Name ▲	Label	Default Value	Required	Encrypted	Read Only	Private Scope	
<input type="checkbox"/>	exposed.appli...	exposed application layer workflow act...		No	No	No	No	
<input type="checkbox"/>	mysql.bindad...	mysql.bindaddress	Ref->\${ENDP:...	Yes	No	No	Yes	
<input type="checkbox"/>	mysql.port	mysql.port	Ref->\${ENDP:...	Yes	No	No	Yes	

“Placed File” workflow actions contain a **Content Parameterized** checkbox that, when selected, causes HP CDA to search for and replace the substitution variables when needed. If the checkbox is not selected, HP CDA will forgo the search-and-substitute operation. The checkbox is necessary because the search-and-substitute takes a lot of processing time, and will occur otherwise even if no substitution variables are included in the placed file.

Substitution Variables in Fields

Substitution variables can also be used in certain fields of some of the interface screens. The fields listed below can contain substitution variables:

- In the “Placed File” workflow action interface:
 - “Destination” field
 - “Group” field
 - “Owner” field
- In the “Executed Script” workflow action interface:
 - “Path” field
 - “Precondition” field
 - “User” field
 - “Working Directory” field

Substitution Variables in Inline Strings

HP CDA supports the use of substitution variables in inline strings that are resolved automatically by HP CDA at the appropriate time into a concatenated string. An example of this is as follows:

```
@{protocol}://@{ApplicationIP}:@{port}/@{path}
```

HP CDA will then replace the substitution variables in the string with the values of the parameters named in the substitution variables.

Such inline strings can be included in executed scripts, placed files, and in fields that support the use of substitution variables (see “Placed File” workflow actions contain a **Content Parameterized** checkbox that, when selected, causes HP CDA to search for and replace the substitution variables when needed. If the checkbox is not selected, HP CDA will forgo the search-and-substitute operation. The checkbox is necessary because the search-and-substitute takes a lot of processing time, and will occur otherwise even if no substitution variables are included in the placed file.

Substitution Variables in Fields on page 17). These inline strings can also be used as a default value in the “Value” field of parameters.

Input Component Parameters							
+ Add X Remove Save as CSV							
<input type="checkbox"/>	Name	Label	Default Value	Required	Encrypted	Read Only	Private Scope
<input type="checkbox"/>	db.host	db host	Ref->\${server.hostname}	Yes	No	No	No
<input type="checkbox"/>	db.path	db path	CDA	No	No	No	No
<input type="checkbox"/>	db.port	db port	3306	Yes	No	No	No
<input type="checkbox"/>	jdbc.url	jdbc url	jdbc:mysql://@{db.host}:@{db.port}/@{db.path}	Yes	No	No	No

In the above example, the parameter named “jdbc.url” contains an inline string that includes substitution variables for the parameters “db.host,” “db.port,” and “db.path.” If the value of the referenced parameter “server.hostname” is “db_srv-017,” then the resolved value for parameter “jdbc.url” would be as follows:

jdbc:mysql://db_srv-017:3306/CDA

Parameter Referencing

In HP CDA, parameters can refer to other parameters for their values through a feature known as *parameter referencing*. HP CDA supports multiple levels of parameter referencing. For example, a parameter named “parameter.one” might obtain its value from a parameter named “parameter.three” either by directly referencing “parameter.three,” or by referencing a parameter named “parameter.two” which in turn references “parameter.three.” When parameter referencing is used, HP CDA protects against circular referencing and will not allow it.

In HP CDA, a parameter references another parameter by specifying a reference variable as its value. In the HP CDA interface, reference variables are identified by the following pattern or format:

`${reference-details}`

“reference-details” is a syntax that is readable by HP CDA and is used by it to locate the parameter that has been referenced.

The reference variable pattern/reference-details syntax combination is complex and difficult to enter manually, but the user does not have to do so. When selecting a parameter to reference, the user instead selects the appropriate reference variable from a “Reference Variables” dialog box. The advantages of the “Reference Variables” dialog box are as follows:

- The user does not need to know the reference-details syntax.
- The parameters that can be referenced are categorized and easy to identify via the reference variable names.
- The dialog box will only present the types of reference variables that are appropriate for the parameter doing the referencing. For example, a parameter of a certain type, such as an “Input Component Parameter” in an application layer workflow action, can only reference Server, Endpoint, Application

Model, and Application Layer parameters, and only those reference variable types will appear in the “Reference Variables” dialog box.

After a reference variable has been selected, the value of the referencing parameter is filled in with the reference variable pattern/reference-details syntax combination in blue font in the interface.

<input type="checkbox"/>	Name	L...	Default Value
<input type="checkbox"/>	mysql.dbname	m..	Ref->\${CNXN:connection["7fd2e6a6-45ad-4a03-be50-f40e2d9d8180"].endpoint["d2a302be-6935-4b...
<input type="checkbox"/>	mysql.ipaddress	m..	Ref->\${CNXN:connection["7fd2e6a6-45ad-4a03-be50-f40e2d9d8180"].endpoint["d2a302be-6935-4b...
<input type="checkbox"/>	mysql.port	m..	Ref->\${CNXN:connection["7fd2e6a6-45ad-4a03-be50-f40e2d9d8180"].endpoint["d2a302be-6935-4b...
<input type="checkbox"/>	mysql.remoteuser.name	m..	Ref->\${MODL:appmodel["80467f24-3d42-464c-ad6d-aa47ec41b610"].property.in["mysql.remoteuser...

Most of the information included in the reference-details is not important for you to know, because it is used by HP CDA. However, the four-letter, upper-case string of characters that follows `Ref->${` for many reference types is a *reference type indicator* that designates the type of reference variable that the parameter is referencing.

The following bulleted list explains the types of reference variables that include a four-character reference type indicator:

- **CNXN** (connection endpoint properties reference variable)

The endpoint associated with a Connection. The endpoint properties of one layer are exposed to the connecting layer via the Connection.

- **ENDP** (endpoint properties reference variable)

If a layer endpoint is created, the following properties can be referenced by layer and component parameters.

path: The URL or other endpoint-specific identifier. As an example, if the endpoint is for a database layer, the path might be set to the database name.

port: The port to use to connect to the endpoint.

protocol: The type of connection, for example, HTTP.

ipaddress: *Not currently supported.*

hostname: If load balanced, this will be the DNS name used in the incoming URL. Specifically in the load balancer configuration, this is the value supplied for the `hpio.application.dns.name` parameter. See [Deployment Topology and Load Balancing](#) on page 68 for details and instructions.

- **LAYR** (application layer parameter reference variable)

These are reference variables for parameters that can be shared across actions in a layer. When used, parameters can be defined at the layer level that can be referenced by parameters at the application layer workflow action level using this reference variable type. If the default values of the referencing parameters subsequently need to change, then that value only needs to be changed at the layer parameter level and all the referencing workflow action parameters will automatically get the new default value.

Note: HP CDA contains two special application layer reference parameters named “`servers.hostname.list`” and “`servers.ipaddress.list`.” These reference parameters represent a comma separated list of hostnames or IP addresses for a given application model layer. They are supplied to

provide HP Application Lifecycle Management (HP ALM) a way to obtain a list of all servers that a layer is applied to. These reference parameters can also be used in “Executed Script” or “Placed File” workflow actions should the script or file need this information.

If the list of hostnames or IP addresses is only needed for HP ALM, the model designer need only create a non-private, layer-level parameter that references either the `servers.hostname.list` or `servers.ipaddress.list` reference parameter. The HP ALM process will then be able to bind to the layer parameter and obtain the list of hosts or IP addresses for that layer.

- **MODL** (application model parameter reference variable)

These are similar to application layer (LAYR) parameter reference variables, and are used to create application models with shared parameters.

- **SOFT** (software level parameter reference variable)

These are similar to application layer (LAYR) parameter reference variables, and allow parameter values to be shared across platform software workflow actions.

- **STEP** (component output Parameter reference variable)

These are not supported in version 1.00 of HP CDA. These reference variables, when implemented, will allow output parameters from platform software or application workflow actions to be referenced by the input parameters of subsequent workflow actions.

- **SWRQ** (software tag parameter reference variable)

Platform software might provide capabilities or have software requirements. This is indicated by capability or requirement *software tags*. If platform software is tagged in such a manner by the platform software model designer, then any of the parameters that have been defined for the capability or requirement tag can be referenced by workflow and software level parameters via these reference variables, and they will be substituted at runtime. For example, if the following tags have been created:

- Tomcat 6, having a defined parameter “`catalina_home`”
- Java 6, having the defined parameter “`java_home`”

And if a platform software model is created that is tagged by the designer as providing Tomcat 6 as a capability and Java 6 as a requirement, then platform software-level parameters and workflow parameters may reference the parameters “`Tomcat 6.catalina_home`” or “`Java 6.java_home`.”

Note: Software tags and associated parameters are created and managed in the Tag Management section under the Administration tab. See Chapter 6 [HP CDA System Administration](#) on page 127 for more information.

The following bulleted list explains the types of reference variables that do not include a four-character reference type indicator:

- **“Per Server”** reference variable

These reference variables allow certain parameters to reference the values of HP CDA-created target server parameters. These parameters, such as a target server’s host name (`server.hostname`) and IP address (`server.ipaddress`), are then resolved by the system at runtime.



- **Monitoring** parameter reference variable

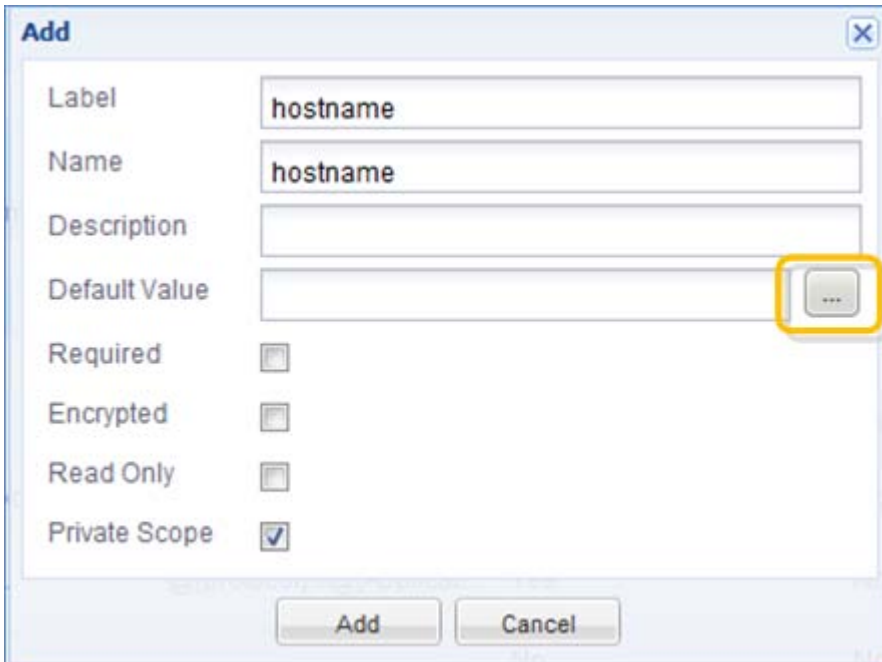
These reference variables allow certain parameters to reference special monitoring-specific parameters.

Referencing Parameters via Reference Variables

Perform the steps in [Adding Parameters](#) until Step 3. When you get to Step 3, do the following for the “Default Value” field:

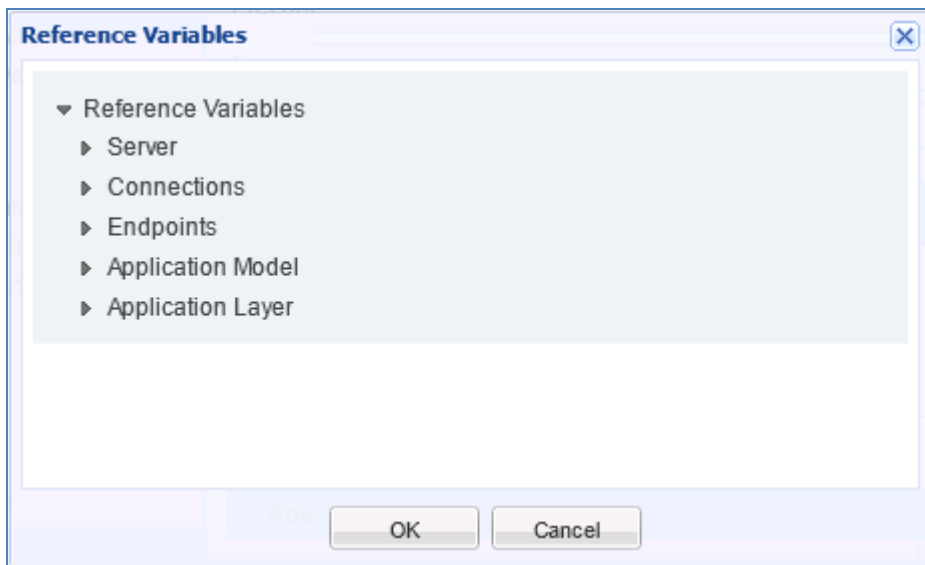
1. Click the ellipsis icon (...) that appears to the right of the “Default Value” field.

Note: If no ellipsis icon appears to the right of the “Default Value” field, then referencing is not supported for the parameter.

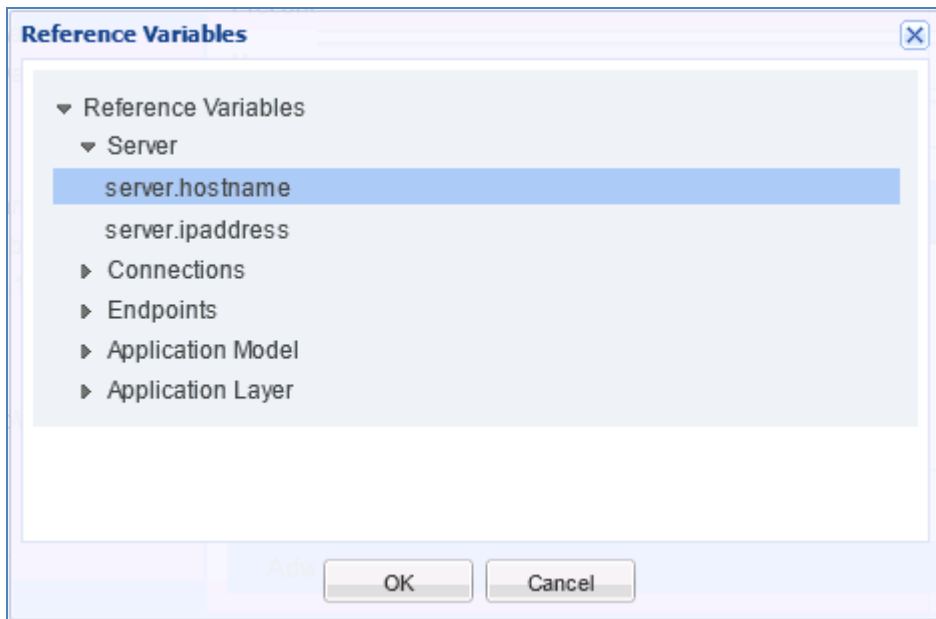


The “Reference Variables” dialog box opens.

The Reference Variables dialog box displays the reference variable types that are valid for the referencing parameter. In the following example graphic, “hostname” is an application layer-level parameter that can reference the variable types that are listed.



2. Click the expand icon (▶) to the left of the reference variable type that you want to reference to expand that category, and in this manner, navigate to and choose the reference variable that the parameter will reference.



- Click **OK** to add the parameter reference and close the “Reference Variables” dialog box.

The reference variable pattern/reference-details syntax combination now appears in the “Add” dialog box.

The screenshot shows a dialog box titled "Add" with the following fields and options:

- Label:** hostname
- Name:** hostname
- Description:** (empty)
- Default Value:** Ref->\${server.hostname} (with a selection icon to the right)
- Required:**
- Encrypted:**
- Read Only:**
- Private Scope:**

At the bottom of the dialog are "Add" and "Cancel" buttons.

- Click the **Add** button in the “Add” parameter dialog box to save the parameter and close the dialog box.

Provisioning Platforms

Platform models are designed to provide certain hardware, OS, and software capabilities as a service once provisioned. Chapter 4, [Platform Development](#) on page 74, provides the information necessary to create platform models. Once a platform model has been provisioned to create a *realized platform*, any application that has requirements that are satisfied by the capabilities of a given provisioned platform can be deployed onto the platform.

The information in this section explains how to provision and deprovision platforms. You need to provision a given platform from a platform model before you can deploy an application onto it.

Provisioning a Platform

The following procedure explains how to provision a platform:

- On the **Platforms** tab in the HP CDA interface, click either the [My Platforms](#) or [All Platforms](#) link, as is appropriate, to display the list of platform models that have been created.

Clicking the [My Platforms](#) link will display a list of platform models that are owned by the user that is currently logged in. The [All Platforms](#) link will display a list of all platform models that have been created.

Note: The tabs that appear above the list of platform models can also be used to switch between the My Platforms and All Platforms view:



2. Click on the name of the platform model that you want to provision from the list of platform models.
The “Platform” screen for the platform model will be invoked.
3. On the **Overview** tab of the “Platform” screen, click the **Provision** icon that appears on the right side of the screen to invoke the “Provision Platform” dialog box.
4. Provide the following information in the “Provision Settings” section of the dialog box:
 - **Provisioned Platform Name:** Enter the screen name to be given to the provisioned platform into this field.
 - **Stage:** Select the lifecycle stage to be assigned to provisioned platform from this drop-down field.
 - **Force Automatic Backout and cleanup on any failure:** Selecting this checkbox will cause a backout operation to begin automatically if the provisioning operation fails (see [Forced Backout](#) on page 31).
5. Either enter or modify the parameter values listed in the “Parameters” section of the dialog box manually, or load a stored parameter set to fill in the parameter values:
 - To load a stored parameter set, click the **Load Parameters** button and select the name of the stored parameter set to load from the drop-down list that appears.
 - If you are entering or modifying the parameter values manually, do so now. For information about parameters and their values, see [Parameters](#) on page 12.

Note: If a parameter set is loaded in order to set the parameter values, the parameter values can still be modified afterwards if necessary.

The parameters that are presented in the “Parameters” section of the dialog box will vary depending on the parameters that were designed into the platform model and the other objects that are related to the platform model. Certain parameters that the designer chose to be “user-exposed” (not private-scoped) will be displayed in the “Parameters” section of the dialog box. The values of the parameters that the designer chose to be read-only will not be editable in the “Parameters” section. The values of the other parameters will be editable.

In addition to the “user-exposed” parameters, parameters related to the chosen Infrastructure Orchestration (IO) server will be exposed. Those parameters are listed below, with descriptions. Required parameters are followed by a red asterisk (*) in the dialog box:

- **Hostname Replacement Text:** Enter the text that IO will use to complete the DNS names of the servers in the service. The text entered here specifies the base part of the server hostname.
 - **Notification Email:** Enter the email address to which IO will send status updates regarding the service.
 - **Server Pool List (Case Sensitive):** Enter the name of the serverpool from which you want IO to provision resources. If no entry is made into this field or the entry is incorrect, IO will choose the serverpool from which it will provision resources.
 - **Service Name:** Enter the name to be given to the provisioned platform service.
6. Click **OK** in the “Provision Platform” dialog box to begin the platform provisioning operation.
A progress bar displays.

Provisioning is being performed. [See report for more details.](#)

4/9

If desired, open a “Report” screen that lists the steps in the provisioning process that are taking place or have taken place by clicking the [See report for more details](#) link.

Note: From the “Report” screen, you can cancel the provisioning operation if desired, and you can also initiate a manual backout operation if the provisioning operation fails and forced automatic backout has not been enabled for the operation. For more information, see [Cancelling Provisioning or Deployment Operations](#) on page 30 and [Manual Backout](#) on page 33.



The provisioning operation will exit with either a completed/finished status or a failed status. Either way, if you have not done so, you can click the [See report for more details](#) link to get more information about the process.

Viewing Provisioned Platforms

After an application has been deployed, it is listed in the “My Deployments” or “All Deployments” screen of the HP CDA interface.

<input type="checkbox"/>	Name ▲	Platform Name	Stage	Deployed Applications	Status
<input type="checkbox"/>	Pet Clinic Provisioned Platform	Pet Clinic Logical Platform	Development	1	

The following information is displayed for each entry in the list:

- **Name:** The screen name of the provisioned platform. This name is given to the provisioned platform when the platform is provisioned by HP CDA (see [Provisioning a Platform](#) on page 23).
- **Platform Name:** The name of the platform model from which the provisioned platform was created.
- **Stage:** The stage of the application lifecycle workflow that the provisioned platform is currently in.
- **Deployed Applications:** The number of applications that are currently deployed onto the provisioned platform.
- **Status:** Whether the platform was provisioned successfully () or provisioning failed (.

To display a list of provisioned platforms, click the Platforms tab in the HP CDA interface, and click either the [My Provisioned Platforms](#) or [All Provisioned Platforms](#) link, as is appropriate, to display the list of provisioned platforms.

Click the [My Provisioned Platforms](#) link to display a list of provisioned platforms that are owned by the user who is currently logged in. Click the [All Provisioned Platforms](#) link to display a list of all provisioned platforms that have been created.

Deprovisioning a Platform

The following procedure explains how to deprovision a provisioned platform:

1. On the Platforms tab in the HP CDA interface, click either the [My Provisioned Platforms](#) or [All Provisioned Platforms](#) link, as is appropriate, to display the list of platforms that have been provisioned.
Clicking the [My Provisioned Platforms](#) link will display a list of provisioned platforms that are owned by the user that is currently logged in. The [All Provisioned Platforms](#) link will display a list of all provisioned platforms.
2. Check the number in the “Deployed Applications” column for the provisioned platform that you want to deprovision.

You cannot deprovision a platform if any realized topologies are associated with the platform. If the number in the “Deployed Applications” column is 0 (zero), then there are no realized topologies associated with the platform.

3. If no realized topologies are associated with the platform, click the name of the platform that you want to deprovision from the list.

The “Platform Deployment” screen for the provisioned platform will be invoked.

4. On the Overview tab of the “Platform Deployment” screen, click the **Deprovision** icon that appears on the right side of the screen.
5. Click **OK** in the “Deprovision” message box that appears to start the deprovisioning operation.

When the deprovisioning operation starts, a “Report” screen opens that lists the steps in the deprovisioning process that are taking place and have taken place.

Note: Once a deprovisioning operation starts, you cannot cancel it. From the “Report” screen, you can initiate a manual backout operation should the deprovisioning operation fail and forced automatic backout had not been enabled for the operation. For more information, [Cancelling Provisioning or Deployment Operations](#) on page 30 and [Manual Backout](#) on page 33.

Deploying Applications

After a platform has been modeled and provisioned, an application has been modeled, and the application’s layers have been mapped to the platform’s tiers, HP CDA can deploy the application. The following sequence occurs during the application deployment process:

- The application is packaged during which a *build* containing all artifacts configured on the **Software Artifacts** tab for the application are accessed and made available to the deployment process.
- The “Deploy” application workflows for each layer of the application model are executed sequentially.
- Monitors are deployed if monitoring policies had been applied to respective application layers in the deployment topology.

The information in this section explains how to deploy and undeploy applications.

Deploying an Application

The following procedure explains how to deploy an application:

1. On the Applications tab in the HP CDA interface, click either the [My Applications](#) or [All Applications](#) link, as appropriate, to display the list of applications that have been created.

Clicking the [My Applications](#) link will display a list of applications that are owned by the user who is currently logged in. The [All Applications](#) link will display a list of all applications that have been created.

The application screen opens by default with the **Simple View** tab selected. Applications are listed alphabetically by application name and version number in this view. The view also provides the deployment status for the Development, QA, Staging, and Production phases in an application’s lifecycle. These status values indicate whether an application is Ready, Ready and Deployed, In Progress, or Not Ready Yet.

2. In the simple view, click on the application version that you want to deploy.

The “Application Version” screen for the application will be invoked.

3. On the **Overview** tab of the “Application Version” screen, click the **Deploy** icon that appears on the right side of the screen to invoke the “Deploy Application” wizard.

Note: The **Deploy** icon also appears on the **Builds** tab of the “Application Version” screen, and in the “Application Model” screen.

4. Provide the following information in the “Deploy Settings” section of the wizard:
 - **Deployment Name:** Enter the screen name to be given to the application deployment in this field.
 - **Package to be deployed:** Select the appropriate radio button, and perform the action as follows:
 - **Deploy New Package:** Select a new package definition that has been defined on the **Software Artifacts** tab in this drop-down field. When you choose this option, a build will be created from the selected package definition when the application is deployed, and the build will subsequently be available on the **Builds** tab.
 - **Deploy Existing Package:** Click the [Select package](#) link, select the build in the “Select Artifact” dialog box that appears, and click **Select**. The builds that are available are listed also on the **Builds** tab.
 - **Application Model:** Select the application model to be deployed in this drop-down field.
 - **Deployment Topology:** Select the deployment topology to be used to map the application model’s layers to the provisioned platform’s tiers in this drop-down field.
 - **Provisioned Platform:** Select the provisioned platform onto which the application is to be deployed in this drop-down field.
 - **Force Automatic Backout and cleanup on any failure:** Selecting this checkbox will cause a backout operation to begin automatically if the deployment operation fails. For more information, see [Forced Backout](#) on page 31.
5. Click **Next** to go to the next screen of the wizard.
6. If any editable parameters appear in the “Application parameters” section of the wizard, enter or modify the parameter values either manually or load a stored parameter set to fill in the parameter values.

Note: If no editable application parameters are available, the message “No parameters to display” will appear under “Application parameters,” in which case click **Next** to go to the next screen of the wizard. Otherwise, set the parameter values as follows:

- To load a stored parameter set, click the **Load Parameters** button and select the name of the stored parameter set to load from the drop-down list that appears.
- If you are entering or modifying parameter values manually, do so now. For information about parameters and their values, see [Parameters](#) on page 12.

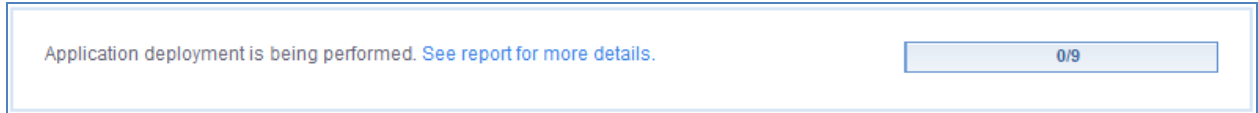
Note: If a parameter set is loaded in order to set the parameter values, the parameter values can still be modified afterwards if necessary.

The parameters that are presented in the “Application parameters” section of the wizard will vary depending on the parameters that were designed into the application model and the other objects that are related to the model. Certain parameters that the designer chose to be “user-exposed” (not private-scoped) will be displayed in the “Application parameters” section of the wizard. The values of the parameters that the designer chose to be read-only will not be editable in the “Application

parameters” section. The values of the other parameters will be editable. All required parameters are followed by a red asterisk (*) in the “Application parameters” section.

7. Click **Next** to go to the next screen of the wizard.
8. Modify the parameters listed under “Monitoring parameters” if necessary. Step 6 provides information on how to do this.
9. Click **Finish** to begin the application deployment operation.

A progress bar displays.



If desired, open a “Report” screen that lists the steps in the deployment process that are taking place or have taken place by clicking the [See report for more details](#) link.

Note: From the “Report” screen, you can cancel the deployment operation if desired, and you can also initiate a manual backout operation should the deployment operation fail and forced automatic backout had not been enabled for the operation. For more information, see [Cancelling Provisioning or Deployment Operations](#) on page 30 and [Manual Backout](#) on page 33.

The deployment operation will exit with either a completed/finished status or a failed status. Either way, if you have not done so, you can click the [See report for more details](#) link to get more information about the process.

Viewing Deployed Applications

After an application is deployed, it displays in the “My Deployments” or “All Deployments” screen of the HP CDA interface.

Name	Application	Application Version	Build	Stage	Status
Pet Clinic Deployed Application	Pet Clinic Application	Pet Clinic Application Version 0.11.0.3	Build_25	Development	✓

The following is displayed for each entry in the list:

- **Name:** The screen name of the application deployment. This name is given to the application deployment when the application is deployed by HP CDA (see [Deploying an Application](#) on page 26).
- **Application:** The name of the application of which the version listed under “Application Version” was deployed.
- **Application Version:** The version of the application that was deployed.
- **Build:** The build that was created by the application deployment or that was specified as an existing package when the application was deployed.
- **Stage:** The stage of the application lifecycle workflow that the deployed application is currently in.
- **Status:** Whether the application deployment was successful (✓) or failed (✗).

To display a list of deployed applications, click the **Applications** tab in the HP CDA interface, and click either the [My Deployments](#) or [All Deployments](#) link, as is appropriate, to display the list of application deployments.

Click the [My Deployments](#) link to display a list of application deployments that are owned by the user that is currently logged in. Click the [All Deployments](#) link to display a list of all application deployments that have been created.

Undeploying an Application

The following procedure explains how to undeploy an application:

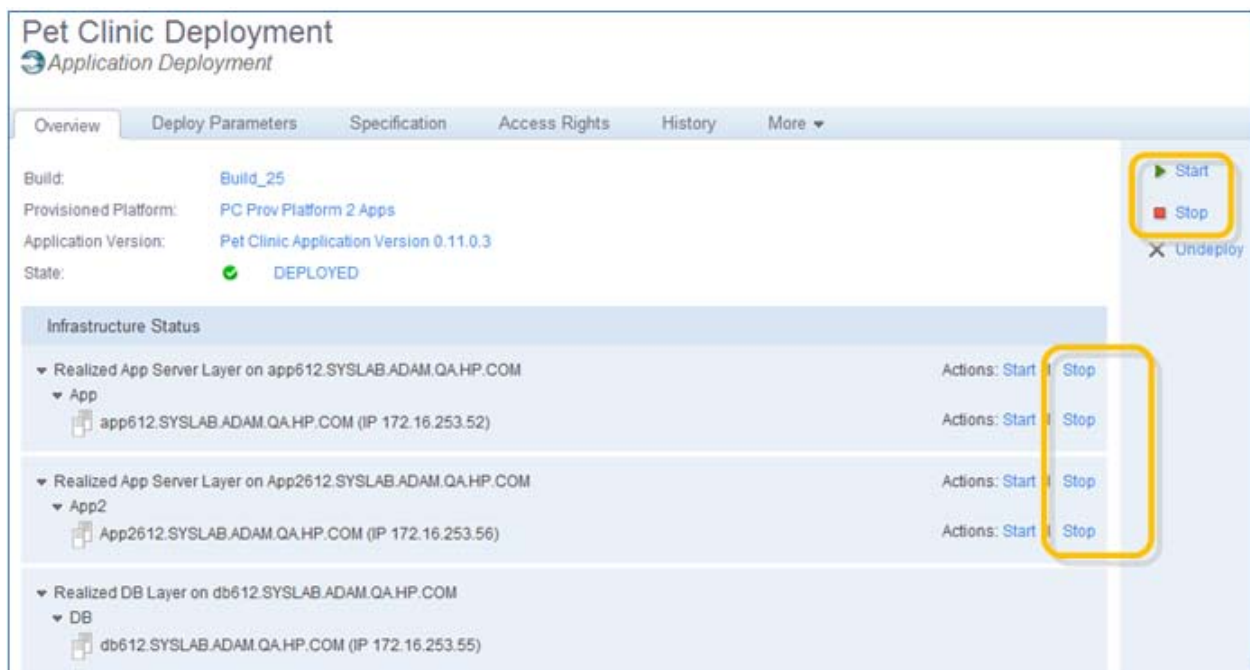
1. On the Applications tab in the HP CDA interface, click either the [My Deployments](#) or [All Deployments](#) link, as is appropriate, to display the list of applications that have been deployed.
2. Clicking the [My Deployments](#) link will display a list of deployed applications that are owned by the user that is currently logged in. The [All Deployments](#) link will display a list of all deployed applications that have been created.
3. Click the name of the deployed application from the list of deployed applications that you want to undeploy.
4. The “Application Deployment” screen for the deployed application will be invoked.
5. On the **Overview** tab of the “Application Deployment” screen, click the **Undeploy** icon that appears on the right side of the screen.
6. Click **OK** in the “Undeploy Application” message box that appears to start the undeploy operation.

When the undeploy operation starts, a “Report” screen opens that lists the steps in the undeployment process that are taking place and have taken place.

Note: Once an undeployment operation starts, you cannot cancel it. From the “Report” screen, you can initiate a manual backout operation should the undeployment operation fail and forced automatic backout had not been enabled for the operation. For more information, see [Cancelling Provisioning or Deployment Operations](#) on page 30 and [Manual Backout](#) on page 33.

Start and Stop Operations

When an application is modeled, the designer has the option of including start and stop workflows in each layer of the application model. If those workflow types have been modeled into the application model, the “Application Deployment” screen for the application that was deployed from the model will contain start and stop icons and links.



In the example graphic, start and stop workflows had been designed into the two application server (App Server) layers of the application model but not into the database (DB) layer, therefore [Start](#) and [Stop](#) links

appear adjacent to both application server layers but not next to the database server layer in the “Application Deployment” screen.

Click the [Start](#) or [Stop](#) link that is adjacent to a given application layer in the “Application Deployment” screen to execute the start or stop workflow only on the respective server. This is useful, for example, if an application is running on redundant application servers and one of them needs to be taken down for servicing.

Click the **Start** or **Stop** icons that appear on the right side of the window to run the respective start or stop workflows on all deployed servers that are part of the application model.

Cancel, Backout, and Forced Cleanup Operations

Cancelling Provisioning or Deployment Operations

During a platform provisioning or application deployment operation, the need might arise for the operation to be cancelled. HP CDA includes a *cancel* feature for such a purpose.

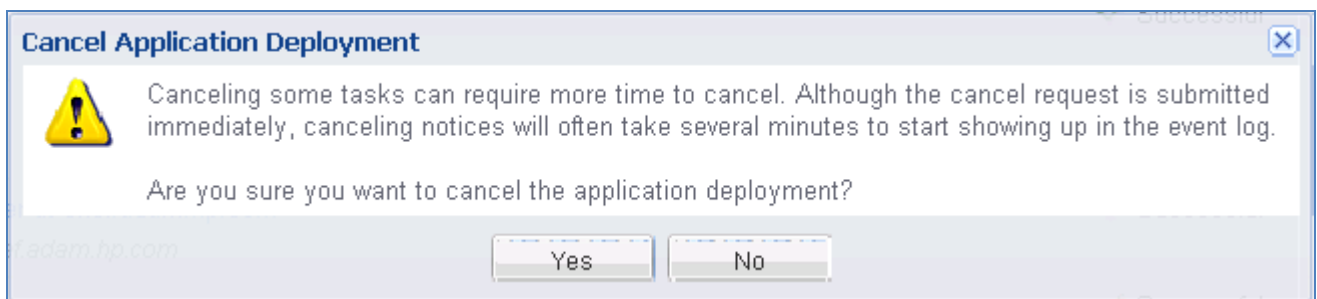
Internally, a cancel sequence, when executed, allows the current step or steps in the provisioning or deployment operation to complete so that the system is in a safe state, and then a backout sequence automatically commences.

Note: In the case of a cancel operation, no failure in the provisioning or deployment operation has likely occurred, but the backout sequence runs anyway to reverse the steps of the operation that had already taken place. For general information on the backout sequence, see [Backing Out of Failed Operations](#) on page 31.

When a platform provisioning or application deployment operation commences, a “Report” screen that lists the steps that are taking place becomes available, and a **Cancel** button appears near the top of the report.



Upon clicking the **Cancel** button, a message similar to the following appears:



Upon clicking the **Yes** button, the report lists the cancel and backout steps that are taking place.

Backing Out of Failed Operations

For a variety of reasons, platform provisioning or application deployment operations can fail to execute successfully. HP CDA includes a *Backout* feature that enables the system to reverse and undo those operations of the failed provisioning or application deployment sequence that have already taken place. The system can be configured to initiate the backout sequence automatically upon a failure (referred to as a *forced backout*), or allow the user to initiate the backout sequence manually (referred to as a *manual backout*).

The following two types of backout steps are used during a backout sequence:

- System-Generated Backout Steps
- User-Created Backout Steps

System-Generated Backout Steps

System-generated backout steps execute automatically during a backout sequence, and are determined by the system. System-generated backout steps undo any system-controlled tasks that have already taken place and delete any system-generated model elements that had been created. For example, if an infrastructure provisioning operation fails after some virtual machines have already been stood up from an IO template, HP CDA will automatically trigger IO to have the hypervisor remove the virtual machines, and then HP CDA will remove data related to those VMs from the HP CDA database.

User-Created Backout Steps

User-created backout steps are not determined by the system, but are designed and created by the application or platform modeler as part of the application or platform modeling process. They specifically pertain to “Deploy” workflows. Backout steps cannot be defined for programming actions associated with Undeploy, Start, Stop, and custom workflows.

Each programming action in a “Deploy” workflow can include an associated backout step. In the HP CDA user interface, user-created backout steps are created on the **Backout** tab for each Deploy workflow programming action that is created.



If no backout step is created on the **Backout** tab for a given programming action in a workflow, then no backout action will take place during backout for that particular programming action.

Forced Backout

When forced backout is enabled for a given platform provisioning or application deployment operation, backout begins automatically when the operation fails. Enabling forced backout for a provisioning or deployment operation consists of selecting a checkbox in the screen in which the operation is initiated. The **Force Automatic Backout and cleanup on any failure** checkbox is available in the following HP CDA UI screens:

- For Platform Provisioning – The “Provision settings” section of the “Provision Platform” dialog box.

Provision Platform

Provision settings

Provisioned Platform Name: *

Stage:

Force Automatic Backout and cleanup on any failure:

Parameters

■ ie_io_server

Hostname Replacement Text: *

Notification Email:

Server Pool List (Case Sensitive):

Service Name: *

- For Application Deployment – The “Deploy settings” screen of the “Deploy Application” wizard.

Deploy Application

Deploy settings

Deployment Name: *

Package to be deployed: Deploy New Package
 Package Definition:

Deploy Existing Package
 Package: [Select package](#)

Application Model: *

Deployment Topology: *

Provisioned Platform: *

Force Automatic Backout and cleanup on any failure:

After the provisioning or deployment operation commences, a “Report” screen that lists the steps that are taking place becomes available. Should a step in the provisioning or deployment operation fail, messages, such as the following, will appear in the report indicating that forced backout has automatically started.

Deployment failed and force backout was selected. Forcing backout. <i>Deployment failed and force backout was selected. Forcing backout.</i>	✓ Successful
Initializing backout process <i>Initializing backout process</i>	✓ Successful

Additional messages will then appear in the report to explain the backout steps that are taking place.

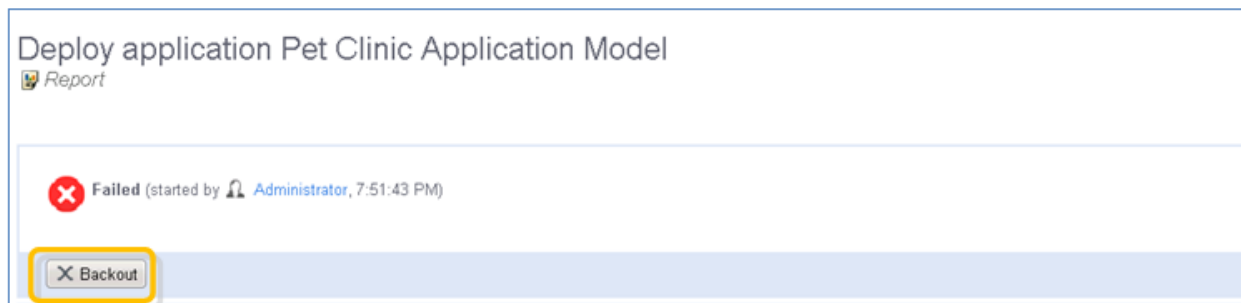
Note: Additional parameter settings that can be made for the individual programming actions of a workflow have an effect on forced backout. See the “Advanced Options” setting descriptions for the programming actions in [Workflows and Programming Actions](#) on page 80.

If a forced backout operation fails, HP CDA includes an additional feature called *forced cleanup* that can be used. Refer to [“Forced Cleanup of Failed Operations”](#) below for more information.

Manual Backout

If forced backout is not enabled and a platform provisioning or application deployment operation fails, HP CDA provides an option for the user to initiate a backout sequence manually.

When a provisioning or deployment operation commences, a “Report” screen that lists the steps that are taking place becomes available. Should a step in the provisioning or deployment operation fail, the provisioning or deployment operation suspends, and a **Backout** button appears near the top of the report.



Upon clicking the **Backout** button, a separate “Report” screen becomes available specifically for the backout sequence. The report lists the backout steps that are taking place.

Note: Additional parameter settings that can be made for the individual programming actions of a workflow have an effect on manual backout. See the “Advanced Options” setting descriptions in [Table 4 Placed File Screen Parameter Settings](#).

If a manual backout operation fails, HP CDA includes an additional feature called *forced cleanup* that can be used (see [Forced Cleanup of Failed Operations](#) on page 33).

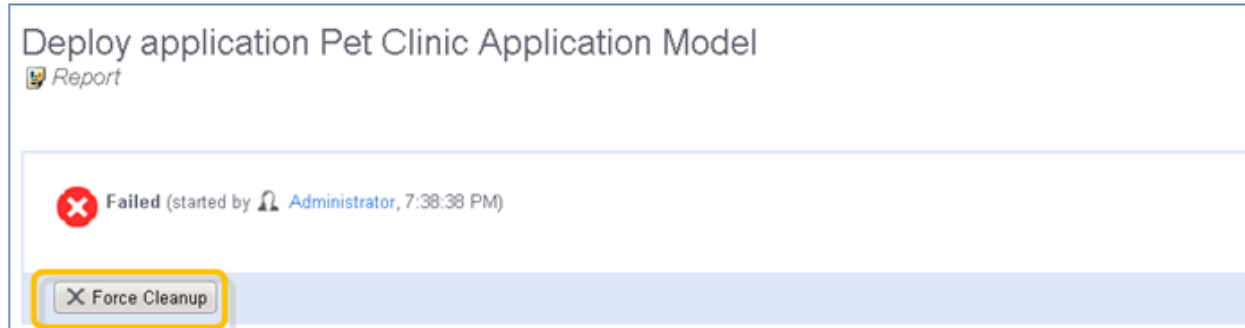
Forced Cleanup of Failed Operations

If a forced backout or cancel operation fails to complete successfully, the system might be left in an unknown state, with certain resources, database objects, and model elements left in place which cannot be cleaned up using normal HP CDA administration tasks. For such cases, HP CDA includes a *forced cleanup* feature.

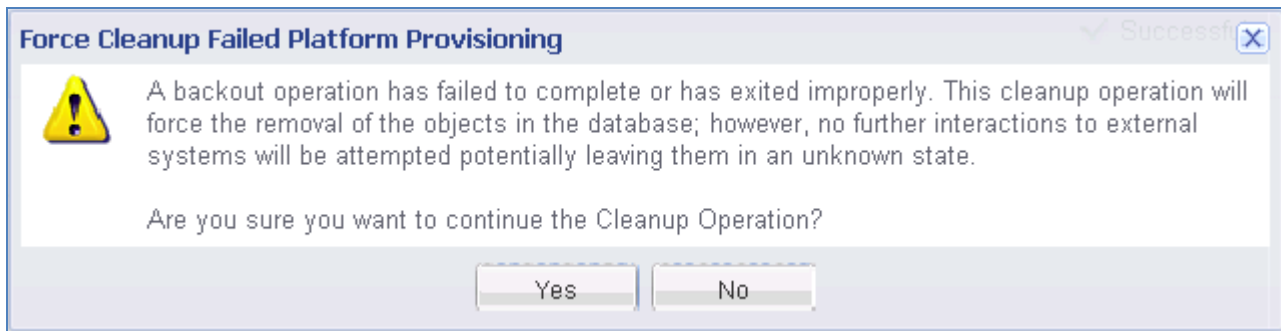
When a forced cleanup sequence executes, HP CDA removes system-created model elements that are related to the failed action from the interface and HP CDA database. It does not attempt to remove any other objects that might have been created by external providers, such as virtual infrastructure, that were not removed during the failed backout or cancel operation.

As an example, HP CDA might have created a Provisioned Platform model element in the database during a platform provisioning operation that subsequently failed, and the ensuing backout sequence also failed. In such a case, the forced cleanup sequence will remove the Provisioned Platform element from the list of provisioned platforms in the HP CDA user interface and from the database, but will not remove related objects, such as virtual machines that were stood up by IO through the hypervisor.

Forced cleanup is available only after a backout or cancel operation fails to complete successfully. When the backout or cancel operation fails, a **Force Cleanup** button appears near the top of the backout report.



Upon clicking the **Force Cleanup** button, a message similar to the following appears.



Upon clicking the **Yes** button, a separate "Report" screen becomes available specifically for the forced cleanup sequence. The report lists the cleanup steps that are taking place.

Archive Management

HP CDA enables you to export selected models and other objects to a ZIP file for later import to the same or a different instance of HP CDA. You can export such objects in bulk or individually. By default, export includes all sub-components that are part of the same aggregations as the objects selected for export.

Exporting Single Models to a ZIP Archive File

To export single models to a Zip archive file, complete the following steps:

1. Perform either of the following two actions to open the "Export" dialog box:
 - In the screen that lists the models of the desired type (for example, My Platforms), select the checkbox that is to the left of the model that you want to export, and then click the **Export** icon that is above the tabular listing.

- In the details screen for the model that you want to export, click the **Export** icon that is on the right side of the screen.
2. (Optional) Change the name of the ZIP file in the “Archive Name” field.

Note: By default, HP CDA exports all the sub-components in the aggregation for the model. To export only the model itself, expand Advanced Options and deselect the Include Dependencies checkbox.

By default, HP CDA does not export the revision history of the model. To export the revision history, expand Advanced Options and select the Include History checkbox.
 3. Click the **Export** button to start the export operation.

A progress bar will appear at the top of the screen as the export operation proceeds, and then the following message appears:

Your download should start automatically. Use [this direct link](#) if it does not or you want to download the archive later.
 4. If you want to download the ZIP file now, continue with the next step. If you want to navigate away from the screen and download the ZIP file at a later time, skip to Step 7.
 5. (Optional) Click [this direct link](#) in the message if a “File Download” dialog box does not appear.
 6. Click the **Save** button in the “File Download” dialog box, and then use the “Save As” dialog box to save the ZIP file to a desired location.
 7. (Optional) If you navigated away from the screen where the export operation was initiated and now want to download the ZIP file, open the report for the export operation, and click the **Download Contents** button that appears near the top of the report.
 8. Click the **Save** button in the “File Download” dialog box that appears, and then use the “Save As” dialog box to save the ZIP file to a desired location.

Exporting Multiple Models to a ZIP Archive File

To export multiple models to a .Zip archive file, complete the following steps:

1. In the screen that lists the models of the desired type (for example, My Platforms), select the checkboxes that are to the left of the models that you want to export, and then click the **Export** icon that is above the tabular listing.
2. Optionally, change the name of the ZIP file in the “Archive Name” field.
3. By default, HP CDA exports all the sub-components in the aggregation for the models. To export only the models themselves, expand **Advanced Options** and deselect the **Include Dependencies** checkbox.
4. By default, HP CDA does not export the revision history of the models. To export the revision history, expand **Advanced Options** and select the **Include History** checkbox.
5. Click the **Export** button to start the export operation.
6. A progress bar will appear at the top of the screen as the export operation proceeds, and then the following message appears:
7. (Optional) Use [this direct link](#) if your download does not start automatically or you want to download the archive later
8. If you want to download the .ZIP file now, continue with the next step. If you want to navigate away from the screen and download the ZIP file at a later time, skip to Step 11.

9. Click [this direct link](#) in the message if a “File Download” dialog box does not appear.
10. Click the **Save** button in the “File Download” dialog box, and then use the “Save As” dialog box to save the ZIP file to a desired location.
11. (Optional) If you navigated away from the screen where the export operation was initiated and now want to download the ZIP file, open the report for the export operation, and click the **Download Contents** button that appears near the top of the report.
12. Click the **Save** button in the “File Download” dialog box that appears, and then use the “Save As” dialog box to save the ZIP file to a desired location.

Note: The administrator can export the content of an entire domain and lifecycle processes..

Importing Model Archives

To import model archives, complete the following steps:

1. Click the [Model Archive](#) link that is in the navigation pane at the left side of the screen.
The [Model Archive](#) link appears in the navigation pane on the **Applications**, **Platforms**, and **Monitoring** tabs. Clicking the link opens the “Repository Archive Import” screen.
2. Select the appropriate “Import from” radio button (**File** or **URL**) in the “Repository Archive Import” screen.
3. Either enter the URL where the file is located, or click the **Browse...** button, browse to and select the file to be uploaded (imported), as is appropriate for the radio button selected above.
4. Select a “Conflicting Data” radio button, as is appropriate:
 - **Preserve Data in Repository** – The properties defined in the models being imported will not overwrite the properties of existing models in the repository that have the same name.
 - **Overwrite Data in Repository** – The properties defined in the models being imported will overwrite the properties of existing models in the repository that have the same name.
5. If desired, expand **Advanced Options** and select the appropriate options in Table 3.

Note: The options in Table 3 only take effect if you have the appropriate permissions.

Table 3, Model Archive Import Advanced Options

Advanced Option	Description
Conflicting Relationships	Selects whether to merge or overwrite relationship conflicts. Merge adds any new relationships defined in the archive for existing models to any existing relationships. Overwrite replaces relationships for existing models with the new relationships defined in the archive.
Import Access Rights	Selects whether to use the access rights defined in the archive. If the checkbox is selected, the new access rights overwrite any access rights for existing models.
Import Lifecycle Status	Selects whether to use the lifecycle statuses defined in the archive. If the checkbox is selected, the import replaces the lifecycle process, lifecycle stage, and status of any existing models.
Destination Domain	Selects whether to import to the domains defined in the archive (Same as Source radio button) or to the current domain (Current radio button). If you select to use domains defined in the archive and the domain does not exist, the models are not imported.

Advanced Option	Description
Artifact Ownership	Selects whether to use ownership defined in the archive (Same as Source radio button) or to assign ownership of imported content to the current user (Current radio button).
System Settings	Select the checkboxes for the system settings that you want to import if the archive is a domain archive.

6. Click the **Import** button to start the archive import process.

The archive import process executes as a bulk operation. A progress bar will appear at the top of the screen to inform you that the operation is in progress, with **Stop** and **Notify Me** buttons available so you can stop the operation and/or have a notification sent to you when the operation completes. The operation executes asynchronously, so you can navigate away and perform other tasks while the operation is running.

Note: HP recommends updating the database schema statistics after importing large amounts of data. Old statistics may impact the performance of some data queries. Consult your database administrator.

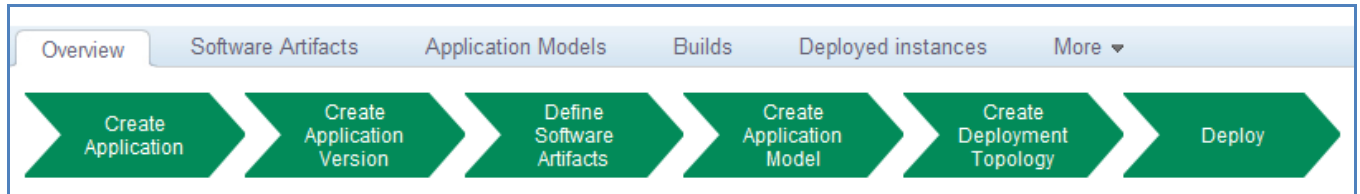
When the operation completes, a report is generated. The report includes an **Undo Changes** button that, when clicked, undoes changes to published models. The undo changes operation also executes as a bulk operation.

The undo changes operation only applies to the revision of a given model that was imported. Any subsequent modifications to a given model will change its revision and the undo changes operation will not rollback those models.

3 Application Development

Application Creation & Management

Figure 1 HP CDA Model



Model Driven Approach

HP Continuous Delivery Automation (HP CDA) provides a model-driven approach to automate application deployment in a development and operations environment. With this approach, you design an application that can be deployed and managed in a repeatable fashion on multiple target platforms.

In HP CDA, you create and manage the application model, deploy the application, and manage the deployment, all with full artifact version control. The application model defines and manages modifications and ensures that whatever is developed and tested is what goes to production.

Application Model

An application model contains layers, and these layers contain application workflows. When a workflow is run, components in the workflow are then translated into a deployer-specific model. The workflow is a container for components that are an abstraction layer on top of the deployer-specific model. In HP CDA, different application models allow an application developer to deploy an application to multiple target platforms.

An application model contains the build artifacts, scripts, and commands that are required to deploy an application. These artifacts include the programming operations that you typically perform on an application, such as Start and Stop. As a best practice, you can also create custom programming operations, such as Pause, Restart, and Backup.

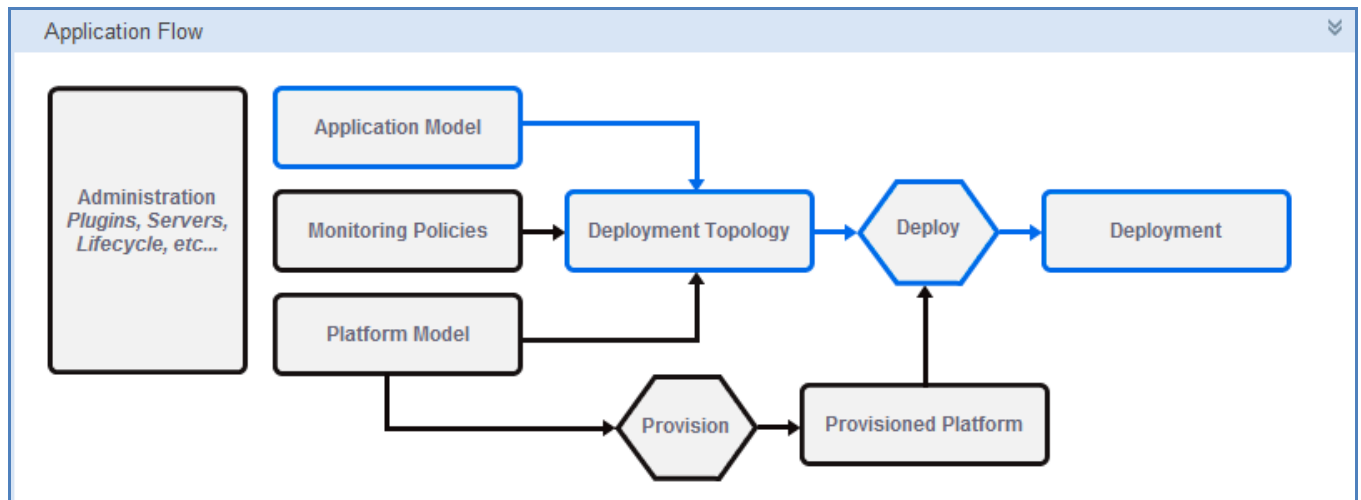
Application Lifecycle Workflow

An application moves through lifecycle stages, such as Development, QA, Staging, or Production. This workflow is commonly known as Application Lifecycle Management (ALM), which is a continuous process of managing an application through governance, development, and maintenance. In HP CDA, you use the same application model in all ALM stages to provide a standard for the application.

In HP CDA, the application workflow is a sequence of steps that can be translated into code for leading HP Software deployment products, such as HP Database and Middleware Automation (HP DMA), HP Operations Orchestration (HP OO), HP Server Automation (HP SA), and other products, such as Opscode Chef. See the *HP Continuous Delivery Automation Installation and Configuration Guide* for more information about deployment providers.

Application, Application Version, & Application Model

Figure 2 HP CDA Application Flow



In HP CDA, you must define how you want the application and build server artifacts deployed on target platforms. The application model is the container for this information. You create and manage application models, deployment topologies, and deployments.

For applications, application versions, and application layers, you define requirements and setup steps, including integration with build systems. You can deploy applications using deployment topologies, manage application deployments, and track the lifecycle of application builds.

Managing the Application Model

Conceptually, the application model is positioned between the continuous integration (build) environment and the target (deployment) environment.

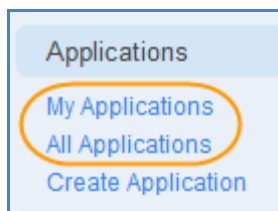
HP CDA provides a component hierarchy for managing application models, using the following terminology:

- **Application** is comprised of application versions.
- **Application Version** is defined by one or more application models.
 - An application version can define more than one application model, if the application can be deployed on different operating systems or middleware containers.
 - An application version defines its package definition (the external build artifacts to access), including the location of the build artifacts. The application version can define multiple package definitions, depending on how the artifacts are managed, either by iterations or by other project milestones.
- **Application Model** defines the layers of the application, including the recipe for managing the deployment of the application.
 - Within the application model, you define the layers of composite applications. For each layer, you can define certain endpoint services that the layer exposes to other layers.
- **Software Artifacts** specify the files you will use in your application model as part of the deployment. Software artifacts can be source files or binary files that your organization develops, or additional files that the application requires.
- **Layers** and **Package Definitions** are modeled, after which the designer models all workflows for each layer to manage the deployment. A workflow step defines the following:

- **Placed File:** Based on the application model's build configuration model, locate the files and then define location and permissions on the target.
 - **Executed Script:** Execute a local script that is appropriate for the target's operating system type on the target.
 - **Service Command:** Execute a local service command on the target.
 - **External Component:** Execute a remote sequence automation from an external service, such as an HP Operations Orchestration (HP OO) workflow or an Opscode Chef cookbook. See the *HP Continuous Delivery Automation Installation and Configuration Guide* for more information about deployment providers.
- **Deployment Topology:** Map the application model's layers to server groups in a certain platform.

Viewing Applications

Access to the following **Applications** views depends on which type of user you are logged in as.



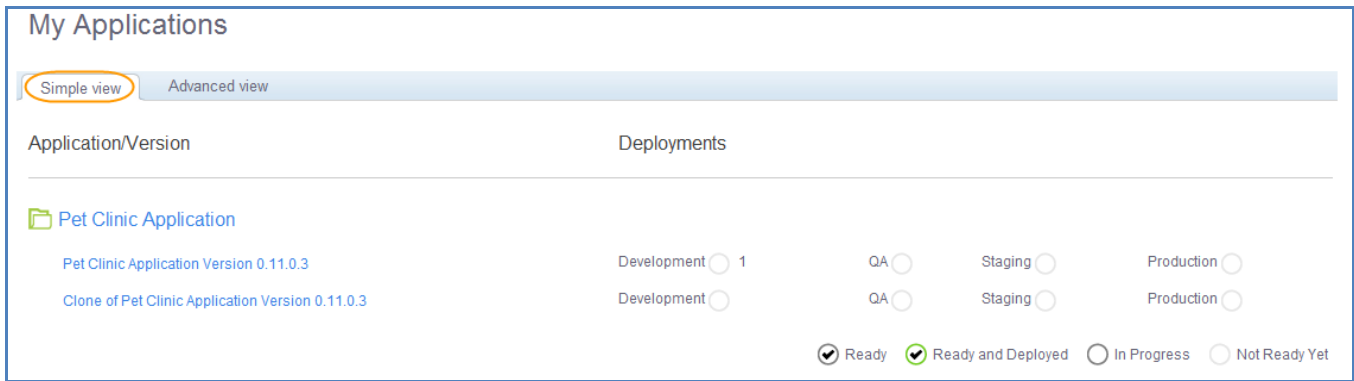
Users who need easy access to application models are application developers, domain administrators, and tenants.

When you install HP CDA, you can choose to install it as a multi-tenancy application. In a multi-tenancy application installation, you can create tenants. Domain administrators manage access to tenants' data. This access is specific to a certain tenant. Tenants cannot access each other's data. For example, tenants can be separate companies, departments within a company, and so on. These companies and departments do not have access to each other's data. See the *HP Continuous Delivery Automation Installation and Configuration Guide* for more information about a multi-tenant installation.

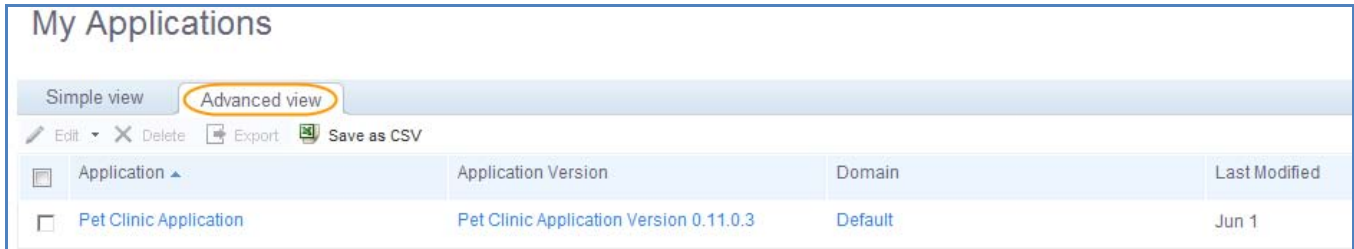
The My Applications window displays all applications and application details that you, an application developer, created.

Following are the views offered on My Applications:

- **Simple View:** Lists all applications that you, as an application developer, created. These applications are listed alphabetically by application name and version number. This view also provides the deployment status for the Development, QA, Staging, and Production stages in an application's lifecycle. These status values indicate whether an application is Ready, Ready and Deployed, In Progress, or Not Ready Yet.



- Advanced View:** Lists all applications that you have access to, identifies the target domain, and shows when the application version was last modified. This view allows you to perform certain actions on application artifacts, such as modifying or deleting artifacts, exporting artifacts, and saving application metadata as a Microsoft® Excel Comma Separated Values File (.csv).



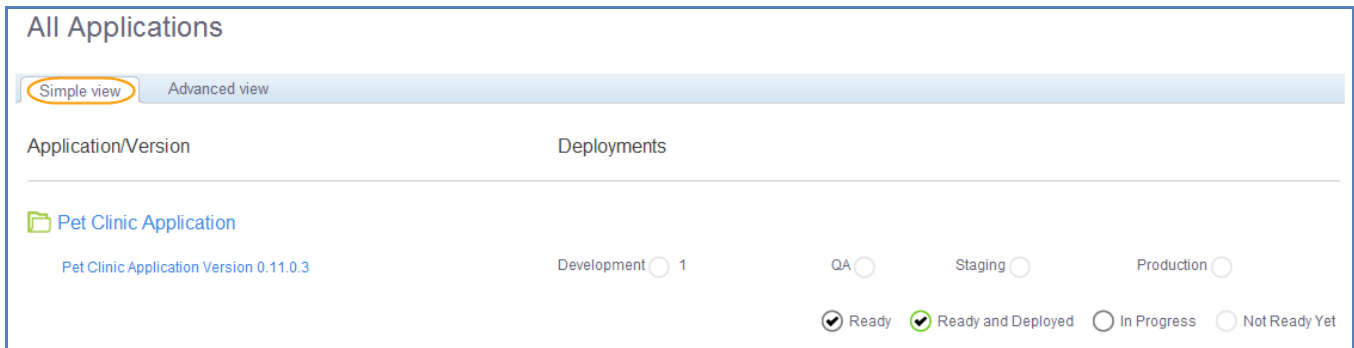
You can perform the following functions in My Applications Advanced View:

- **Edit:** Edits all selected application artifacts. You can edit application properties, change the application owner, change Read/Write permissions, change the target domain, grant Read-only permissions to all users, and revoke Read-only permissions from all users.
- **Delete:** Deletes all selected application artifacts from the Definitive Software Library (DSL).
- **Export:** Exports all selected application artifacts to a model archive, which is a .zip file that you save to your local file system.
- **Save as CSV:** Saves all application Advanced View information as a Microsoft® Excel Comma Separated Values File (.csv).

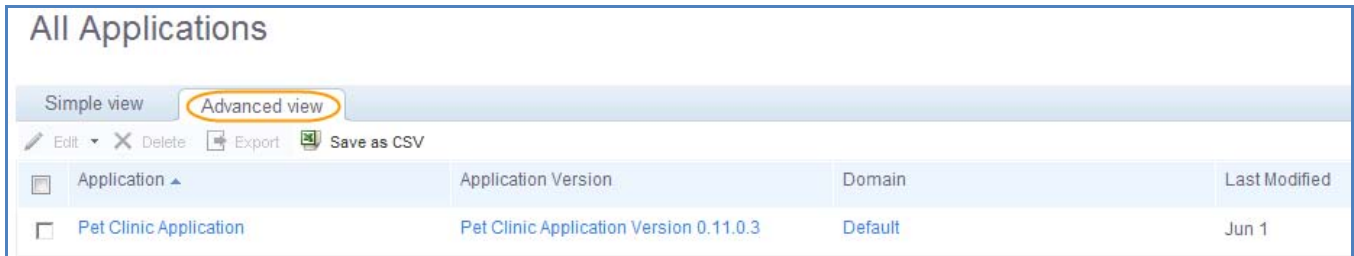
The All Applications window displays all applications and application details that have been created. These are applications that you have access to, which depends on the type of user you are logged in as.

Following are the views offered on My Applications:

- Simple View:** Lists all applications that application developers created. These applications are listed alphabetically by application name and version number. This view also provides the deployment status for the Development, QA, Staging, and Production stages in an application’s lifecycle. These status values indicate whether an application is Ready, Ready and Deployed, In Progress, or Not Ready Yet.



- Advanced View:** Lists all applications that application developers created. This information lists all applications that application developers created, identifies the target domain, and shows when the application version was last modified. This view allows you to perform certain actions on application artifacts, such as modifying or deleting artifacts, exporting artifacts, and saving application metadata as a Microsoft® Excel Comma Separated Values File (.csv).

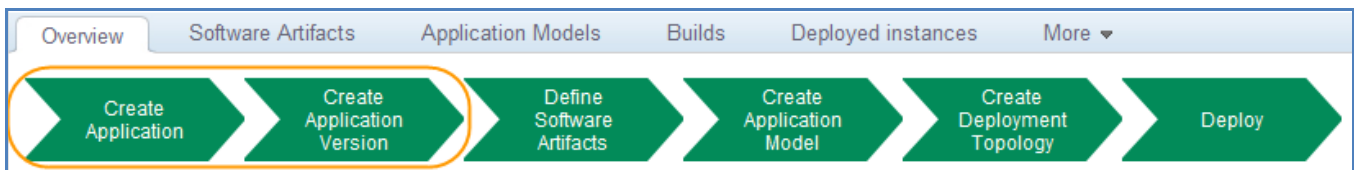


You can perform the following functions in All Applications Advanced View:

- Edit:** Edits all selected application artifacts. You can edit application properties, change the application owner, change Read/Write permissions, change the target domain, grant Read-only permissions to all users, and revoke Read-only permissions from all users.
- Delete:** Deletes all selected application artifacts from the Definitive Software Library (DSL).
- Export:** Exports all selected application artifacts to a model archive, which is a .zip file that you save to your local file system.
- Save as CSV:** Saves all application Advanced View information as a Microsoft® Excel Comma Separated Values File (.csv). This information includes the application name and version, the link to artifact, and lifecycle stage.

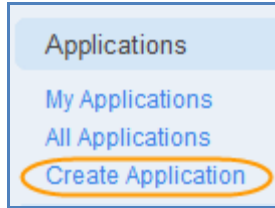
Creating an Application and Application Version

Figure 3 HP CDA Model: Create Application



When new software is introduced, you create an application version that defines a new or modified set of software artifacts. To distinguish between versions, assign an application version a unique, sequential number, such as HP Server Automation 9.10, HP Server Automation 9.11, and so on. As a best practice, the application designer should verify that version numbers are unique.

When you create an application, you must first name it and then define its version properties. Optionally, you can also briefly describe the application and application version. The Create Application and Version wizard guides you through the steps required to create an application.



To create an application, complete the following wizard steps:

1. In the Application Properties step:
 - a. Enter a name for the application that you are working on.
 - b. (Optional) Enter a brief description of the application.
 - c. Click Next to advance to the next wizard step.
2. In the Application Version Properties step:
 - a. Enter a version name and number for the application that you are working on.
 - b. (Optional) Enter a brief description of the application version.
 - c. From the Build Lifecycle Process drop-down list, select **Production Build Process** or **Simple Build Process**.
 - **Production Build Process:** The application will move from Development to QA to Staging to Production, without rules or tasks defined in each stage. This process can be used as a template and customized for an application lifecycle in your environment.
 - **Simple Build Process:** The application will move from Development to QA, without rules or tasks defined in each stage. This process can be used as a template and customized for an application lifecycle in a Lab Management Automation environment.
3. Click **Finish** to save your application properties. You have created the application and the application version that you are working on.

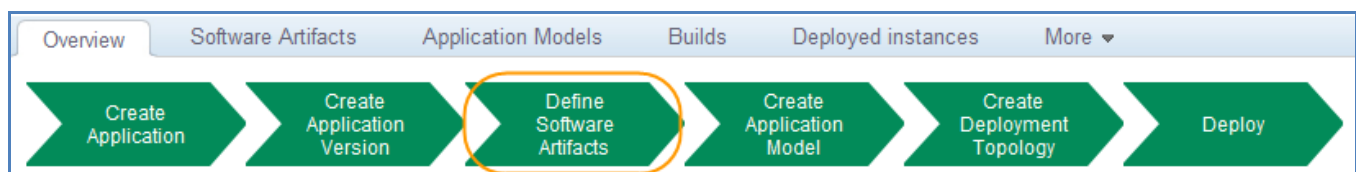
Software Artifacts

The initial stage in the application lifecycle is creating an application. This means developing the source code and structuring the placement of files in the repository.

The files required to deploy an application are known as *software artifacts*. Software artifacts can be source files or binary files that your organization develops, or any additional files that the application requires.

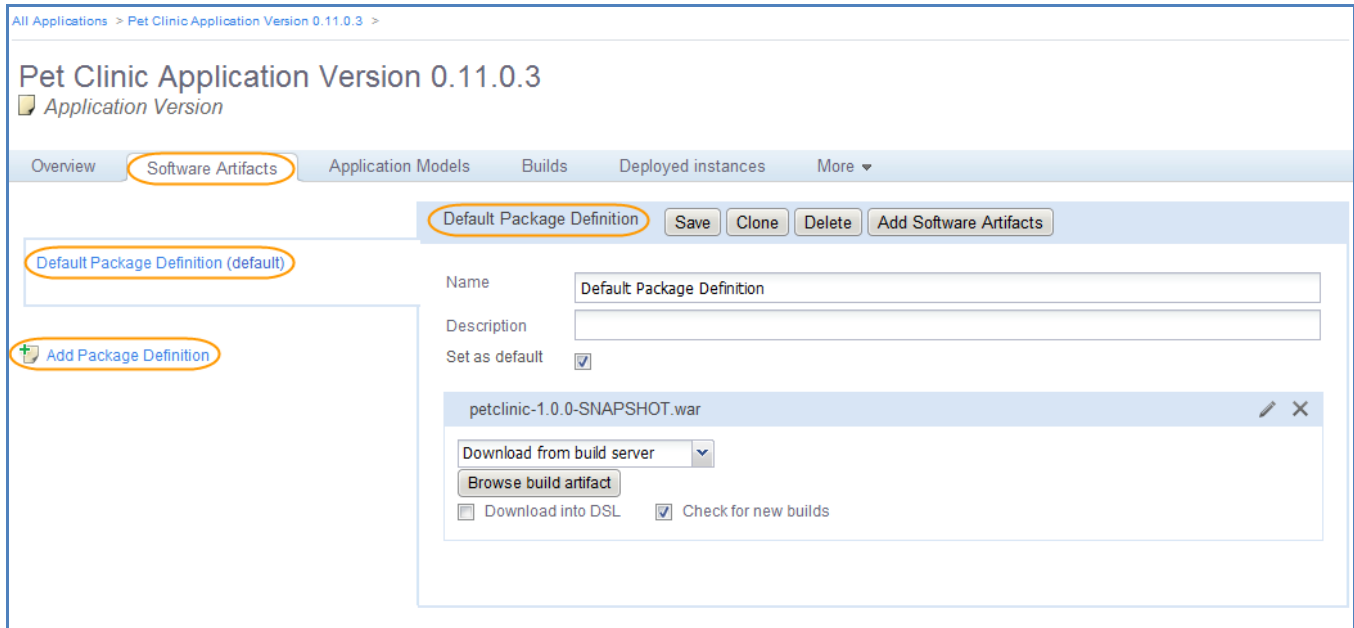
Software artifacts can reside locally, such as on your build server, or they can reside remotely and be linked using a URL. They are managed by a continuous integration (CI) tool. You use the CI tool to manage the build process.

Figure 4 CDA Model: Define Software Artifacts



Defining Software Artifacts

Software artifacts are required to integrate applications with your build server. Your build server is where you map high level software artifacts that are generated (by your build server) for an application. Software artifacts also specify what you will use in your application model as part of the deployment.

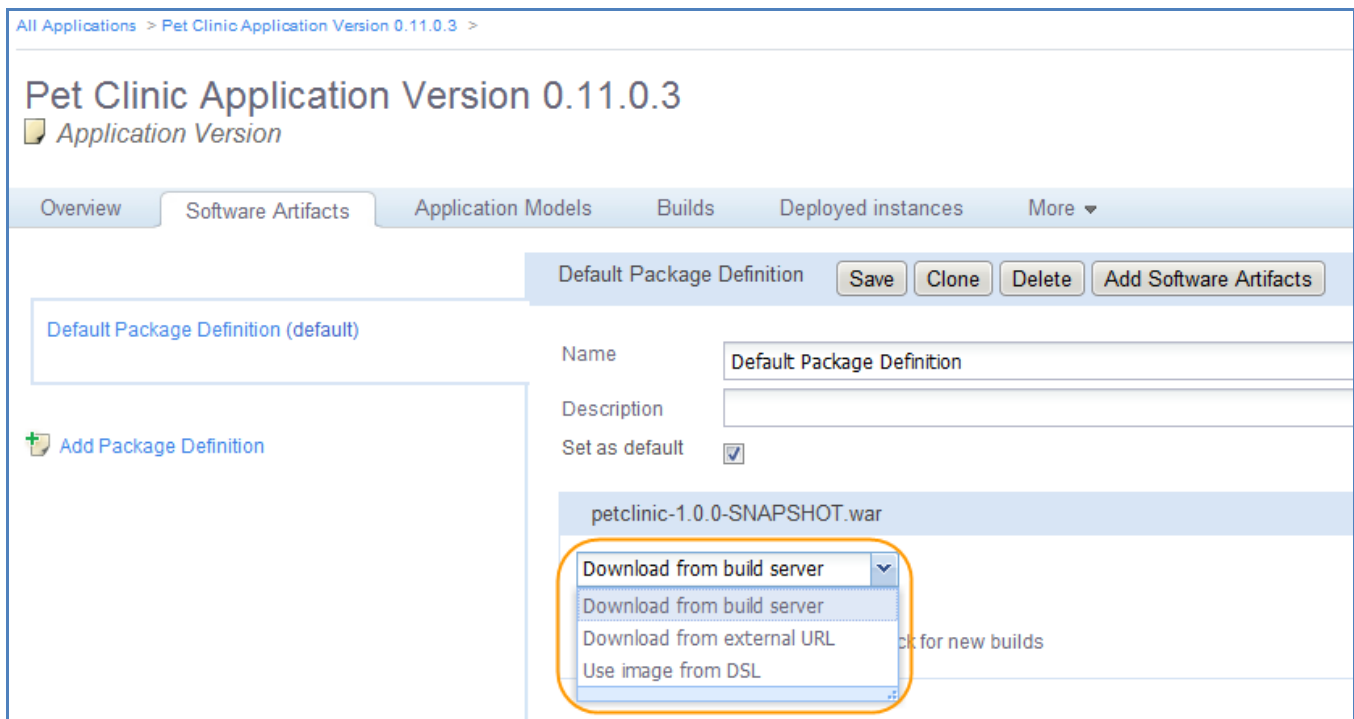


The Software Artifacts tab of the Application Version window offers:

- **Default Package Definition:** The default package definition that is used in the deployment stage.
- **Add Package Definition:** Create a new package definition to make it available in the deployment stage.

Note: If you want to use a build server, you must know the connection information for that build server, such as the URL and username and password, if required. When you select a build artifact to use, you must also know the artifact structure so that you can find an artifact.

See the *HP Continuous Delivery Automation Installation and Configuration Guide* for more information about Continuous Integration (CI) servers, such as Jenkins.



There are several ways you can reconfigure a software artifact that is defined in a package definition:

- Download from build server
- Download from external URL
- Use image from DSL

Download from build server

To reconfigure the software artifact by downloading from a build server:

1. Select **Download from build server** from the drop-down list.
2. Click **Browse build artifact** to connect with a build server and make your selections.
3. In the Select Build Artifact dialog, select **Latest builds** or **All builds**.
4. Click **OK** to save your selection.
5. (Optional) As the default, keep the **Download into DSL** check box selected. The first time HP CDA runs a deployment, the build artifact will be downloaded from the build server and then uploaded into the DSL. This option allows you to reuse that file in subsequent executions of the application workflow that use the build artifact.
6. (Optional) Select **Check for new builds**. During each deployment, you will get the latest build from the build server, based on your application deployment schedule. This is not based on the build server's build schedule. As a best practice, select this option to check for new builds on a daily basis, such as every morning, so that you always get the latest build that is available.

Download from external URL

To reconfigure the software artifact by downloading from an external URL:

1. Select **Download from external URL** from the drop-down list.
2. *(Optional)* As the default, keep the **Download into DSL** check box selected. The first time HP CDA runs a deployment, the build artifact will be downloaded from the URL and then uploaded into the DSL. This option allows you to reuse that file in subsequent executions of the application workflow that use the build artifact.
3. *(Optional)* Select the **Check for new builds** check box. During each deployment, you will get the latest build from the build server, based on your application deployment schedule.

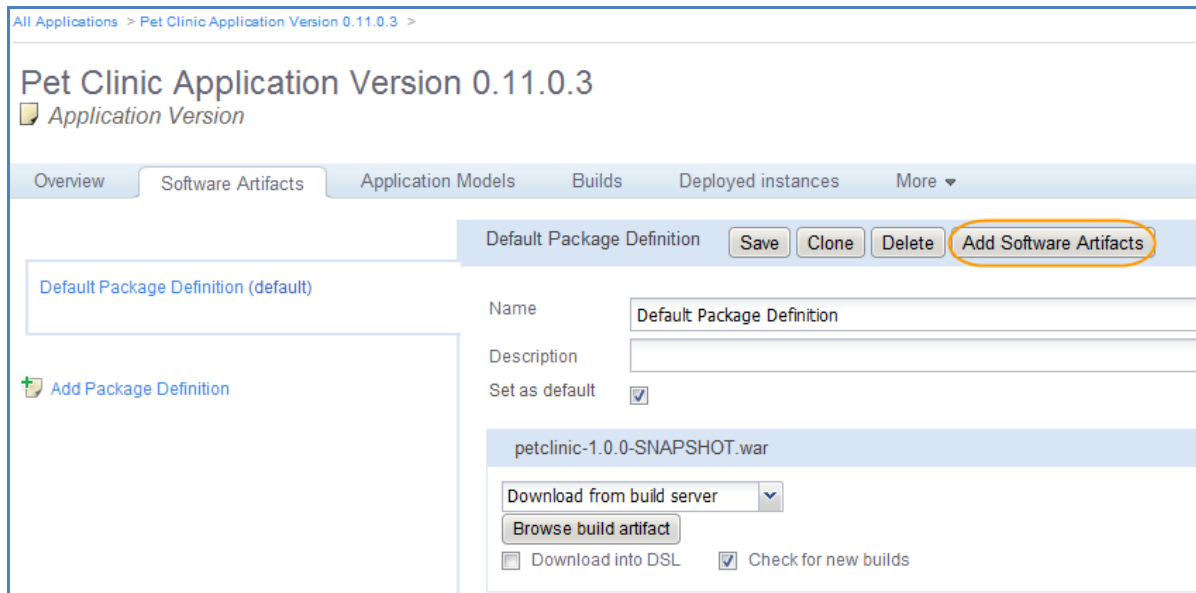
Use image from DSL

To reconfigure the software artifact by using an image from the DSL:

1. Select **Use image from DSL** from the drop-down list and specify whether you want to upload or change the image (file).
2. *(Optional)* Click **Upload**.
3. In the Attach Data Content window, enter the file name or URL.
4. Click **Upload**.
5. *(Optional)* Click **Change**.
6. In the Select Artifacts window, make your selections.
7. Click **Add**.

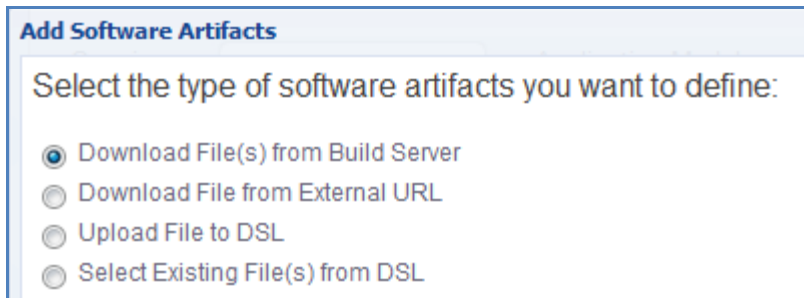
Adding Software Artifacts

In some use cases, an application requires multiple software artifacts, such as a .war (Java Web Archive) file for a Web application and a database configuration script for the DB layer. If these types of software artifacts need to be referenced separately in the workflows, you must define them as separate software artifacts.



Using the Wizard to Add Artifacts

The Add Software Artifacts wizard guides you through the steps required to add software artifacts to your application deployment.



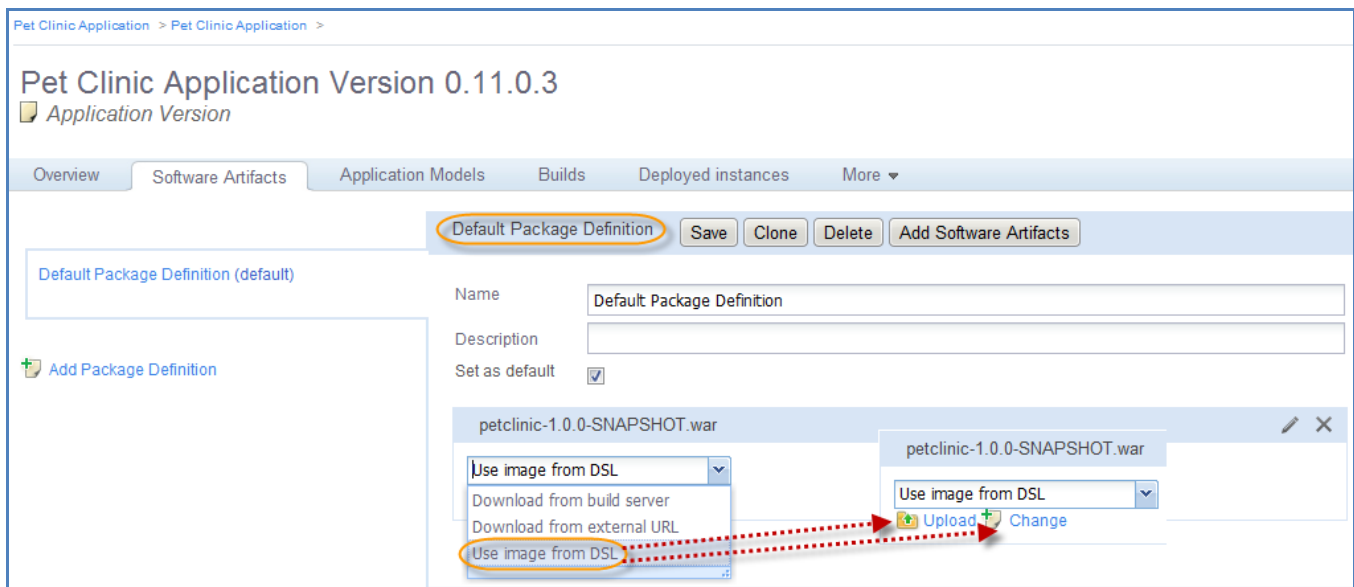
To add software artifacts using the Add Software Artifacts wizard, complete the following steps:

1. Select the type of software artifacts you want to add:
 - a. **Download File(s) from Build Server**
 - b. **Download File from External URL**
 - c. **Upload File to DSL**
 - d. **Select Existing File(s) from DSL**
2. Click **Next** to advance to the next wizard step.
3. (Optional) If you select:
 - a. **Download File(s) from Build Server:** Select **Latest builds** or **All builds** in the Add Software Artifacts window to select artifacts that you want downloaded from the build server.
 - b. **Download File from External URL:** enter the full, external URL for the artifact in the Add Software Artifacts window to specify where you want artifacts downloaded from.
 - c. **Upload File to DSL:** in the Upload a File to DSL window, specify the source file location (in your local file system) or URL, and the server folder.

- d. **Select Existing File(s) from DSL:** In the Add Software Artifacts - Use Existing Files from DSL window, select one or multiple files.
4. (Optional) As the default, keep the **Download into DSL** check box selected. The first time HP CDA runs a deployment, the build artifact will be downloaded and uploaded into the DSL. This option allows you to reuse that file in subsequent executions of the application workflow that uses the build artifact.
5. (Optional) Select the **Check for new builds** check box. During each deployment, you will get the latest build from the build server, based on your application deployment schedule. This is not based on the build server's build schedule. As a best practice, select this option to check for new builds on a daily basis, such as every morning, so that you always get the latest build that is available.
6. Click **Finish** to save your changes.

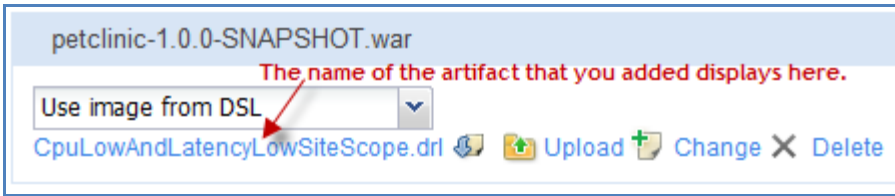
Using the Drop-down List to Add Artifacts

To add software artifacts using the drop-down list in the Default Package view, complete the following steps:



1. (Optional) If you select **Download from build server**, click **Browse build artifact**. In the Select Build Artifact dialog, select **Latest builds** or **All builds**. Click **OK** to save your changes.
2. (Optional) If you select **Download from external URL**, enter the URL. Select the **Download into DSL** or **Check for new builds** check box.
 - a. (Optional) As the default, keep the **Download into DSL** check box selected. The first time HP CDA runs a deployment, the build artifact will be downloaded from an external URL and then uploaded into the DSL. This option allows you to reuse that file in subsequent executions of the application workflow that use the build artifact. Click **Save**.
 - b. (Optional) Select the **Check for new builds** check box. During each deployment, you will get the latest build from the build server, based on your application deployment schedule. Click **Save**.
3. (Optional) If you select **Use Image from DSL**, you must also select the **Upload** link or the **Change** link.
 - a. **Upload:** In the Attach Data Content window, specify the source file location (in your local file system) or URL and the server folder. Click **Upload**.
 - b. **Change:** In the Select Artifacts window, specify the type of artifact you want to add. (Optional) Click **Add Criteria** if you need to add an artifact type or category. Make your selections and then click

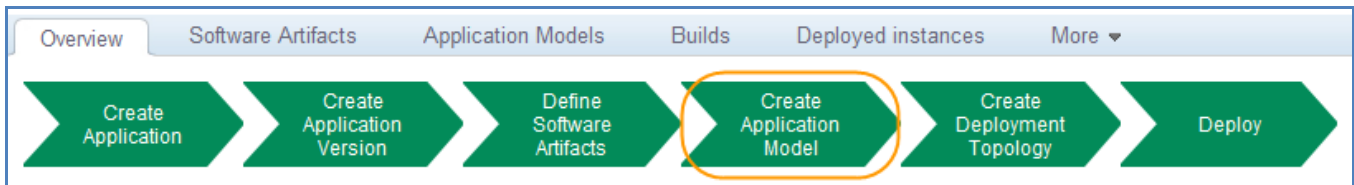
Add. The name of the artifact displays in the Default Package Definition view. Click **Change** to add the existing artifact.



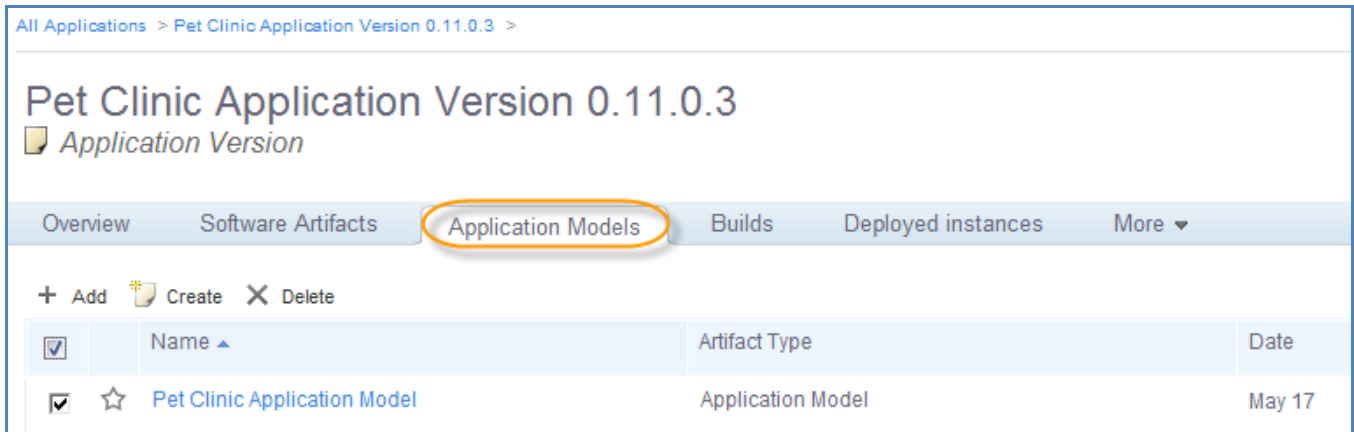
Application Model Components

The main components of the application model are its layers. A layer corresponds to a tier in a multi-tier application. An application model is a specific definition that defines the layers of the application, including the recipe for managing the deployment of the application.

Figure 5 HP CDA Model: Create Application Model




You can select which application model to work on by using the Application Models tab.



Application Model Tabs

The screenshot shows the 'Pet Clinic Application Model' page. At the top, there is a breadcrumb trail: 'All Applications > Pet Clinic Application Version 0.11.0.3 >'. Below this, the title 'Pet Clinic Application Model' is displayed with a globe icon and the text 'Application Model'. A horizontal navigation bar contains six tabs: 'Overview', 'Layers', 'Deployment Topologies', 'Deployments', 'Parameters', and 'More'. The 'Overview' tab is currently selected and highlighted with an orange border. Below the navigation bar, the following information is displayed:

- Domain: [Default](#)
- Owner:  [Administrator](#)
- Application Model for: [Pet Clinic Application Version 0.11.0.3](#)

You can control the selected application model by using the display options in your view. A selected application model provides the following tabs and drop-down list:

- Overview: See [“Overview” Application Model Tab](#) on page 50.
- Layers: See [“Layers” Application Model Tab](#) on page 51
- Deployment Topologies: See [“Deployment Topologies” Application Model Tab](#) on page 52.
- Deployments: See [“Deployments” Application Model Tab](#) on page 53.
- Parameters: See [“Parameters” Application Model Tab](#) on page 53.
- More: See [“More” Application Model Tab](#) on page 54.

“Overview” Application Model Tab

Information displayed in the Overview tab is automatically populated, based on the type of user who performs the action. For example, when you are logged in as an administrator, HP CDA assigns the **Domain** to Default.

Pet Clinic Application Model
Application Model

Overview Layers Deployment Topologies Deployments Parameters More ▾

Domain: System
Owner: Administrator
Application Model for: Pet Clinic Application Version 0.11.0.3

Latest Events

- Cloning Application Model finished successfully New Administrator, May 24
- Cloning Application Model finished successfully New Administrator, May 21
- Application deployment was performed New Administrator, May 17

More ...

The Overview tab displays the following information about your application model:

- **Domain:** The domain that the application model is part of.
- **Owner:** The person who is creating the artifact.
- **Application Model for:** The application version that is associated with the application model. You can have multiple application models for a given application version.

“Layers “Application Model Tab

Within the application model, you must define the layers of the composite applications for deployment. For each layer, you define specific endpoint services that the layer exposes to other layers. A layer can also define its connections to the endpoints provided by other layers in the application model.

All Applications > Pet Clinic Application Version 0.11.0.3 >

Pet Clinic Application Model
Application Model

Overview Layers Deployment Topologies Deployments Parameters More ▾

+ Add Layer

- ▶ DB Layer
- ▶ App Server Layer

After you model the layers and package definitions, you must model all workflows per layer to manage the deployment. A workflow consists of the following components:

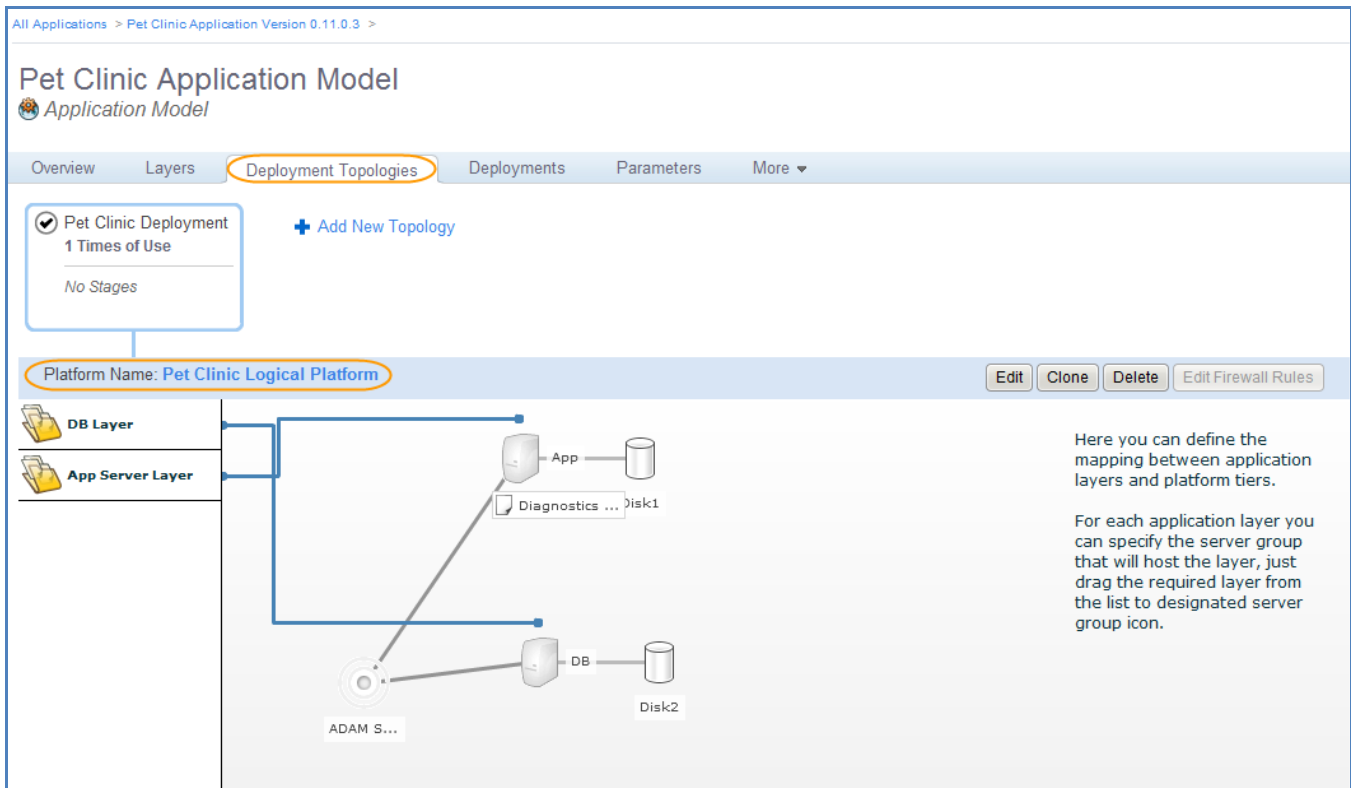
- **Placed File:** Based on the application version’s package definition, locate the files, and then define location, including file name, and permissions on the target.
- **Executed Script:** Execute a local script that is appropriate for the target’s operating system type on the target.
- **Service Command:** Execute a local service command on the target.
- **External Component:** Execute a remote sequence automation from an external service, such as an HP Operations Orchestration (HP OO) workflow or an Opscode Chef cookbook. See the *HP Continuous Delivery Automation Installation and Configuration Guide* for more information about deployment providers.

Note: These components are supported only when you can interact with the application being deployed by using the command line. When you cannot interact with the application being deployed by using the command line, use a different tool that automates user interface actions on remote systems or use HP CDA to control a scriptable tool that automates user interface interactions.

“Deployment Topologies” Application Model Tab

In HP CDA, the topology is required to map the application model’s layers to server groups in a certain platform. This mapping information specifies which sets of steps (layers) must be run on which targets (server groups).

The Deployment Topologies tab shows the topologies that have already been defined for an application model and allow you to create new ones. As a best practice, create different topologies for different stages, such as Development, QA, Staging, or Production.

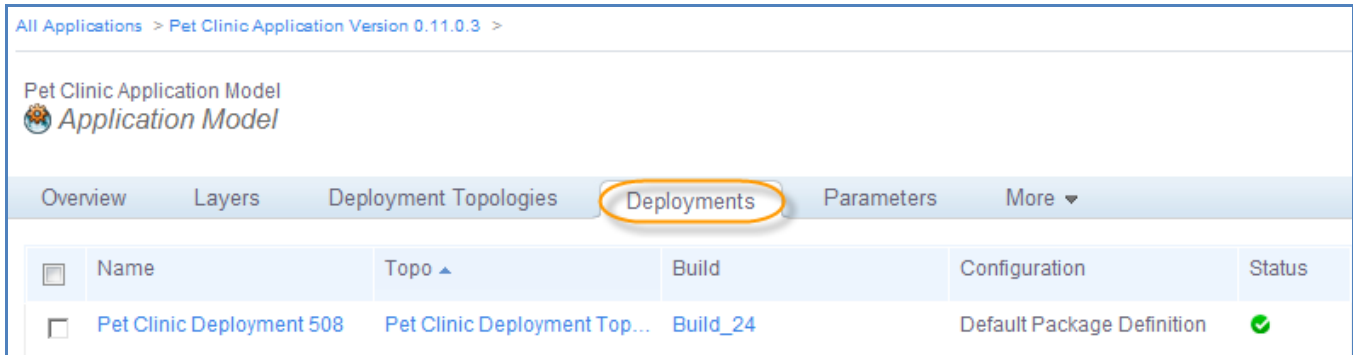


See [Deploying Applications](#) on page 26 for more information.

“Deployments” Application Model Tab

In HP CDA, you can manage deployments (application models) that have already been deployed. After you deploy your application, you will (at some point) want to execute another workflow action on the deployment, such as Start, Stop, or Undeploy. The Deployments tab lists the deployments for an application model.

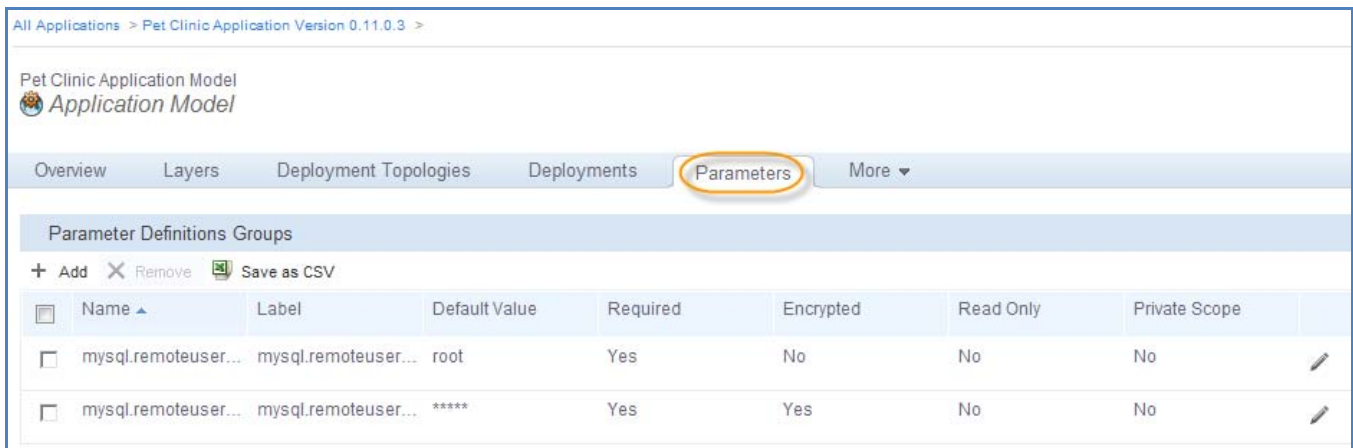
Note: You can also view the status of a deployment in the Application Version view. This status shows whether the application deployment is successful (✔) or failed (✘).



See [Deploying Applications](#) on page 26 for more information.

“Parameters” Application Model Tab

In HP CDA, you can define parameters at the application model level. These parameters are visible to all components in all layers in this application model. As a best practice, define these parameters in the Parameters tab so that they can be re-used. This practice is commonly referred to as parameterizing values—values that will be needed in several places in HP CDA.



See [Parameters and Parameterization](#) on page 12 for more information.

“More” Application Model Tab

The screenshot shows the 'Pet Clinic Application Model' interface. The 'More' dropdown menu is open, displaying the following options: Stored Parameters, Specification, Access Rights, History, and Events. The 'More' button is circled in orange.

Overview	Layers	Deployment Topologies	Deployments	Parameters	More
Domain:	System			Stored Parameters	
Owner:	Administrator			Specification	
Application Model for:	Pet Clinic Application Version 0.11.0.3			Access Rights	
Latest Events				History	
Cloning Application Model finished successfully				Events	
Cloning Application Model finished successfully					

The More drop-down list displays additional types of information about an application model, such as:

- **Stored Parameters:** Groups of values for parameters required for deployment. Stored parameters can be used to deploy multiple times with the same parameter values.
- **Specification:** Information (in a text box) that is shared with other users, such as when an application developer wants to share information with another application developer. You can also use the Attach Document action to share information with other users. See [“Overview” Application Model Tab](#) on page 50 for more information.

Application Model Actions

In HP CDA, you can perform many actions on an application model by selecting an action from the following pane.

The screenshot shows the Application Model Actions pane with the following actions:

- Deploy
- Add to Favorites
- Edit
- Clone
- Delete
- Export
- Feeds
- Send Message
- Change Domain

You can choose from the following application model actions:

- **Deploy:** Launches the deployment wizard for this application model. See [Deploying Applications](#) on page 26 for more information.
- **Add to Favorites:** Marks the artifact as a “favorite” in list views that support displaying this information, and causes the artifact to appear in lists of “favorites” of that artifact type.
- **Edit:** Edit artifact properties, such as the name and description.
- **Clone:** Clones the entire application model that includes all of the information you have added, including layers, topologies, and parameters. As a common use case, you create a copy of (clone) the application model and then make minor changes to the clone to produce the next version of the application model or to produce a different application model for a certain application version. As a best practice, cloning enables you to easily make changes when there are minor changes between the application models and you want to replicate information.
- **Delete:** Deletes the artifact from the DSL.
- **Export:** Exports the artifact to a model archive that can later be imported for a different installation of HP CDA. Exports artifacts and associated sub-artifacts into a single archive file. By default, **Include Dependencies** is checked, which means that the export process follows aggregation relationships. Uncheck this option if you want to export aggregates without their dependencies.
- **Change Domain:** Publish artifacts, subartifacts, and dependencies to a certain target domain. As the administrator who originally created the artifacts, by default you manage access and permissions for this domain.

Editing an Application Model

You can change the artifact properties of an application model.

To edit properties, complete the following steps:

1. Go to the Application Version view.
2. Select the **Application Models** tab.
3. Select an application model name.
4. In the actions pane, select **Edit**.
5. Change the name or description.
6. Click **Save**.

Cloning an Application Model

You can clone the entire application model, including all information you have added, such as layers, topologies, and parameters. As a common use case, you create a copy of (clone) the application model and then make minor changes to the clone to produce the next version of the application model or to produce a different application model for a certain application version. Cloning enables you to easily make changes when there are minor changes between the application models and you want to easily replicate information.

To clone an application model, complete the following steps:

1. Go to the Application Version view.
2. Select the **Application Models** tab.
3. Select an application model name.

4. In the actions pane, select **Clone**.
5. In the Clone dialog, click **OK** to confirm that you want to clone the application model.
6. The New Events dialog indicates that cloning is in progress. Wait until loading finishes.

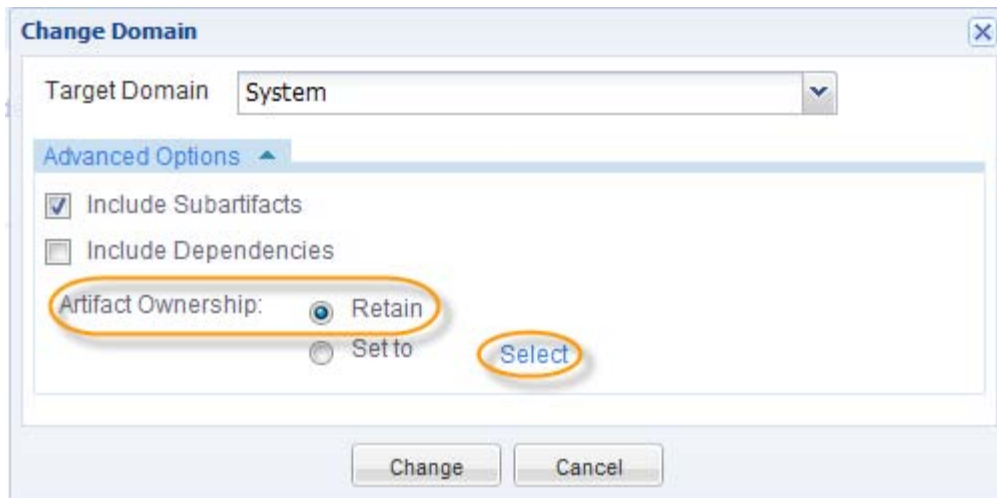
Note: HP CDA produces a copy of the application model and names it “Clone of <xxx> Application Model”, where <xxx> is the name of your original model. As a best practice, rename the clone to a meaningful name that avoids ambiguity. Use **Edit** in the Application Model Actions pane to rename a cloned application model.

Changing the Domain for an Application Model

You can publish artifacts, subartifacts, and dependencies for an application model to a certain target domain. As the administrator who originally created the artifacts, by default you manage access and permissions for this domain.

To change the domain, complete the following steps:

1. Go to the Application Version view.
2. Select the **Application Models** tab.
3. Select the application model name.
4. In the actions pane, select **Change Domain**.



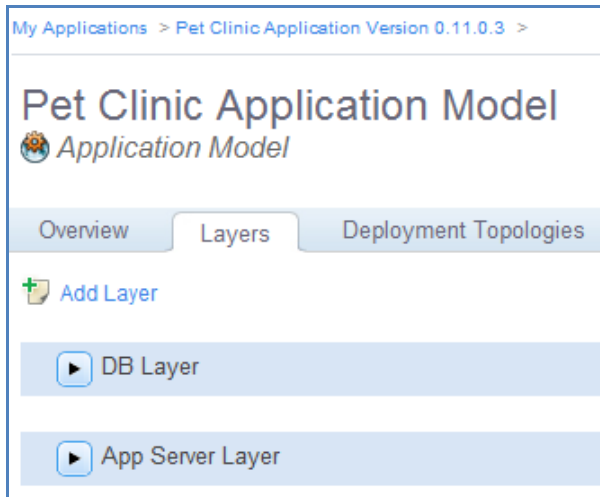
5. In the Change Domain dialog, select a Target Domain, such as System or Default. If you create custom domains, they will also display in this drop-down list.
6. (Optional) Expand **Advanced Options**.
7. Check the check boxes for including sub artifacts and dependencies, as needed.
8. Make sure that Retain is set as the Artifact Ownership value.
9. (Optional) If you need to reassign ownership, click the [Select](#) link to make changes.
10. In the Change Domain dialog, click **Change** to save your changes.

Defining an Application Layer

Multiple layers are required when different servers require different steps executed on them for the same workflow. When the application is multi-tier, it is recommended that the application designer use a separate layer for each tier, even if some or all of the tiers may ultimately be deployed to the same server group.


In this Pet Clinic Application Model example, some servers become database servers and some servers become application servers:

- **DB Layer:** The database servers run MySQL and host the data that the web applications display.
- **App Server Layer:** The application servers run Tomcat and serve the Pet Clinic web application.



Note: The steps executed on the database server group (DB Layer) are different from the steps executed on the application server group (App Server Layer).

To define an application layer:

1. Go to the Application Model view.
2. Select the **Layers** tab.
3. Select the  **Add Layer** link.
4. In the Layer Name dialog, enter a name, and then click **Add**.
5. Expand the layer that you just created.
6. In the Workflows tab, define the actions that will be executed, such as Deploy, Undeploy, Start, and Stop. See [Workflow Actions & Programming Operations](#) on page 58 for more information.
7. (Optional) Expand the [Add Workflow](#) link to define custom workflows, such as Pause, Restart, and Backup. See [Workflow Actions & Programming Operations](#) on page 58 for more information.
8. In the Parameters tab, specify the parameter definition groups for workflows and steps. You can also define parameters at a higher level to share information across multiple application layers, such as a host name or an IP address. See [Parameters and Parameterization](#) on p. 12 for more information.
9. In the Requirements tab, select the [Change](#) link to define the hardware, operating system, and software prerequisites for this application layer. These requirements include RAM, disk storage, OS version, and software version. These requirements are included in the deployment topology.

10. In the Connections tab, define one or more endpoints and connections. Endpoints are exposed by this layer, using a port and an HTTP, HTTPS, SSL, or TCP protocol, and (optionally) a path. Connections are defined as **to Layer**.
 - a. Select the [Endpoints](#) link and then select **Create New Endpoint**. In the Add window, enter a name and port. *(Optional)* Select the **Load Balanceable** check box to specify that the layer will be load balanced. Click **OK** to save your changes.

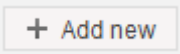
Note: If you choose load balancing by selecting the **Load Balanceable** checkbox, you must enter load balancer parameter values in the deployment topology map. For details and instructions, see [Deployment Topology and Load Balancing](#) on page 68.
 - b. Select the [Add Connection](#) link and then select **to Layer**.
 - c. In the Select Artifact window, specify the endpoint.

Application Workflow

An application workflow is a sequence of actions that are performed on an application. In HP CDA, the predefined application workflows are Deploy, Undeploy, Start, and Stop. You can also create custom application workflows, such as Pause, Restart, and Backup. A custom application workflow is commonly known as a user-defined application workflow.

Workflow Actions & Programming Operations

An application workflow performs the following actions:

- **Deploy:** Programming operations for Placed File, Executed Script, Service Command, and External Component. See [Application, Application Version, & Application Model](#) on page 39 for descriptions of these operations and their parameters.
- **Undeploy:** Programming operation that deletes an application database, stops deployment scripts, and restores configuration scripts.
- **Start:** Programming operation that is performed on the application to start it.
- **Stop:** Programming operation that is performed on the application to stop it.
- **Add Action:** Custom (user-defined) programming operations actions, such as Pause, Restart, and Backup.
-  Click this button to add components to the selected workflow, such as custom (user-defined) programming operations for Placed File, Executed Script, Service Command, and External Component.

Note: When you add a new External Component, choose a component type (Plugin) and a plugin configuration, such as Chef Recipe, DMA Flow, OO Flow, SA Package, or SA Policy. In the Action tab, click **Browse** to select an external component. For descriptions of these input component parameters and setup instructions, see [Table 7 External Component Screen Parameter Settings](#) explains the settings to be made in that screen, with instructions on how to set them where necessary. .

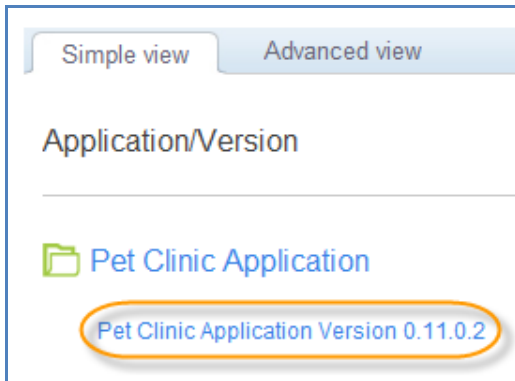
Table 7 External Component Screen Parameter Settings on page 90.

For more information about parameters, see [Parameters and Parameterization](#) on p. 12.

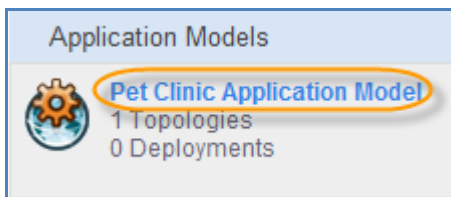
Accessing an Application Workflow

To access an application workflow:

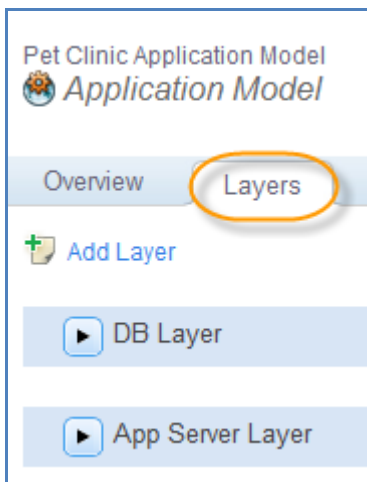
1. In the Simple View, select the application version.



2. In the Application Version view, select an application model.



3. In the Application Model view, select the Layers tab to display the collapsed layers.



4. Expand a layer to view the workflows. Wait for this information to load.
5. In the expanded layer view, select the Workflows tab to view workflows for the layer. For more information, see [Workflow Actions & Programming Operations](#) on page 58.

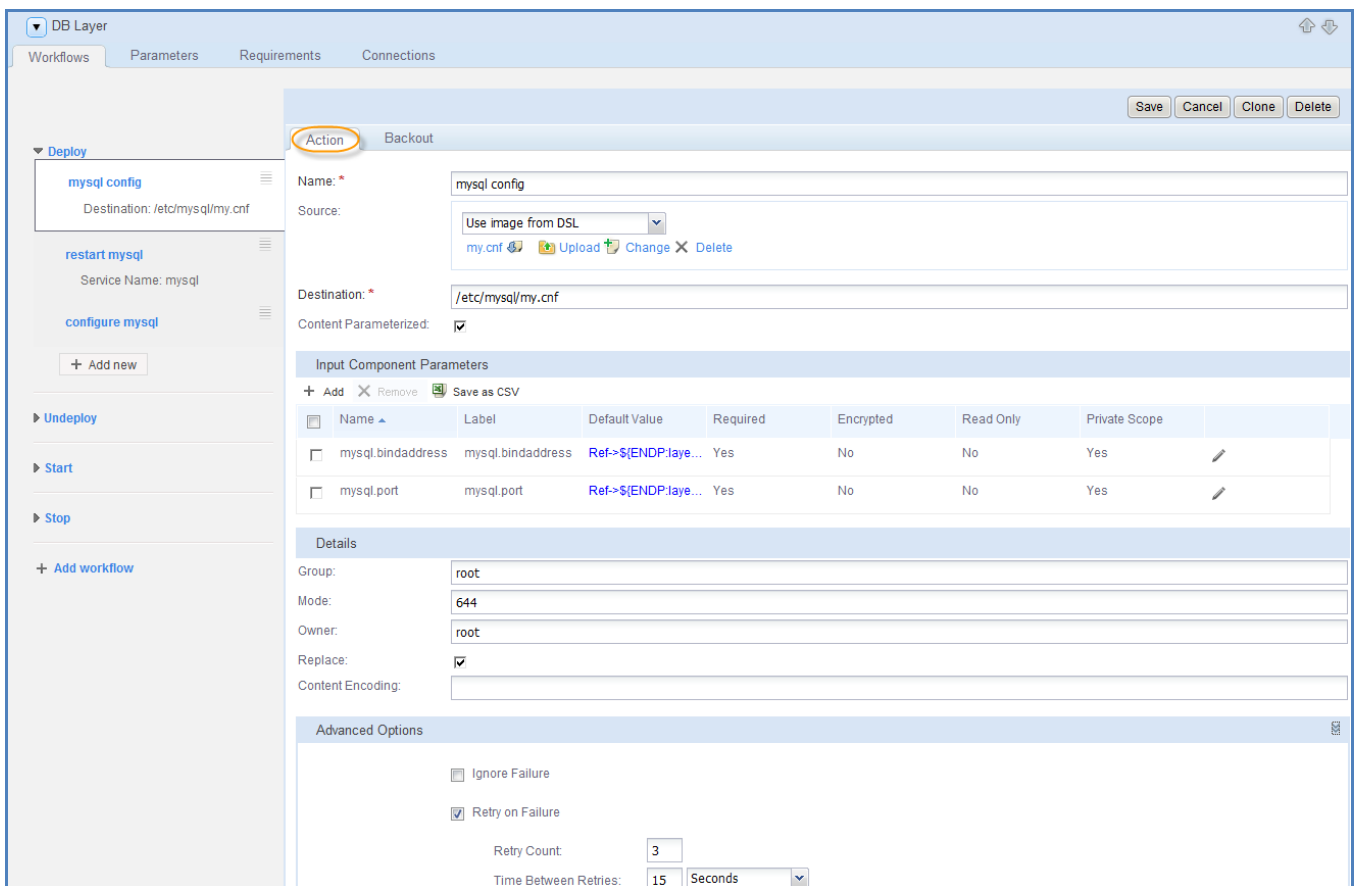
Configuring a Workflow Layer Action

In HP CDA, you can configure the Placed File, Executed Script, Service Command, and External Component workflow actions at the application model layer level. See the tables that define the parameters and setup instructions for these workflow actions in [Workflow Actions & Programming Operations](#) on page 58.

The following graphic is an example of configuring the Placed File workflow action for a layer. In this use case, you want to place the `my.cnf` file in `/etc/mysql`, where you specified the following configuration:

- **Source: Use the image from DSL:** As the application developer, upload this file when you define the `Deploy > mysql config` workflow step. This file will be available directly from the DSL.
- **Destination:** This is where the file will be placed on the target machine.
- **Content Parameterized:** The content of your placed file (in this example, the image you upload into the DSL) contains parameters that HP CDA must search for and replace.
- **Replace:** Communicates to HP CDA to overwrite an existing file that is in the same location. In this example, if `/etc/mysql/my.cnf` already exists and you select the **Replace** check box, the existing file on the target will be overwritten with the specified content.

The following instructions are generic—they apply to configuring any type of layer workflow action in an application model.



To configure a workflow layer action:

1. Go to the Application Model view.
2. Select the **Layers** tab.
3. Expand a layer, such as the DB Layer, the App Server Layer, and so on. By default, the **Action** tab is selected.
4. Complete the setup instructions that are described in the following references:
 - For the **Placed File** action, see [Placed File Permissions](#) on page 61 and [Table 4 Placed File Screen Parameter Settings](#) on page 82.

- For the **Executed Script** action, see [Table 5](#) Executed Script Screen Parameter Settings on page 85.
- For the **Service Command** action, see [Table 6](#) Service Command Screen Parameter Settings on page 87.
- For the **External Component** action, see [Table 7](#) External Component Screen Parameter Settings explains the settings to be made in that screen, with instructions on how to set them where necessary.
- [Table 7](#) External Component Screen Parameter Settings on page 90.

5. Click **Save**.

Placed File Permissions

In the Details section for a Placed File action, you can configure certain permissions on the file that will be placed on the target server.

These types of use cases include:

- If the deployer is Opscode Chef and the target platform is Linux, complete this section to set up Linux-style permissions.
- If the deployer is HP Server Automation running on a Microsoft Windows or a Linux platform, complete this section to set up platform-based permissions.

Note: If the deployer is Opscode Chef and the target platform is Microsoft Windows, the Details section does not apply.

Configuring a Workflow Backout

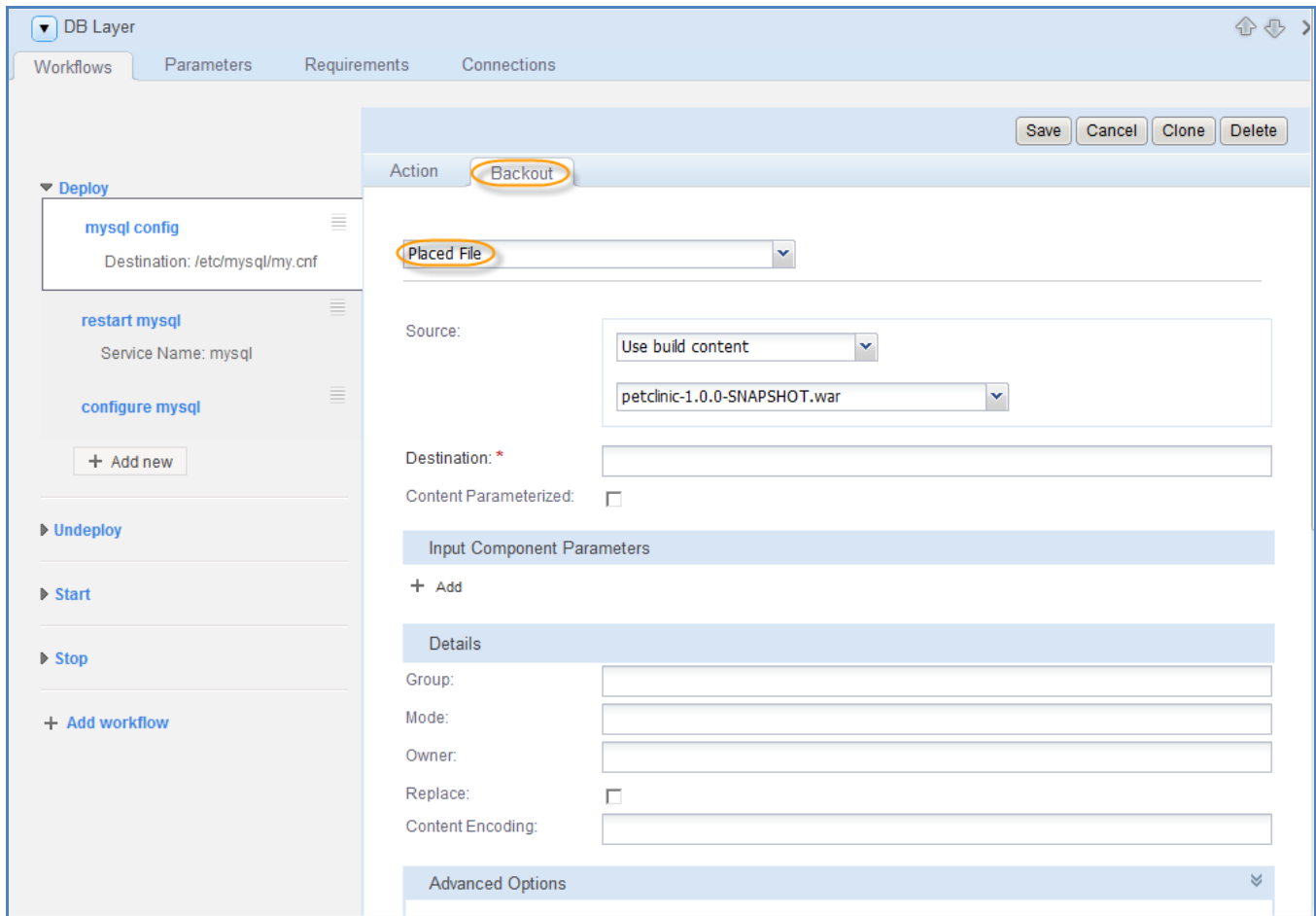
In this use case, you define the steps that are required to clean up in the event of a failure during execution of the workflow. This process will back out actions that previously completed.

HP CDA supports one backout step per component. For example, if you have a step to place the `my.cnf` file, you can have only one backout step to remove it from where it was placed. See [Backing Out of Failed Operations](#) on page 31.

The following graphic is an example of defining the backout steps that are required to clean up in the event of a Placed File action failure. This will back out actions that previously completed.

Note: HP CDA supports one custom action per one backout step. For example, if you have a step to place the `my.cnf` file, you can have only one backout step to remove it from where it was placed.

The instructions provided below this graphic are generic—they apply to configuring a backout for any type of layer workflow action in an application model.



To configure a workflow layer backout operation:

1. Go to the Application Model view.
2. Select the **Layers** tab.
3. Expand a layer, such as the DB Layer, the App Server Layer, and so on.
4. Select the **Backout** tab.
5. Complete the setup instructions and **Advanced Options** that are described in the following references:
 - For the **Placed File** action, see [Placed File Permissions](#) on p. 61 and [Table 4](#) Placed File Screen Parameter Settings on page 82.
 - For the **Executed Script** action, see [Table 5](#) Executed Script Screen Parameter Settings on page 85.
 - For the **Service Command** action, see [Table 6](#) Service Command Screen Parameter Settings on page 87.
 - For the **External Component** action, see [Table 7](#) External Component Screen Parameter Settings explains the settings to be made in that screen, with instructions on how to set them where necessary. .
 - [Table 7](#) External Component Screen Parameter Settings on page 90.
6. Click **Save**.

Input Component Parameters

In HP CDA, you can add input components at different levels of the application model:

- At the workflow action level, such as in the `configure mysql` script.
- At the layer level, such as the DB Layer, App Server Layer, and so on.
- At the top of the application model.

As a best practice, add these parameters so that you can reuse them for multiple deployments.

See [Parameters and Parameterization](#) on page 12 for information about how the `@{name-of-parameter}` Substitution Variable is used in HP CDA.

Adding an Input Component Parameter to a Workflow

In the following example, the `@{mysql.dbname}` Substitution Variable in the “`configure mysql`” script will be replaced by the Input Component Parameter value you defined.

Pet Clinic Application Model

Application Model

Overview Layers Deployment Topologies Deployments Parameters More ▾

+ Add Layer

DB Layer

Workflows Parameters Requirements Connections

Deploy

- mysql config
Destination: /etc/mysql/my.cnf
- restart mysql
Service Name: mysql
- configure mysql**

+ Add new

Undeploy

Start

Stop

+ Add workflow

Action Backout

Name: * configure mysql

Path: /usr/bin

Precondition:

User:

Working Directory:

Content

```
#!/bin/sh
mysql -e "CREATE DATABASE IF NOT EXISTS @mysql.dbname;" -u @mysql.localuser.name -p@mysql.localuser.password
mysql -e "GRANT ALL ON @mysql.dbname.* TO '@mysql.remoteuser.name'@mysql.remoteuser.password;" -u @mysql.localuser.name -p@mysql.localuser.password
exit 0
```

Input Component Parameters

+ Add ✕ Remove Save as CSV

<input type="checkbox"/>	Name ▲	Label	Default...	Required	En
<input type="checkbox"/>	mysql.db...	mysql.dbname	Ref->\${E...	Yes	No
<input type="checkbox"/>	mysql.lo...	mysql.localuser.name	Ref->\${L...	Yes	No
<input type="checkbox"/>	mysql.lo...	mysql.localuser.pass...	****	Yes	Ye

To add an input component parameter to a workflow:

1. Go to the Application Model view.
2. Select the **Layers** tab.
3. Expand a layer and then select its **Workflow** tab.
4. Expand a workflow action, such as Deploy, Undeploy, Start, or Stop.
5. In the Action tab, in the Input Component Parameters view, click **+ Add**.
6. Complete the Add dialog. See [Parameters and Parameterization](#) on page 12 for information about how the `@{name-of-parameter}` Substitution Variable is used in HP CDA.
7. Click **Add** to save the parameter.

Adding an Input Component Parameter to a Layer

The screenshot shows the 'Pet Clinic Application Model' interface. The 'Layers' tab is selected in the top navigation bar. Under the 'DB Layer', the 'Parameters' tab is selected. Below this, there is a section for 'Parameter Definitions Groups' with an '+ Add' button circled in red. Below the table, there is an 'App Server Layer' section.

Name	Label	Default Value	Required	Encrypted	Read Only	Private Scope
mysql.localuser...	mysql.localuser...	root	Yes	No	No	Yes
mysql.localuser...	mysql.localuser...	*****	Yes	Yes	No	Yes

To add an input component parameter to a layer:

1. Go to the Application Model view.
2. Select the **Layers** tab.
3. Expand a layer and then select its **Parameters** tab.
4. In the Parameter Definitions Group view, click **+ Add**.
5. Complete the Add dialog. See [Parameters and Parameterization](#) on page 12 for information about how the `@{name-of-parameter}` Substitution Variable is used in HP CDA.
6. Click **Add** to save the parameter.

Adding a Parameter Definitions Group to an Application Model

The screenshot shows the 'Pet Clinic Application Model' interface. The 'Parameters' tab is selected and circled. Below the navigation tabs, there is a section for 'Parameter Definitions Groups' with '+ Add', 'Remove', and 'Save as CSV' buttons. A table lists two parameter definitions:

<input type="checkbox"/>	Name	Label	Default Value	Required	Encrypted	Read Only	Private Scope	
<input type="checkbox"/>	mysql.remoteuse...	mysql.remoteuse...	root	Yes	No	No	No	
<input type="checkbox"/>	mysql.remoteuse...	mysql.remoteuse...	*****	Yes	Yes	No	No	

To add an input component parameter at the top of the application model:

1. Go to the Application Model view.
2. Select the **Parameters** tab.
3. In the Parameter Definitions Group view, click **+ Add**.
4. Complete the Add dialog. See [Parameters and Parameterization](#) on page 12 for information about how the `@{name-of-parameter}` Substitution Variable is used in HP CDA.
5. Click **Add** to save the parameter.

Connections & Endpoints

Connections and endpoints capture how an application layer connects to other layers and how other layers can connect to it. In HP CDA, when you have a multi-tier application, the layers often need to interact with each other—to pass information from one layer to another layer. Connections and endpoints allow the application designer to define which layers interact and which port, protocol, and path to use.

Note: Creating or deleting a connection or endpoint allows you to reference the values through parameters within workflow actions and application model layers. This does not execute a command on deployment.

Creating, Modifying, & Deleting Endpoints

For each layer in an application model, you can define, modify, and delete certain endpoint services that the layer exposes to other layers.

Endpoints are exposed by their relative layer, using a port, path, and an HTTP, HTTPS, SSL, or TCP protocol.

As a best practice, select the **Load Balanceable** check box when the endpoint service allows load balancing.

To create or modify an endpoint:

1. Go to the Application Model view.
2. Select the Layers tab.
3. Expand a layer.
4. Select the Connections tab.
5. Expand **Endpoint(s)**.

6. *(Optional)* To edit or remove an endpoint that has already been defined, select the endpoint and then select **Edit** or **Remove**.
7. *(Optional)* To create a new endpoint, select **Create New Endpoint**.
8. In the Add dialog box, enter a name and the port.
9. *(Optional)* Select the **Load Balanceable** check box to specify that the layer will be load balanced. By default, this option is not selected.

Note: If you choose load balancing by selecting the **Load Balanceable** checkbox, you will need to enter load balancer parameter values in the deployment topology map. For details and instructions, see [Deployment Topology and Load Balancing](#) on page 68.

10. Click **OK** to save your changes.

Defining and Deleting Connections

In HP CDA, connections are defined as **to layer**. Connection definitions help you understand the direction of communication, such as at what port and at what protocol. This information can help you define your firewall rules.

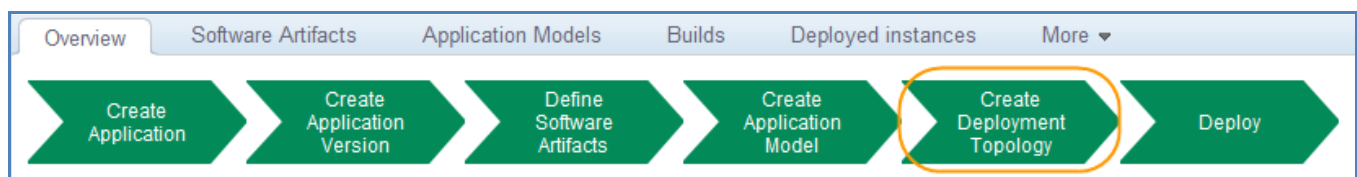
To define or modify a connection:

1. Go to the Application Model view.
2. Select the **Layers** tab.
3. Expand a layer.
4. Select the Connections tab.
5. Expand Connection(s).
6. *(Optional)* To delete a connection that is already defined, select the connection and then select Remove. You cannot edit a connection—either the layer connects to an endpoint or it does not connect to an endpoint. You cannot specify other details for a connection.
7. *(Optional)* To add an endpoint, select **Add Connection**.

Deployment Topology

A deployment topology defines a unique deployment mapping. A topology shows the dynamic relationship between composite application layers and compute resources within a certain platform.

Figure 6 HP CDA Model: Create Deployment Topology



Modeled Relationships

When you create an application model, a topology relationship dynamically maps the layers of a specific application model to the compute resources of a platform. This is a modeled relationship that enables an application version to participate in many topologies with heterogeneous platforms. This relationship also provides the capability that a given platform model can be used as the target for many application models.

After applications, application versions, and application models are created in HP CDA, application designers create topology relationships to support the release. A certain topology can be deployed multiple times across multiple provisioned platforms.

The screenshot shows the 'Pet Clinic Application Model' configuration in the HP CDA interface. The 'Deployment Topologies' tab is selected, showing a topology named 'Pet Clinic Logical Platform'. The diagram illustrates the mapping between application layers and platform tiers. On the left, there are two layers: 'DB Layer' and 'App Server Layer'. On the right, there are two server groups: 'ADAM S...' and 'DB'. The 'App Server Layer' is mapped to the 'ADAM S...' server group, which hosts an 'App' component connected to 'Diagnostics ... Disk1'. The 'DB Layer' is mapped to the 'DB' server group, which hosts a 'DB' component connected to 'Disk2'. A text box on the right explains that users can define the mapping between application layers and platform tiers by dragging layers to designated server group icons.

See [Deploying Applications](#) on page 26 for more information.

Deployment Topology and Load Balancing

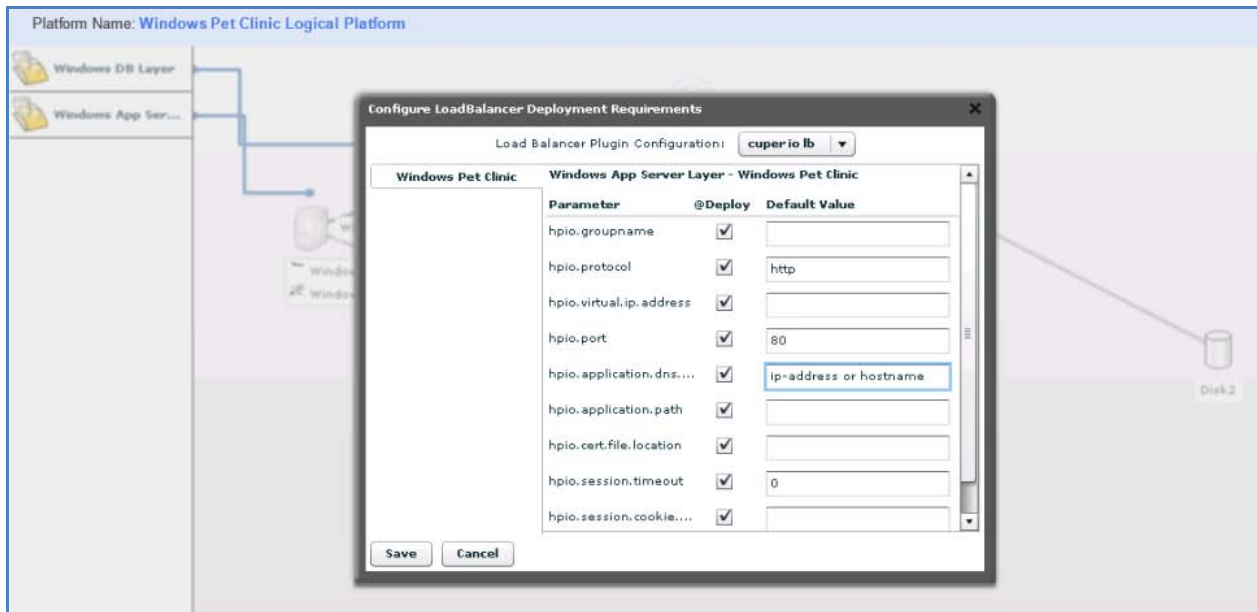
In order to model load balancing of an application layer, create an endpoint in the layer and select the **Load Balanceable** checkbox. You find this step in the following section:

- [Defining an Application Layer](#) on page 57.
- [Creating, Modifying, & Deleting Endpoints](#) on page 66.
- [Adding Connections](#) on page 103.

If you chose load balancing, you will need to enter load balancer parameter values in the deployment topology map. The following procedure explains how to do this.

To specify load balancer parameter values:

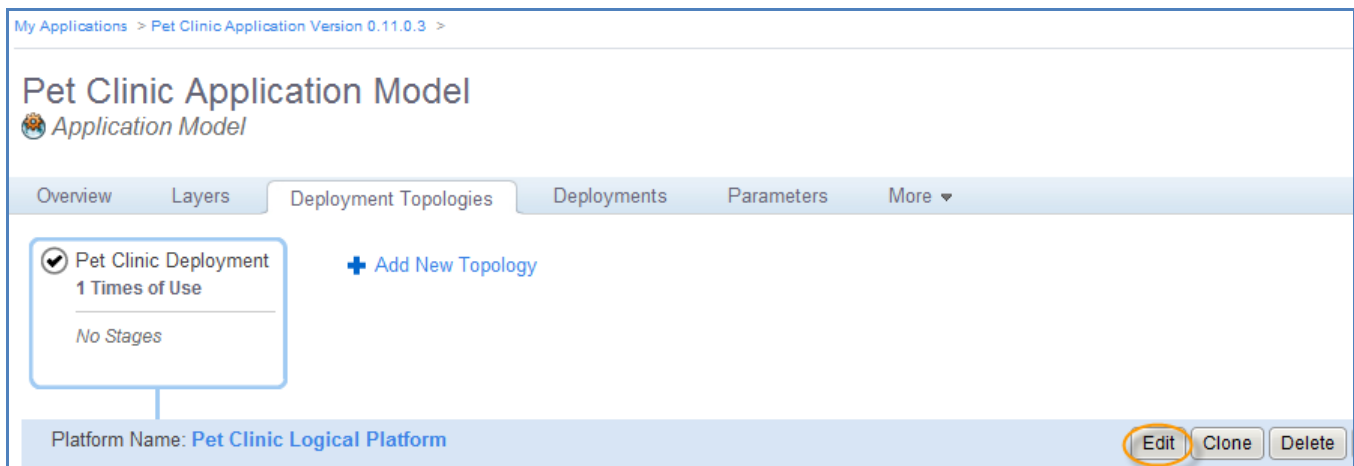
1. In the deployment topology map, click on the server group symbol for a server group for which you want to apply load balancing. If load balancing is enabled on the layer that is mapped to the selected server group, a "Configure LoadBalancer Deployment Requirements" dialog box appears, like the following figure:



2. Provide the following information for the value of each parameter in the dialog box:
 - **hpio.groupname**: The name for the load balancer group to be configured.
 - **hpio.protocol**: The communication protocol (http, https, etc.) that the application service will use.
 - **hpio.virtual.ip.address**: The virtual IP address to be used by the load balancer to receive requests.
 - **hpio.port**: The port on which the load balancer will be listening.
 - **hpio.application.dns....**: The DNS name of the application service being load-balanced.
 - **hpio.application.path**: The URI path.
 - **hpio.cert.file.location**: Location of the SSL certificate file used (if needed) for secure connections to the load balancer.
 - **hpio.session.timeout**: The number of seconds between incoming requests that the load balancer will persist a session.
 - **hpio.session.cookie....**: The name or ID of a http(s) cookie to store session information.
 - **hpio.source.address....**: A mask for filtering source address based session persistence?
3. Click the **Save** button to accept the parameter values and close the “Configure LoadBalancer Deployment Requirements” dialog box.
4. Repeat steps 1 through 3 for each load balanceable server group in the deployment topology map.

Editing a Deployment Topology

In HP CDA, you can change the name and monitoring details of a topology.

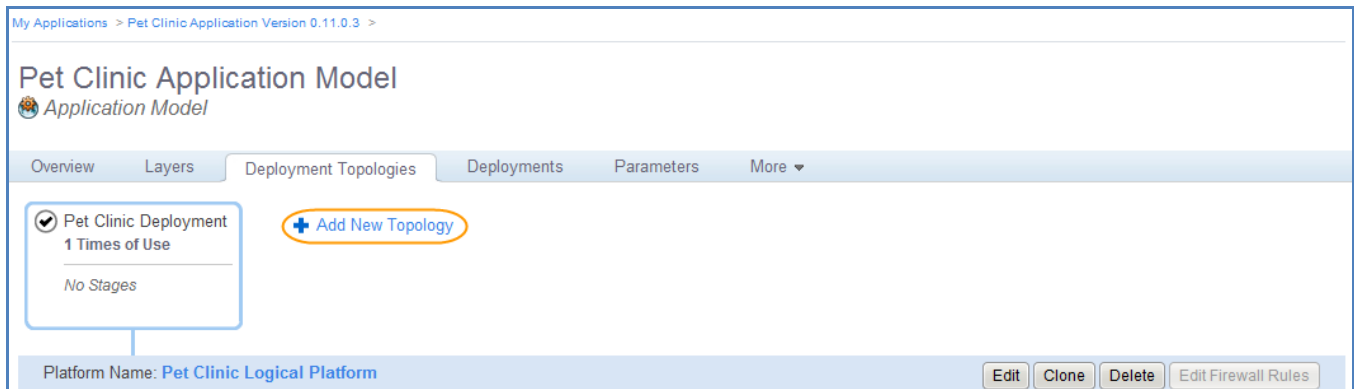


To edit a deployment topology:

1. Go to the Application Model view.
2. Select the **Deployment Topologies** tab.
3. Select an existing topology, and then click **Edit**.
4. Step through the wizard to change the name and monitoring settings.
5. Click **Finish** to save your changes.

Adding a Deployment Topology

In HP CDA, the topology is required to map the application model's layers to server groups in a certain platform. This mapping information specifies which sets of steps (layers) must be run on which targets (server groups). As a best practice, create different topologies for different stages, such as Development, QA, Staging, or Production.



To add a deployment topology:

1. Go to the Application Model view.
2. Select the **Deployment Topologies** tab.
3. Select **+ Add New Topology** to launch the wizard that will guide you through this process.

Wizard steps

- Basic Properties**
- Logical Platform
- Monitoring Policies

4. In the first step, **Basic Properties**, enter the name of the topology and specify a stage, such as Development, QA, Staging, or Production.
5. Click **Next** to view the **Logical Platform** information, such as a list of platforms that fulfill layer requirements and platforms with the selected stage, such as Development, QA, Staging, or Production.

Pet Clinic Application Model > Pet Clinic Application Model > Create Deployment Topology >

Deployment Topology

Basic Properties

Showing only platforms fulfilling layer requirements:
 App Server Layer: Tomcat 6 and newer, Linux and newer.
 DB Layer: MySQL 5.1 and newer, Linux and newer.
 Another Layer: no requirements.

Showing only platforms with stages: QA.

DO NOT USE

Pet Clinic Logical Platform

DO NOT USE

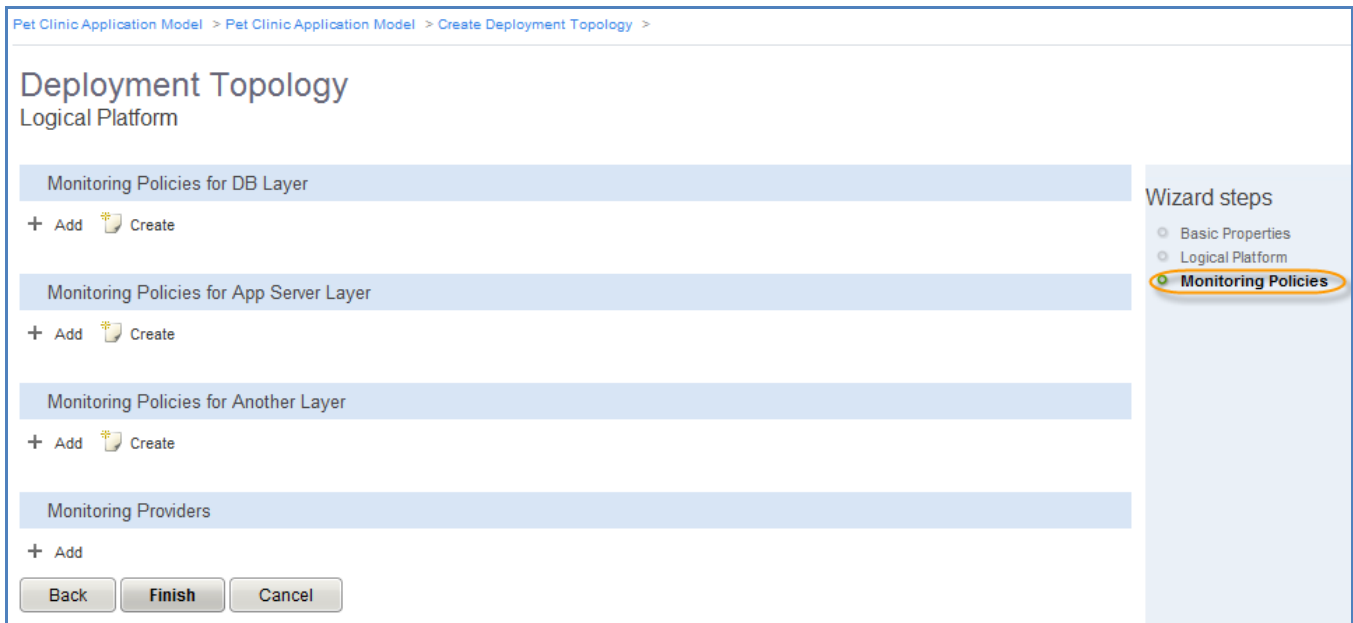
App	
Hardware:	1 CPU, 512 MB RAM
Operating System:	Linux
Software:	Tomcat 6
DB	
Hardware:	1 CPU, 512 MB RAM
Operating System:	Linux
Software:	MySQL 5.1

Back **Next** Cancel

Wizard steps

- Basic Properties
- Logical Platform**
- Monitoring Policies

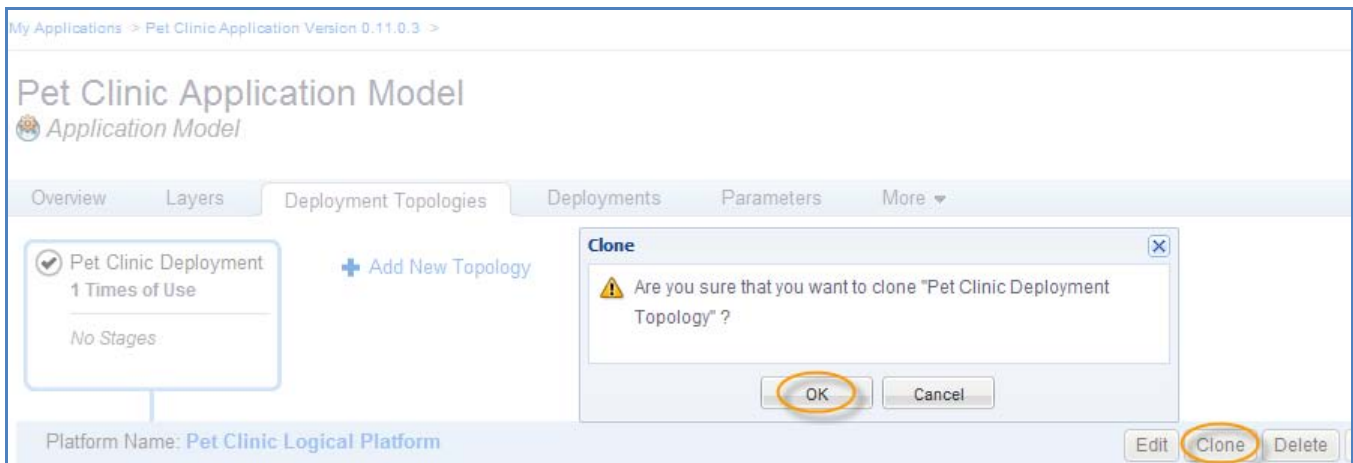
6. Click **Next** to advance to the **Monitoring Policies** step.



7. See the [Monitoring Administration Overview](#) on page 105 for instructions on how to define monitoring policies and monitoring providers, and how to assign monitoring policies to application topologies.
8. Click **Finish** to save your changes.

Cloning a Deployment Topology

In this use case, you need a topology that is very similar to an existing one. You want to clone (copy) the existing topology and make minor changes to it for a different deployment.



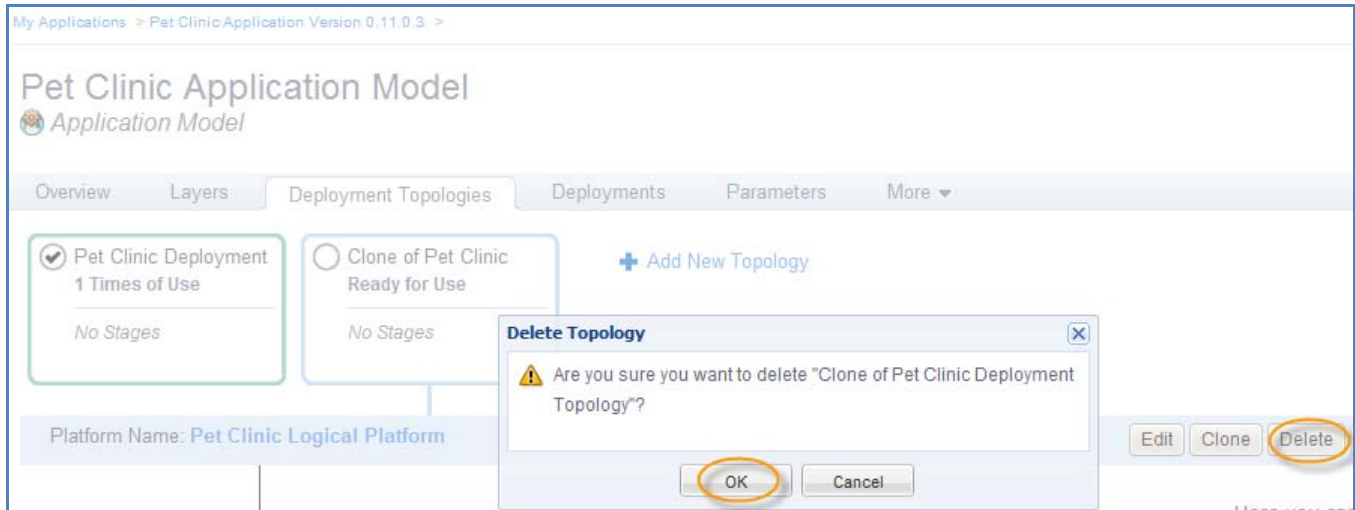
To clone an application topology:

1. Go to the Application Model view.
2. Select the **Deployment Topologies** tab.
3. Select an application model and then click **Clone**.
4. In the Clone dialog, click **OK** to confirm that you want to clone the selected application topology.

Note: HP CDA produces a copy of the deployment topology and names it "Clone of <name of original topology>". As a best practice, rename the clone to a meaningful name that avoids ambiguity. Click **Edit** to change the clone name. See [Editing a Deployment Topology](#) on page 68.

Deleting a Deployment Topology

If you no longer want to deploy your application to a given platform, you can delete the topology that maps it to that platform.



To delete a deployment topology:

1. Go to the Application Model view.
2. Select the **Deployment Topologies** tab.
3. Select an existing topology, and then click **Delete** to remove it from the DSL.
4. In the Delete Topology dialog, click **OK** to confirm this action.

4 Platform Development

The HP CDA platform developer is primarily responsible for maintaining an up-to-date collection of infrastructure templates within HP CDA, creating and managing a library of platform models, managing deployed platforms, and managing platform-related model archives. If a given application has software requirements that are not satisfied by an infrastructure template, the platform developer is also responsible for modeling the deployment of the external software onto the platform.

Platform Modeling Basics

A *platform model* describes a way to create an environment, called a *platform*, onto which one-to-many applications can be deployed. A platform is composed of one or more *tiers*, and a tier is a group of one or more servers that contain the same OS and software and reside in a common network configuration.

The platform model describes each tier and the *capabilities* that each tier provides. A tier's capabilities might come from the infrastructure template upon which the platform is modeled, or the platform model can be designed to deploy the software that provides the capabilities onto provisioned infrastructure. During the platform modeling phase, the model designer includes tags in the model to describe the resulting platform's capabilities. A platform, once created from the platform model, can host any application for which that application's dependencies are satisfied by the platform's capabilities.

The HP CDA product integrates with Infrastructure as a Service (IaaS) providers. Each provider typically presents a graphical designer in its user interface for defining the compute, storage, and network characteristics of infrastructure services. These definitions can represent compute capacity that is physical (HP Matrix hardware based) or virtual (public cloud, private cloud, or hybrid). Such infrastructure service definitions are known as *infrastructure templates*. In the current release of HP CDA, the following two sources of infrastructure templates are supported:

- HP Matrix OE Infrastructure Orchestration (IO).
- HP CDA's Existing Infrastructure provider.

Note: As mentioned, one of the sources of infrastructure templates is IO. In order for HP CDA to discover such templates, IO needs to be integrated with HP CDA. The procedure to integrate IO with HP CDA is contained in the *HP CDA Installation and Configuration Guide*, which is available as part of the HP CDA documentation suite.

The foundation of a platform model in HP CDA is an infrastructure template. Infrastructure templates typically define the core capabilities of a platform. After a new platform model has been associated with a specific infrastructure template, the modeler then uses HP CDA's tagging feature to tag the platform compute resources with built-in capabilities. The capabilities that can be tagged include any *built-in software* (database, middleware, and so on) as well as OS and hardware details. This tagging creates a complete profile of the capabilities of a given platform that can be referenced by application designers.

Platform modelers understand the underlying infrastructure templates and their capabilities. They are also aware of the unique requirements arising from the application teams when they are modeling an application. If an infrastructure template cannot provide the complete software environment for applications that will be deployed onto the resulting platform, the platform modeler can design additional software, known as *add-on software*, into the platform model. Add-on software is installed during platform provisioning time.

Platform modelers can use the HP CDA platform software library to create models of add-on software packages. This library can be leveraged to model additional platform capabilities to be added to the platform model. This ensures that the resulting platform that is provisioned from the model meets all organizational requirements and standards.

Infrastructure Template Management

Infrastructure templates serve as the starting point for platform modeling in HP CDA. They define the base compute, software, storage, and network characteristics on which a platform model can be built. Infrastructure templates are initially modeled either using the designer portion of HP Matrix OE Infrastructure Orchestration or HP CDA's Existing Infrastructure command line interface. This section explains how to manage infrastructure templates in HP CDA.

Infrastructure Templates – Important Information

This section provides important information related to infrastructure templates.

- IO-created infrastructure templates are discoverable in HP CDA as soon as they are saved in IO, regardless of whether they have been validated or published. The HP CDA interface provides no indication of the validation/publication status of a given template, however, the “Edit in HP IO” feature of HP CDA can be used to invoke the HP Matrix OE Infrastructure Orchestration Designer interface in a browser window where such status for a given template can be viewed and the template can be edited, verified, and published.
- If you plan to use HP Server Automation (HP SA) as the deployer for a given VM, HP CDA cannot be used to add the SA agent software to the platform as add-on software. One workaround is to include the SA agent software in the infrastructure template for the VM.

Manually Synchronizing IO Infrastructure Templates with HP CDA

All infrastructure templates residing on integrated IO servers are automatically discovered by HP CDA and appear on the “Synchronize IO Templates” screen. Infrastructure templates cannot be used within HP CDA until they have been imported into HP CDA via a process known as *synchronization*. Infrastructure templates can be manually synchronized at any time.

1. Invoke the “Infrastructure Templates” screen by clicking the [Infrastructure Templates](#) link that is on the **Platforms** tab in the HP CDA interface.
2. In the “Select Template Provider” drop-down field, select the name of the IO server for which you want to synchronize templates.

Note: When the “Infrastructure Templates” screen is initially invoked, the option displayed in the “Select Template Provider” drop-down field is **Existing Infrastructure Template Provider**. That option is used when managing Existing Infrastructure templates. For more information, see [Creating Infrastructure Templates Using the Existing Infrastructure](#) on page 77.

The “Infrastructure Templates” screen lists all of the discovered templates that are stored on the integrated IO server that is chosen in the “Select Template Provider” drop-down field. The list of templates is presented in two-column tabular form with the template name in the left column and the synchronization status in the right column. One of the following three icons will appear in the “Synchronization Status” column for each template:



Not Synchronized Yet

The infrastructure template has been discovered but not yet synchronized with HP CDA.



The infrastructure template has been synchronized with HP CDA and the local instance of the template in HP CDA matches the template as stored on the IO server.



Changes have been made to the template stored on the IO server that are not reflected in the local instance in HP CDA. The template needs to be resynchronized with HP CDA in order to update the local instance.

3. Select the checkboxes that are adjacent to the names of the infrastructure templates that you want to synchronize, click the **Synchronize IO Template(s)** icon, and then click **OK** in the confirmation message box that appears.

Note: To synchronize all templates listed in the table, select the checkbox that is adjacent to “Template Name” in the header of the tabular listing.

When the synchronization operation starts, a status message will appear above the screen header. As the synchronization operation proceeds, the message “Template is being synchronized” will be displayed. After synchronization completes, one of the following messages will be displayed:

- Template was synchronized
 - Template synchronization was completed with error
4. If synchronization completes with errors, click the [See report for more details](#) link to obtain details about the synchronization failure.

Editing an IO Infrastructure Template from HP CDA

To edit an IO infrastructure template, complete the following steps:

1. Invoke the “Infrastructure Templates” screen by clicking the [Infrastructure Templates](#) link that is on the **Platforms** tab in the HP CDA interface.
2. If more than one IO server has been integrated with HP CDA, select the IO server containing the infrastructure template that you want to edit in the “Select Template Provider” drop-down field.

Note: When the “Infrastructure Templates” screen is initially invoked, the option displayed in the “Select Template Provider” drop-down field is **Existing Infrastructure Template Provider**. That option is used when managing Existing Infrastructure templates. For more information, see [Creating Infrastructure Templates Using the Existing Infrastructure Provider](#) on page 77.

3. In the “Template Name” column in the “Infrastructure Templates” screen, click on the name of the template that you want to edit.

This action invokes a template-specific screen that displays the same graphical representation of the template that is displayed in the designer interface of IO.

Note: A given template must have been synchronized at least one time with HP CDA in order for you to be able to click on the template’s name and invoke its template-specific screen.

4. Click **Edit in HP IO** on the side of the screen to invoke the Designer tab of the IO designer interface in a separate browser window, with the template to be edited open in the screen. The credentials used to log in are the same credentials specified in the server configuration.
5. Edit the template as is desired in the IO designer, and then save it. Refer to the *HP Matrix OE Infrastructure Orchestration Designer* documentation for details on editing the template.

Note: After editing and saving the template, you might choose to log out of the IO designer and close the browser in which it was opened. The browser remains open otherwise.

6. Return to the browser in which the HP CDA interface is displayed, and resynchronize the template with HP CDA.

For information about template synchronization, see [Manually Synchronizing IO Infrastructure Templates with HP CDA](#) on page 75.

Creating Infrastructure Templates Using the Existing Infrastructure Provider

HP CDA currently supports the following two sources of infrastructure templates:

- HP Matrix OE Infrastructure Orchestration (IO).
- HP CDA's Existing Infrastructure provider.

Management of IO infrastructure templates has been explained above. HP CDA's Existing Infrastructure provider allows infrastructure templates to be built from existing compute resources and used as the basis for infrastructure platform models.

Compute resources (virtual, physical, or cloud-based) that have been provisioned outside of HP CDA form the basis of Existing Infrastructure templates. The basic steps involved in creating an infrastructure template are as follows:

- Represent existing compute resources as logical objects called *servers*.
- Group the compute resources having similar properties as the same OS and memory size together by defining logical objects called server pools; also known as server groups.
- Create logical objects representing the infrastructure template in HP CDA, and attach the server pool object(s) to be included in the infrastructure template.

In the current version of HP CDA, these operations are accomplished using the HP CDA CLI.

Note: Network related configuration is not part of the HP CDA CLI as it is assumed that the compute resources are already connected to the appropriate networks and are accessible.

The CLI also contains commands that can be used to edit existing templates and template components, list information related to template components, etc. Information related to the HP CDA CLI can be found in Appendix A, [The HP CDA Command Line Interface](#).

To create infrastructure templates using the CLI, complete the following steps:

1. Identify the existing compute resources that you wish to add to the infrastructure template, and note their hostnames and IP addresses.
2. Use the `existinginfrastructure add server` command to create logical objects in HP CDA to represent each of the compute resources.
3. Identify the server logical objects that will be grouped into logical server pools.

Note: Server pools are groupings of compute resources having similar properties, such as the same OS and memory size. HP highly recommends that you group compute resources by operating system as OS specific.

4. Use the `existinginfrastructure add serverpool` command to create logical objects in HP CDA to represent each server pool, and to add compute resource objects to the server pool objects.

Note: The `-attachserver` option of the `existinginfrastructure add serverpool` command can be used to add compute resource objects to a server pool object as it is created, or they

can be added to the server pool object subsequently using the `existinginfrastructure set serverpool` command.

5. Use the `existinginfrastructure add template` command to create a logical object in HP CDA to represent the infrastructure template being created, and to add server pool objects to the infrastructure template object.

Note: The `-attachserverpool` option of the `existinginfrastructure add template` command can be used to add server pool objects to the infrastructure template object as it is created, or they can be added to the infrastructure template object subsequently using the `existinginfrastructure set template` command.

After you add the templates to HP CDA, they become visible in the “Infrastructure Templates” screen when the **Existing Infrastructure Template Provider** option is chosen in the “Select Template Provider” drop-down field. The infrastructure templates can then be used as the basis for platform models.

Creating a Pet Clinic Infrastructure Template Using the Existing Infrastructure CLI Example

This example explains how to create an infrastructure template for an application called Pet Clinic. The Pet Clinic example calls for an infrastructure template that contains two server groups named App and DB, each having a single provisioned server named `app511.syslab.adam.qa.hp.com` and `db511.syslab.adam.qa.hp.com` respectively. There are many ways to achieve the same Existing Infrastructure environment. Since the `add` operation was explained in detail above, this example will use the HP CDA CLI `set` command in the example below. For more information related to the HP CDA CLI, see Appendix A, [The HP CDA Command Line Interface](#).

To create the example infrastructure template, complete the following steps:

1. To create the logical template, logical server pools, and logical servers for the Pet Clinic example, run the following commands:
 - `cdaexec existinginfrastructure add template --name PetClinicTemplate`
 - `cdaexec existinginfrastructure add serverpool --name App --ostype Linux`
 - `cdaexec existinginfrastructure add serverpool --name DB --ostype Linux`
 - `cdaexec existinginfrastructure add server --hostname app511.syslab.adam.qa.hp.com --ip 172.16.253.60`
 - `cdaexec existinginfrastructure add server --hostname db511.syslab.adam.qa.hp.com --ip 172.16.253.61`
2. To associate the two existing server pools with the existing template, run the following command:
 - `cdaexec existinginfrastructure set template PetClinicTemplate --attachserverpool App --attachserverpool DB`
3. To associate the two existing servers with the existing server pools, run the following commands:
 - `cdaexec existinginfrastructure set serverpool App --attachserver app511.syslab.adam.qa.hp.com`
 - `cdaexec existinginfrastructure set serverpool DB --attachserver db511.syslab.adam.qa.hp.com`
4. After the `PetClinicTemplate` has been used to provision an infrastructure platform, its compute resources will be exhausted. To add support for two more platforms, add two more existing servers to each server pool by running the following commands:

- `cdaexec existinginfrastructure add server --hostname app512.syslab.adam.qa.hp.com --ip 172.16.253.62`
- `cdaexec existinginfrastructure add server --hostname app513.syslab.adam.qa.hp.com --ip 172.16.253.64`
- `cdaexec existinginfrastructure set serverpool App --attachserver app512.syslab.adam.qa.hp.com --attachserver app513.syslab.adam.qa.hp.com`
- `cdaexec existinginfrastructure add server --hostname db512.syslab.adam.qa.hp.com --ip 172.16.253.63`
- `cdaexec existinginfrastructure add server --hostname db513.syslab.adam.qa.hp.com --ip 172.16.253.65`
- `cdaexec existinginfrastructure set serverpool DB --attachserver db512.syslab.adam.qa.hp.com --attachserver db513.syslab.adam.qa.hp.com`

Note: Deprovision infrastructure platforms that are no longer in use in order to release their compute resources and make the compute resources available to future infrastructure platforms.

Platform Software Management

Infrastructure templates specify much of the characteristics that need to be included in a platform model. At times, however, a template does not specify all of the software capabilities that will be needed on the infrastructure being modeled, in which case, additional software, known as *add-on software*, will need to be modeled and added to the platform model. This section explains how to create and manage platform software.

Understanding Platform Software Models (Overview)

A platform software model fundamentally consists of the following parts:

- **Requirements and Capabilities** – *Requirements* identify what the platform software requires of the platform onto which it is to be deployed in order for it to function successfully. Such requirements might include the operating system, minimum amount of memory, other software, etc., that must be present on the platform. *Capabilities* identify the additional capabilities that the platform software brings to the platform onto which it is installed. Requirements and capabilities are added to the platform software using HP CDA's tagging feature, and the tags can be referenced by the designers during platform and application modeling.
- **Workflows** – *Workflows* contain sequences of programming operations that perform particular actions when run (see [Workflows and Programming Actions](#) on page 80).
- **Parameter Definitions Groups** - Parameters that can be referenced by the "Input Component Parameters" that are defined within the programming operations that comprise workflows are created under *Parameter Definitions Groups*. Such parameters provide platform software-level reference points for values that need to be used within more than one programming operation, eliminating the need to enter the same value locally in various programming operations.

Parameters that are defined in the parameter definitions group in turn can reference the following other types of parameters:

- Server Parameters
- Parameters associated with the platform software as defined in the assigned capability tags
- Parameters associated with software on which the platform software has a dependency as defined in the assigned requirement tags.

Workflows and Programming Actions

Workflows can either be user-defined, or four pre-defined workflows can be used. Following are your workflow definition choices:

- User-defined workflow – *User-defined workflows* are named by the platform software designer and are typically included to perform actions on the platform software that are not covered in one of the pre-defined workflows, such as a software “pause” operation.
- “Deploy” workflow – The *Deploy workflow* is included with HP CDA specifically to contain the sequence of programming operations that are executed to deploy the platform software onto the provisioned platform during a platform provisioning operation. The Deploy workflow runs automatically during infrastructure provisioning, after the infrastructure that the software is to be deployed on has been successfully provisioned and is in a functioning state.
- “Undeploy” workflow – The *Undeploy workflow* is included with HP CDA specifically to contain the sequence of programming operations to be executed to undeploy the platform software from the provisioned platform during an infrastructure deprovisioning operation.

Note: An option that is set when the deployer parameters are configured in the platform designer screen determines whether or not the Undeploy workflow will run. See [Specifying Deployer Parameters](#) on page 102 for more information.

- “Start” workflow – The *Start workflow* is included with HP CDA specifically to contain the sequence of programming operations that are executed to start the platform software.
- “Stop” workflow – The *Stop workflow* is included with HP CDA specifically to contain the sequence of programming operations that are executed to stop the platform software during an infrastructure deprovisioning operation.

Note: Workflows that do not contain any operations are ignored during platform provisioning and deprovisioning operations.

As noted earlier, a workflow contains a sequence of one or more *programming operations* that run sequentially in order to accomplish some objective related to the platform software being modeled. The following four sections explain the types of programming operations that are available in HP CDA.

“Placed File” operation

A Placed File operation copies a file from a specified location to a specified directory on the platform server to which the external software is to be installed (referred to as the *target platform server*), and specifies the permissions of that file on the platform server.

The following example graphic shows HP CDA window in which you design the Placed File operation.

The screenshot shows the configuration interface for a 'Placed File' operation. At the top right, there are buttons for 'Save', 'Cancel', 'Clone', and 'Delete'. The main interface is divided into several sections:

- General Information:**
 - Name:** setup sh
 - Source:** Use image from DSL (dropdown), DiagProbeSetup.sh (file icon), Upload (upload icon), Change (change icon), Delete (delete icon)
 - Destination:** /tmp/HPDiagnostics/setup.sh
 - Content Parameterized:**
- Input Component Parameters:**
 - + Add
- Details:**
 - Group:** root
 - Mode:** 777
 - Owner:** root
 - Replace:**
 - Content Encoding:** (empty field)
- Advanced Options:**
 - Ignore Failure
 - Retry on Failure
 - Retry Count:** (input field)
 - Time Between Retries:** (input field) Seconds (dropdown menu)

Table 4 explains the various settings to be made in that screen, with instructions on how to set them where necessary.

Table 4 Placed File Screen Parameter Settings

Parameter Name	Setting Instructions
Name	Enter the name to be used to identify the Placed File operation.
Source	<p>Choose one of the following options in the drop-down field, and perform the instructions provided:</p> <ul style="list-style-type: none"> • Download from external URL – If this option is chosen, enter the URL that references the file to be downloaded into the adjacent text entry field, for example - <code>ftp://1.2.3.4/HP/Diagnostics/Diagnostics-installer/instFile.zip</code>. • Use image from DSL – If this option is chosen, do the following: <ul style="list-style-type: none"> ○ Click the Upload icon to invoke the “Attach Data Content” dialog box. ○ Select the appropriate “Source” radio button. ○ Enter either the URL or file system path to the file and the file name in the “File” field. ○ In the “Server Folder” field, enter the location on the HP CDA server where the uploaded file will be placed. The default is <code>/defaultDomain</code>. ○ Click the Upload button.
Destination	Enter the path and the name of the file as it will be placed in the HP CDA server file system, for example - <code>/tmp/Diagnostics/installer/instFile.zip</code> .
Content Parameterized	Select this checkbox if the content of the placed file is to be checked for parameters and substitution variables. Any parameters and substitution variables that are found will be replaced with their resolved values. If this checkbox is not selected, the file will be placed as-is with no attempt to perform substitution on the content of the file.
Input Component Parameters	
Add	<ul style="list-style-type: none"> • Click the Add icon to invoke the “Add” dialog box. • Enter the label for the parameter into the “Label” field. <ul style="list-style-type: none"> Note: When the “Label” field is initially filled out, the same characters are automatically entered into the “Name” field, with spaces replaced by periods. • Modify the name for the parameter in the “Name” field, if necessary. • Enter a Description for the parameter into the “Description” field. • Enter the textual default value of the parameter into the “Default Value” field, leave the field blank to indicate no default value, or click the ellipsis icon (...) and choose an existing variable to be referenced by this parameter. <ul style="list-style-type: none"> Note: Input component parameters can reference the following: <ul style="list-style-type: none"> • Server parameters • Platform software-level parameters • Parameters associated with the platform software as defined in the assigned capability tags • Parameters associated with software on which the platform software has a dependency as defined in the assigned requirement tags. <p>If the parameter to be referenced has not been created yet when you create an input component parameter, you will need to edit the input component parameter to include the reference once the parameter to be referenced has been created.</p> • Select the “Required” checkbox if a value for the parameter is required. • Select the “Encrypted” checkbox to mask the parameter value in screen displays. • Select the “Read Only” checkbox if the parameter’s value is to be read only (not writable). • Select the “Private Scope” checkbox to limit the scope of the parameter to within the platform software. Parameters with a private scope will not be listed when, for example, the platform is provisioned.

Parameter Name	Setting Instructions
Remove	<ul style="list-style-type: none"> • Select the checkbox that precedes the parameters in the list that you want to delete. • Click the Remove icon. • Click the Remove button in the warning message box that appears.
Save as CSV	<p>Note: This function saves the entire list of parameters specified in the Placed File screen to a MS Excel comma-separated value (.csv) file.</p> <ul style="list-style-type: none"> • Click the Save as CSV icon, and click the Save button in the dialog box that appears. • Choose the path and name for the comma-separated value file, and then click Save in the “Save As” dialog box that appears. • Choose the desired action in the “Download complete” dialog box that appears.
Details	
Group	Enter the group to which the placed file is to belong.
Mode	Enter the user, group, and “other” permissions to be applied to the placed file. The input format is the same as traditional UNIX permission format using three character octal notation.
Owner	Enter the name of the owner of the placed file.
Replace	Select this checkbox if the file to be placed should overwrite a file with the same name at the placed file destination, if it exists. If unselected, an existing file will not be replaced.
Content Encoding	Enter the character encoding of the content. This parameter can be used to prevent HP CDA from corrupting the placed file content during parameter replacement. For example, if the placed file content is parameterized (the Content Parameterized checkbox is selected), the file is parsed, parameter tokens are replaced, and the file is rewritten, the encoding specified here will be used for reading and re-writing the file for the particular deployment. The default character encoding is UTF-8.
Advanced Options	
Ignore Failure	Select this checkbox to continue the deployment should the operation fail to deploy correctly. If this checkbox is not selected and a failure occurs, the deployment will be left in its failed state, and the user can choose to start the Backout operation from the job report at a later time.
Retry on Failure	<p>Select this checkbox if you want the operation to retry if it fails, and then do the following:</p> <ul style="list-style-type: none"> • In the “Retry Count” field, enter the number of times that the operation is to be retried. • In the “Time Between Retries” field, enter the amount of time between retries into the text entry field, and select the units of time from the drop-down field. <p>If the Retry on Failure checkbox is selected and all retry attempts fail, subsequent deployment behavior depends on whether the user has selected the Force Automatic Backout and cleanup on any failure checkbox in the “Provision Platform” dialog box. If the checkbox is selected, then the Backout operation should begin automatically when all retries have failed.</p>
<p>¹ The “Details” section is used to configure specific permissions on the file being placed on the target server. The behavior of HP CDA regarding these settings is as follows:</p> <ul style="list-style-type: none"> • If the deployer is Opscode Chef and the target platform is Microsoft Windows, the settings in the “Details” section do not apply and should not be used. • If the deployer is Opscode Chef and the target platform is Linux, the settings in the “Details” section can be used for setting Linux style permissions. • If the deployer is HP Server Automation and the target platform is either Microsoft Windows or Linux, the settings in the “Details” section can be used for setting permissions and the values will be used as supported by the target platform. 	

“Executed Script” operation

An Executed Script operation executes a script that is included as part of the programming operation.

Note: Application models are only as intelligent as the individual steps that are included in them, and following some scripting standards will assist in avoiding issues and debugging if required. The following best practices apply to scripts used in an “Executed Script” operation:

- Use standard scripting practices and include pre- and post-execution checks to debug the execution if required.
- Include code to create a log that contains details on the operation being done via the script.

Action
Backout

Name: *

Path:

Precondition:

User:

Working Directory:

Content

```
#!/bin/sh
mkdir -p /opt/HPDiagnostics
unzip /tmp/HPDiagnostics/installer/ProbeInstall.zip -d /opt/HPDiagnostics
cd /opt/HPDiagnostics/etc
sed -e 's/Id=@(diagnostics.application.name)_hostname~/g' -e 's/Group~/g' probe.properties > /tmp/temp
mv /tmp/temp probe.properties
sed -e 's/Url=http://(diagnostics.server.host):@(diagnostics.server.port)/Registrar~/g' dispatcher.properties > /tmp/temp
mv /tmp/temp dispatcher.properties
sed -e 's/Tomcat.initialization.delay.seconds~/Tomcat.initialization.delay.seconds=60~/g' metrics.config > /tmp/temp
mv /tmp/temp metrics.config
mv /tmp/HPDiagnostics/setup.sh /usr/share/tomcat6/bin/
```

Input Component Parameters

+ Add X Remove Save as CSV

Name	Label	Default Value	Required	Encrypted	Read Only	Private Scope	
<input type="checkbox"/>	diagnostics.applicat...	PetClinic	Yes	No	No	No	<input type="text"/>
<input type="checkbox"/>	diagnostics.server.h...	adam-r215.ftc.ada...	No	No	No	No	<input type="text"/>
<input type="checkbox"/>	diagnostics.server.p...	2006	Yes	No	No	No	<input type="text"/>

Environment Variables

Add

Advanced Options

Ignore Failure

Retry on Failure

Retry Count:

Time Between Retries: Minutes

Table 5 explains the various settings to be made in that screen, with instructions on how to set them where necessary.

Table 5 Executed Script Screen Parameter Settings

Parameter Name	Setting Instructions
Name	Enter the name to be used to identify the Executed Script operation.
Path	Enter the path on the target system to use when searching for commands in scripts.
Precondition	Enter a command that will test that some important condition is met. For example, if you need to ensure that some file exists before executing the script, you can use <code>test -f /tmp/myfile</code> on Unix as the precondition to make sure that <code>/tmp/myfile</code> exists before the script executes.
User	Enter the name of the user to run the script as on the target system.
Working Directory	Enter the current working directory to run the command from on the target system.
Content	
N/A	<p>Enter the script to be executed into the text entry field in the “Content” portion of the Executed Script screen. The script can be typed or pasted into the text entry field.</p> <p>Note: HP CDA assumes that the content block entered into this field contains content that can be run as a shell script on Unix targets or a Powershell script on Windows targets. This is common to both Chef and SA deployers.</p>
Input Component Parameters	
Add	<ul style="list-style-type: none"> Click the Add icon to invoke the “Add” dialog box. Enter the label for the parameter into the “Label” field. <p>Note: When the “Label” field is initially filled out, the same characters are automatically entered into the “Name” field, with spaces replaced by periods.</p> Modify the name for the parameter in the “Name” field, if necessary. Enter a Description for the parameter into the “Description” field. Enter the textual default value of the parameter into the “Default Value” field, leave the field blank to indicate no default value, or click the ellipsis icon (...) and choose an existing variable to be referenced by this parameter. <p>Note: Input component parameters can reference the following:</p> <ul style="list-style-type: none"> Server parameters Platform software-level parameters Parameters associated with the platform software as defined in the assigned capability tags Parameters associated with software on which the platform software has a dependency as defined in the assigned requirement tags. <p>If the parameter to be referenced has not been created yet when you create an input component parameter, you will need to edit the input component parameter to include the reference once the parameter to be referenced has been created.</p> Select the “Required” checkbox if a value for the parameter is required. Select the “Encrypted” checkbox to mask the parameter value in screen displays. Select the “Read Only” checkbox if the parameter’s value is to be read only (not writable). Select the “Private Scope” checkbox to limit the scope of the parameter to within the platform software. Parameters with a private scope will not be listed when, for example, the platform is provisioned.
Remove	<ul style="list-style-type: none"> Select the checkbox that precedes the parameters in the list that you want to delete. Click the Remove icon. Click the Remove button in the warning message box that appears.

Parameter Name	Setting Instructions
Save as CSV	<p>Note: This function saves the entire list of parameters specified in the Executed Script screen to a MS Excel comma-separated value (.csv) file.</p> <ul style="list-style-type: none"> Click the Save as CSV icon, and click the Save button in the dialog box that appears. Choose the path and name for the comma-separated value file, and then click Save in the "Save As" dialog box that appears. Choose the desired action in the "Download complete" dialog box that appears.
Environment Variables	
Add	Click Add , and then enter the name of the environment variable in the leftmost entry field, and enter the value of the environment variable in the rightmost entry field.
Advanced Options	
Ignore Failure	Select this checkbox to continue the deployment should the operation fail to deploy correctly. If this checkbox is not selected and a failure occurs, the deployment will be left in its failed state, and the user can choose to start the Backout operation manually from the job report at a later time.
Retry on Failure	<p>Select this checkbox if you want the operation to retry if it fails, and then do the following:</p> <ul style="list-style-type: none"> In the "Retry Count" field, enter the number of times that the operation is to be retried. In the "Time Between Retries" field, enter the amount of time between retries into the text entry field, and select the units of time from the drop-down field. <p>If the Retry on Failure checkbox is selected and all retry attempts fail, subsequent deployment behavior depends on whether the user has selected the Force Automatic Backout and cleanup on any failure checkbox in the "Provision Platform" dialog box. If the checkbox is selected, then the Backout operation should begin automatically when all retries have failed.</p>

"Service Command" operation

A Service Command operation performs a specified action on a local service on the target platform server, and specifies the behavior of that service when the target platform server reboots.

The screenshot shows a configuration window for a Service Command operation. At the top right are buttons for 'Save', 'Cancel', 'Clone', and 'Delete'. Below these are two tabs: 'Action' (selected) and 'Backout'. The main configuration area includes the following fields:

- Name:** * Pet Clinic Service
- Service Name:** * petClinic.exe
- Action:** Restart (dropdown menu)
- Startup Type:** Enable at Startup (dropdown menu)
- Path:** \Program Files\Pet Clinic\2012\PetClinic Service\
- Pattern:** (empty text field)

Below the main fields is a section for 'Input Component Parameters' with an '+ Add' button. Underneath is the 'Advanced Options' section, which contains:

- Ignore Failure
- Retry on Failure
 - Retry Count: 5
 - Time Between Retries: 90 Seconds (dropdown menu)

Table 6 Service Command Screen Parameter Settings explains the settings to be made in that screen, with instructions on how to set them where necessary.

Table 6 Service Command Screen Parameter Settings

Parameter Name	Setting Instructions
Name	Enter the name to be used to identify the Service Command operation.
Service Name	Enter the name of the local service to be acted upon by the Service Command operation.
Action	Choose one of the following options in the drop-down field: <ul style="list-style-type: none"> • Restart – Starts the local service when the Service Command operation runs if it stopped. If the service is already started, this action restarts it. • Start – Starts the local service if it is not running when the Service Command operation runs. • Stop – Stops the local service if it is running when the Service Command operation runs.
Startup Type	Choose one of the following options in the drop-down field: <ul style="list-style-type: none"> • Disable at Startup – Disables (does not automatically start) the local service when the target platform server reboots. • Don't change – Does not change the startup type from its current setting. • Enable at Startup – Enables (automatically starts) the local service, if it was disabled, when the target platform server reboots.
Path	Enter the path on the target platform server to where the local service executable is located.
Pattern	Enter the pattern to search for in the process table, for instance on Unix systems where the service script has a different name than the name of the process in the process table.
Input Component Parameters	
Add	<ul style="list-style-type: none"> • Click the Add icon to invoke the “Add” dialog box. • Enter the label for the parameter into the “Label” field. Note: When the “Label” field is initially filled out, the same characters are automatically entered into the “Name” field, with spaces replaced by periods. • Modify the name for the parameter in the “Name” field, if necessary. • Enter a Description for the parameter into the “Description” field. • Enter the textual default value of the parameter into the “Default Value” field, leave the field blank to indicate no default value, or click the ellipsis icon (...) and choose an existing variable to be referenced by this parameter. Note: Input component parameters can reference the following: <ul style="list-style-type: none"> • Server parameters • Platform software-level parameters • Parameters associated with the platform software as defined in the assigned capability tags • Parameters associated with software on which the platform software has a dependency as defined in the assigned requirement tags. If the parameter to be referenced has not been created yet when you create an input component parameter, you will need to edit the input component parameter to include the reference once the parameter to be referenced has been created. • Select the “Required” checkbox if a value for the parameter is required. • Select the “Encrypted” checkbox to mask the parameter value in screen displays. • Select the “Read Only” checkbox if the parameter’s value is to be read only (not writable). • Select the “Private Scope” checkbox to limit the scope of the parameter to within the platform software. Parameters with a private scope will not be listed when, for example, the platform is provisioned.
Remove	<ul style="list-style-type: none"> • Select the checkbox that precedes the parameters in the list that you want to delete. • Click the Remove icon. • Click the Remove button in the warning message box that appears.

Parameter Name	Setting Instructions
Save as CSV	<p>Note: This function saves the entire list of parameters specified in the Service Command screen to a MS Excel comma-separated value (.csv) file.</p> <ul style="list-style-type: none"> • Click the Save as CSV icon, and click the Save button in the dialog box that appears. • Choose the path and name for the comma-separated value file, and then click Save in the “Save As” dialog box that appears. • Choose the desired action in the “Download complete” dialog box that appears.
Advanced Options	
Ignore Failure	<p>Select this checkbox to continue the deployment should the operation fail to deploy correctly. If this checkbox is not selected and a failure occurs, the deployment will be left in its failed state, and the user can choose to start the Backout operation from the job report at a later time.</p>
Retry on Failure	<p>Select this checkbox if you want the operation to retry if it fails, and then do the following:</p> <ul style="list-style-type: none"> • In the “Retry Count” field, enter the number of times that the operation is to be retried. • In the “Time Between Retries” field, enter the amount of time between retries into the text entry field, and select the units of time from the drop-down field. <p>If the Retry on Failure checkbox is selected and all retry attempts fail, subsequent deployment behavior depends on whether the user has selected the Force Automatic Backout and cleanup on any failure checkbox in the “Provision Platform” dialog box. If the checkbox is selected, then the Backout operation should begin automatically when all retries have failed.</p>

“External Component” operation

An External Component operation can be used to execute an automation sequence from an integrated HP CDA service provider as part of a HP CDA workflow. The following automation sequences are supported:

- HP Operations Orchestration flows
- HP Server Automation packages and policies
- HP Database and Middleware Automation flows
- Chef recipes

Operation Backout

Name: *

Specify external component

Plugin: ▼

Plugin configuration: ▼

Path: Opsware/Tools/Database & Middleware Automation/DMA Agent Support

External name: DMA Agent Support

Component parameters

Execution Action: * ▼

Component output parameters

Advanced Options

Ignore Failure

Retry on Failure

Retry Count:

Time Between Retries: Minutes

Table 7 External Component Screen Parameter Settings explains the settings to be made in that screen, with instructions on how to set them where necessary. .

Table 7 External Component Screen Parameter Settings

Parameter Name	Setting Instructions
Name	Enter the name to be used to identify the External Component operation.
Description	Enter a description for the External Component operation.
Specify external component	
Plugin	Choose the type of external component you wish to reference in this drop-down field. The following component types are available: <ul style="list-style-type: none"> • OO Flow • SA Package • SA Policy • DMA Flow • Chef Recipe
Plugin Configuration	If more than one provider of the component type chosen in the "Plugin" field has been integrated with HP CDA, choose the name of the plugin configuration associated with the desired provider in this drop-down field.
Path	The path and file name of the component selected in the "External Name" field.
External Name	Click the Browse button to invoke the "Browse external component" dialog box, navigate to and select the external component that you want to associate with the External Component operation, and then click OK.
Component parameters	
parameter names	The parameter names listed in the left column are taken from the external component and cannot be modified here. Enter a value for each parameter into the text entry fields that are adjacent to each parameter name.
Component output parameters	
parameter names	The output parameter names listed in the left column are taken from the external component and cannot be modified here. Enter a value for each output parameter into the text entry fields that are adjacent to each output parameter name.
Advanced Options	
Ignore Failure	Select this checkbox to continue the deployment should the operation fail to deploy correctly. If this checkbox is not selected and a failure occurs, the deployment will be left in its failed state, and the user can choose to start the Backout operation from the job report at a later time.
Retry on Failure	Select this checkbox if you want the operation to retry if it fails, and then do the following: <ul style="list-style-type: none"> • In the "Retry Count" field, enter the number of times that the operation is to be retried. • In the "Time Between Retries" field, enter the amount of time between retries into the text entry field, and select the units of time from the drop-down field. If the Retry on Failure checkbox is selected and all retry attempts fail, subsequent deployment behavior depends on whether the user has selected the Force Automatic Backout and cleanup on any failure checkbox in the "Provision Platform" dialog box. If the checkbox is selected, then the Backout operation should begin automatically when all retries have failed.

Displaying Platform Software

To display platform software, complete the following steps:

1. Invoke the “Software” screen by clicking the [Browse Software](#) link that is on the Platforms tab in the HP CDA interface.

The “Software” screen contains the following three tabs:

- All – A listing of all platform software that exists in HP CDA’s platform software library appears on this tab. This tab is selected by default when the “Software” screen is invoked.
- My – A listing of all platform software in the platform software library for which the user currently logged in is the owner.
- Favorites - A listing of the platform software that have been marked as favorites.

2. Select the appropriate tab, if necessary.

The list of platform software is presented in the screen in tabular format. The following information is presented for each platform software in the list:

- Name – The name of the platform software.
- Owner – The current owner of the platform software.
- Domain – The domain to which the platform software belongs.

Creating Platform Software

This procedure explains how to create platform software in the HP CDA interface.

Note: You cannot use HP CDA to add SA agent software to a platform as add-on software; therefore, you should not create platform software for a SA agent. Two alternative methods of including the SA agent software are to include it in the infrastructure template for any VMs that might use HP SA as the deployer, or to add the SA agent software to the provisioned platform manually and register the provisioned platform with the SA server.

Create the Platform Software Object

To create a platform software object, complete the following steps:

1. Invoke the “Software” screen by clicking the [Create Software](#) link that is on the Platforms tab in the HP CDA interface.
2. Enter a name that will be used to identify the platform software being created in the “Name” field.
3. Enter a Description for the platform software being created, if desired, into the “Description” field.
4. Under the “Requirements” header, click the [Change link](#), if desired, to invoke the “Change Layer Requirements” dialog box.

Note: For information about platform software requirements, see “Requirements and Capabilities” in [Understanding Platform Software Models \(Overview\)](#) on page 79.

5. In the “Change Layer Requirements” dialog box, click the tab for the requirement that you want to tag.

For example, if the platform software being created requires a platform running Linux in order to run, select the **Operating System** tab.

6. Click **Add Requirement** to invoke the pertinent “New Requirement” dialog box.
7. Select the pertinent requirement.
For example, if the platform software being created requires a 64 bit version of Linux 2.6 or newer in order to run, do the following:
 - Select **Exact and Newer Versions** in the “Operator” drop-down field.
 - Drill down to and select **Linux 2.6** in the “Requirement” section of the dialog box (Unix ▶ Linux ▶ Linux2.6).
 - Select **64bit** in the “Architecture” drop-down field.
8. Click **OK** to close the “New Requirement” dialog box.
9. Repeat steps 5 through 8 as is necessary to add other necessary platform software requirements.
10. Click **Save Changes** to save the requirements and close the “Change Layer Requirements” dialog box.
11. Click **Save** to save the platform software.
A “Software” screen that is specific to the created platform software displays.

Add Platform Software Requirements and Capabilities

Note: You specify platform software requirements and capabilities by tagging the software with requirement and capability tags. HP CDA contains several default tags from which you can select when you perform this procedure. If you do not see a desired tag, the HP CDA administrator can add it for you.

To add, complete the following steps:

1. On the **Overview** tab in the screen under the “Requirements” header, click the **Change** link, if desired, and make any desired changes to the platform software requirements.

The steps necessary to change requirements are very similar to steps 5 through 10 above under [Creating Platform Software](#) on page 91.

2. On the **Overview** tab in the screen under the “Capabilities” header, click the **Change** link, if desired, and add any desired platform software capabilities.

The steps necessary to add or change capabilities are very similar to steps 5 through 10 above under [Creating Platform Software](#) on page 91.

Note: For information about platform software requirements, see “Requirements and Capabilities” in [Understanding Platform Software Models \(Overview\)](#) on page 79.

Add Workflows

To add workflows, complete the following steps:

1. Click the **Workflows** tab.

On the **Workflows** tab, you will be creating the workflows that will be needed at various times during the existence of the platform software. For example, all of the programming operations that are necessary to deploy the platform software to the platform will be added as part of the “Deploy” workflow. For information about workflow types, see [Workflows and Programming Actions](#) on page 80.

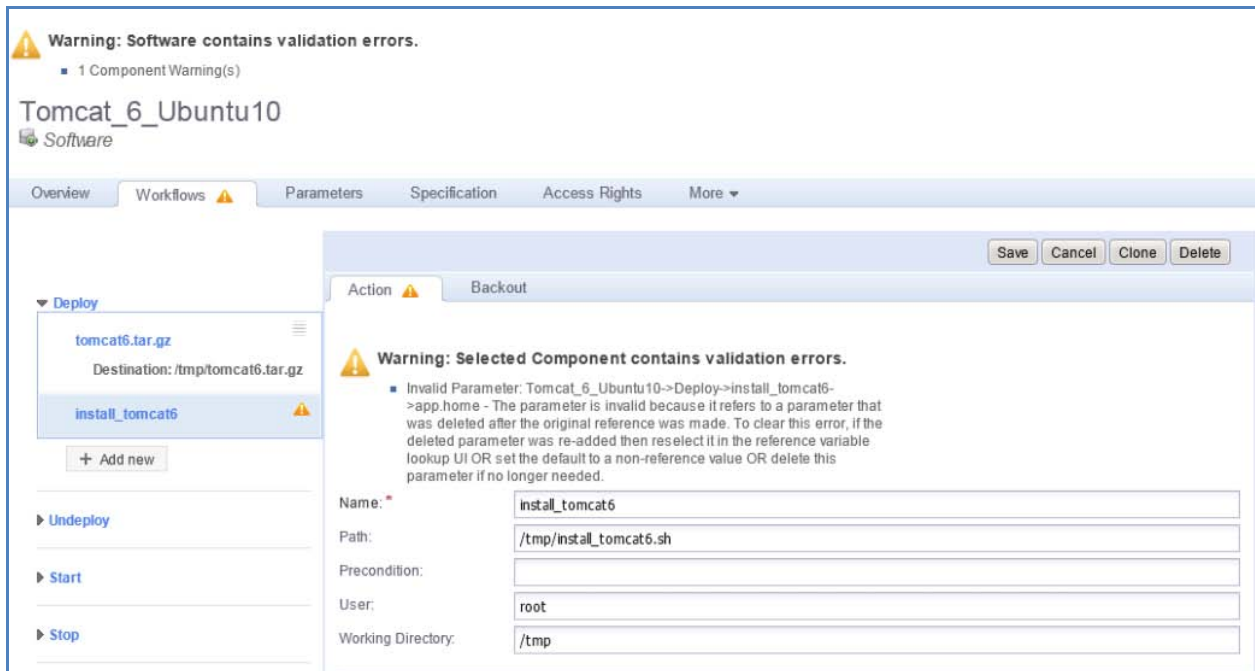
2. Click on the workflow type (for example, **Deploy**) under which you want to create a workflow.

Add New appears under the workflow type name when you click on it.

3. Hover the mouse pointer over **Add New** to bring up a menu of programming operation types, and select the programming operation type that you want to create.
4. Fill in the fields under the **Operation** tab in the programming operation screen as is necessary.

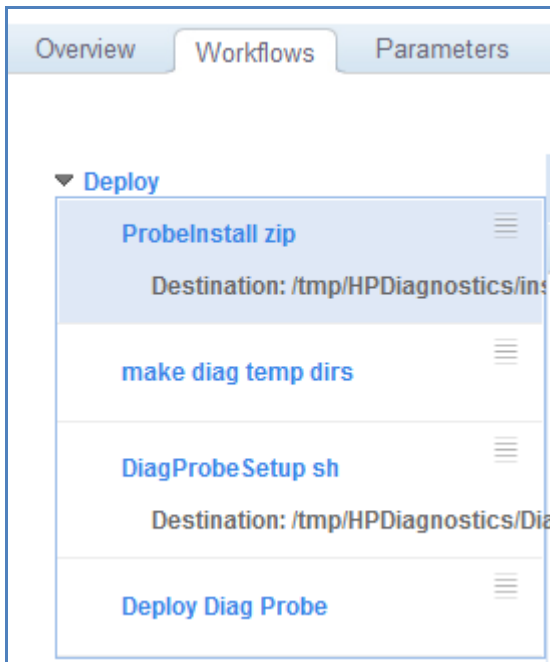
Refer to the tables within [Workflows and Programming Actions](#) on page 80 when filling in the fields.

Note: Parameter references from Input Component Parameters in a workflow operation are not validated until you click the **Save** button to save the operation. If a parameter validation error occurs, a warning displays. The text of the warning contains information to help resolve the error condition.

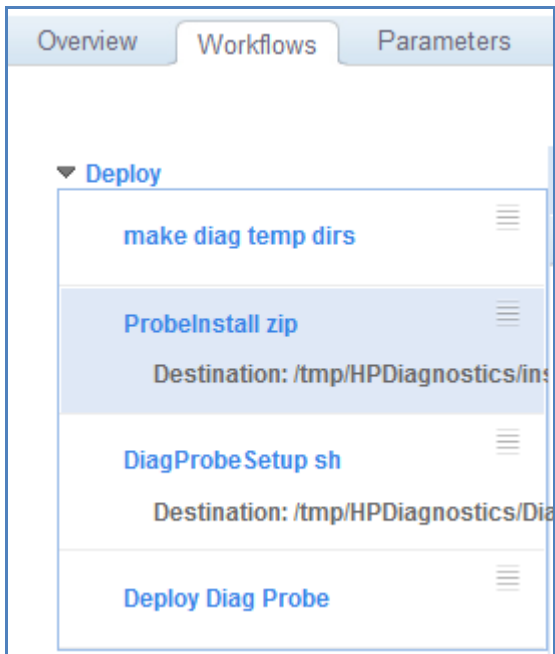


5. If desired, click the **Backout** tab, and select and create an operation that will be executed should the programming operation fail to execute correctly.
6. Click the **Save** button that is on the upper right side of the screen to save the programming operation.
7. Repeat steps 3 through 6 to create all of the programming operations that constitute the workflow.

Note: When a workflow runs during platform provisioning or deprovisioning, programming operations execute in the order in which they appear under workflow type header in the screen. In the following graphic example, the programming operations under the **Deploy** workflow header will execute in the order shown.



Programming operations within a given workflow can be rearranged by dragging and dropping them to create the desired order. In the following example graphic, if you grab and drag **ProbelInstall.** to the area between **make diag temp dirs** and **DiagProbeSetup sh.** Operations will have the order shown in the following example graphic.



- Repeat steps 2 through 7 to create all of the workflows necessary for the platform software.

Adding Parameters Definitions Groups

Parameters created under Parameter Definitions Groups can be referenced by the “Input Component Parameters” that have been included in the programming operations that were added in the Add Workflows (see [Add Workflows](#) on page 92) procedure.

Such parameters created in this procedure provide reference points at the platform software-level for values that need to be used within more than one programming operation in a workflow, eliminating the need to enter the same value locally in various programming operations.

To add a parameters definition group, complete the following steps:

1. Click the **Parameters** tab.
2. Click the **+ Add** icon to invoke the “Add” dialog box.
3. Enter the label for the parameter into the “Label” field.

Note: When the “Label” field is initially filled out, the same characters are automatically entered into the “Name” field, with spaces replaced by periods.

4. Modify the name for the parameter in the “Name” field, if necessary.
5. Enter a description for the parameter into the “Description” field.
6. Enter the textual default value of the parameter into the “Default Value” field, leave the field blank to indicate no default value, or click the ellipsis icon (...) and choose an existing variable to be referenced by this parameter.

Note: Platform software-level parameters can reference the following:

- Server parameters
 - Parameters associated with the platform software as defined in the assigned capability tags
 - Parameters associated with software on which the platform software has a dependency as defined in the assigned requirement tags.
7. Select the “Required” checkbox if a value for the parameter is required.
 8. Select the “Encrypted” checkbox to mask the parameter value in screen displays.
 9. Select the “Read Only” checkbox if the parameter’s value is to be read only (not writable).
 10. Select the “Private Scope” checkbox to limit the scope of the parameter to within the platform software. Parameters with a private scope will not be listed when, for example, the platform is provisioned.
 11. Click the **Add** button to add the platform software-level parameter to the Parameter Definitions Group.
 12. Repeat steps 2 through 11 as is necessary to add any remaining platform software-level parameters to the Parameter Definitions Group.

Create References from Input Component Parameters to Platform Software-Level Parameters

During the Add Workflows part of Creating Platform Software, you might have created input component parameters within the programming operations of the workflows that were to be referenced to platform software-level parameters. If you have followed Creating Platform Software from the start up until this point, those input component parameters were created before platform software-level parameters were created during the Adding Parameters Definitions Groups part the procedure.

Now that software-level parameters have been created, edit the necessary input component parameters to create the references. Referencing parameters has already been explained in this chapter.

Platform Model Management

A *platform model* is a logical representation of the combination of infrastructure, software, and parameter settings that, when provisioned, results in a realized platform on which an application can be deployed. It includes the necessary configurations and elements that will be provisioned independently of the application when a given application is deployed.

In most cases, a platform model is based on an infrastructure template which includes the software, operating systems, and configuration required by the application, and additional software and capabilities not included in the template are added during the platform modeling process.

This section explains how to create and manage platform models.

Creating a Platform Model from a Template

To create a platform model from a template, complete the following steps:

1. Invoke the "Platform" screen by clicking the **Create Platform** link that is on the **Platforms** tab in the HP CDA interface.
2. Enter a name that will be used to identify the logical platform being created in the "Name" field.
3. Enter a Description for the logical platform being created, if desired, into the "Description" field.
4. Under "Stages," select the checkboxes that represent the stages in the software development cycle to which the logical platform being created belongs.
5. Select the deployer that will be used to deploy software onto the provisioned platform using the "Deployer" drop-down field.
6. On the left side of the screen under "Infrastructure Template," select the name of the infrastructure template that is to serve as the foundation for the platform to be deployed.

The core hardware, OS, and software capabilities of the selected infrastructure template are shown on the right side of the screen.

7. Click the **Save** button to save the logical platform model.

The platform model is saved. A platform model-specific screen displays.

8. (*Optional*) You can further define the logical platform model in that screen.

Pet Clinic Logical Platform

Platform

Overview Designer Connections Inventory Stored Parameters More ▾

Infrastructure Template: [Pet-Clinic_Template](#)

Domain: [Default](#)

Owner: [Administrator](#)

Valid: [Yes](#)

Server Groups

App

Hardware: 1 CPU, 512 MB RAM

Operating System: Linux

Software: Tomcat 6

DB

Hardware: 1 CPU, 512 MB RAM

Operating System: Linux

Software: MySQL 5.1

Latest Events

Provisioning failed New Administrator, Mar 30

[More ...](#)

Provision

Add to Favorites

Edit

Delete




Export

Change Domain

Last Edit

[Administrator, Mar 30](#)

A platform model-specific screen contains the following tabbed section:

- **Overview:** This tab contains some general information pertaining to the platform model, such as the infrastructure template on which the model is based, domain and user ownership information, the currently specified platform capabilities, and important events related to the platform model.
- **Designer:** This tab contains a graphical platform designer which can be used to further define the platform model. The section entitled “Managing Platform Properties Using the Platform Designer” explains the graphical platform designer.
- **Connections:** Endpoints that are exposed by platform software, and connections to such endpoints from other software and external services, are defined on this tab. The section entitled “Adding Connections” explains endpoints and connections.
- **Inventory:** This tab lists the platforms that have been provisioned using the platform model. The list of provisioned platforms is presented in the screen in tabular format. The following information is presented for each provisioned platform in the list, and the name of provisioned platform can be clicked to invoke a “Platform Deployment” screen which includes details of the provisioned platform:
 - **Name:** The name of the provisioned platform.
 - **Stage:** The stage of the software development cycle to which the provisioned platform belongs.
 - **Deployed Applications:** The number of applications that are currently deployed onto the provisioned platform.
 - **Status:** An icon indicating the status of the platform provisioning process, for example, provisioned () , failed () , deprovisioning () , etc.

- **Stored Parameters:** This tab lists the stored parameter sets that have been saved for the platform model. The list of stored parameter sets is presented in the screen in tabular format. The following information is presented for each stored parameter set in the list:
 - **Name:** The name of the stored parameter set.
 - **Owner:** The name of the owner of the stored parameter set.
 - **Updated:** The last time that the stored parameter set was updated.
 - **Action:** Clicking the [Edit](#) link in this column opens a dialog box in which the stored parameter set can be edited.

Note: HP CDA initially displays these tabs and the More tab. When you select an item from the **More** drop-down menu, it replaces tab 5, and tab 5's topic becomes the first item in the **More** drop-down menu.

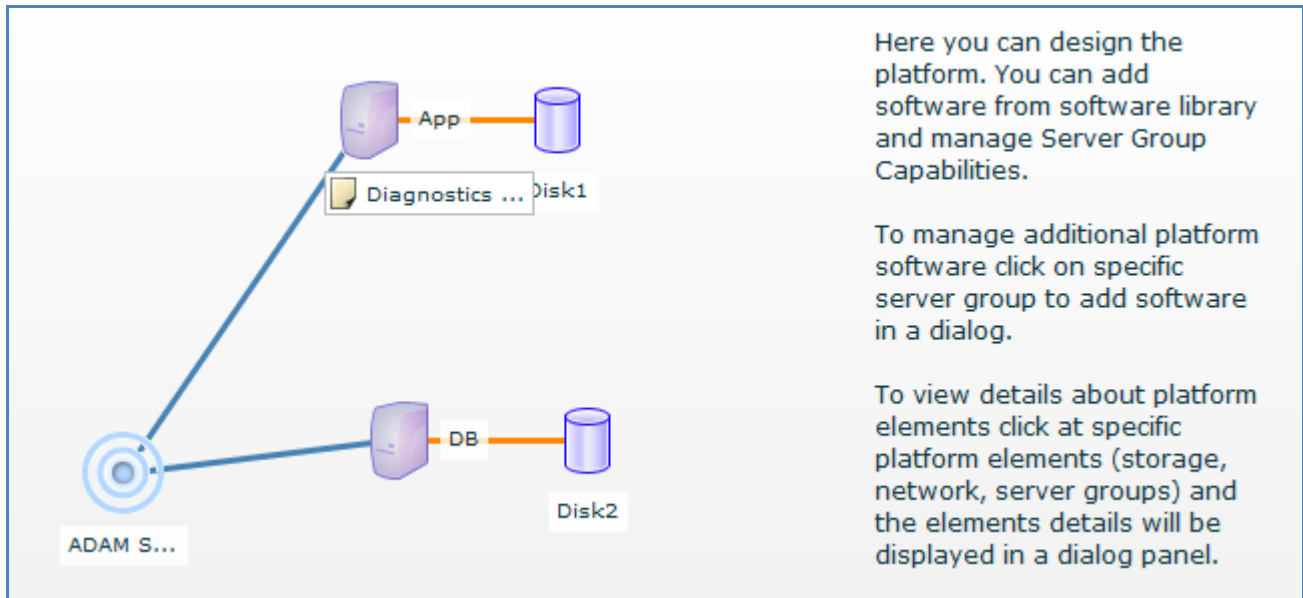
- **More:** When you click **More**, a drop-down menu displays. The menu initially offers the following options:
 - **Specification:** allows you to enter or edit specifications for the platform model in a text editor window. Clicking the **Source** button in the text editor displays the HTML formatting tags in the specification.
 - **Access Rights:** displays the access rights of the various CDA user roles in a tabular format, and allows those access rights to be modified.
 - **History:** lists the revision history of the platform model in a tabular format, and allows revisions to be compared.
 - **Events:** lists events related to the platform model in a tabular format, and provides search and filtering capabilities for the events.

Note: When you select an item from the **More** drop-down menu, it replaces tab 5, and tab 5's topic becomes the first item in the **More** drop-down menu.

Managing Platform Properties Using the Platform Designer

After a platform model has been created and saved, certain properties of the platform model can be amended and revised as is necessary using HP CDA's *platform designer*. Figure 7 shows a graphical representation of a platform model in the platform designer.

Figure 7, Example Platform Model Displayed in the Platform Designer



The platform components shown in the platform designer, their layout, and the connections between them cannot be modified in the designer. That information is set in the infrastructure template upon which the platform mode is based.

The example platform model shown in Figure 7 consists of the following platform components:

- **Network** – The symbol labeled “ADAM S...” represents a network from which the NICs of the compute resources will obtain their IP addresses and over which they will communicate. Clicking the network symbol opens a dialog box containing certain properties of the network configuration. Network properties cannot be changed in the platform designer.
- **Compute Resources** – The symbols labeled “App” and “DB” represent the compute resources specified in the model. The color of compute resource symbols in the platform designer indicates whether the compute resources are physical or virtual server groups. Physical server groups are yellow, and virtual server groups are purple. Additionally, the symbols for cloud-based virtual compute resources contain a small cloud symbol in the center of them. Clicking a server group symbol opens a dialog box in which you can view as well as add and modify certain properties of the server group.
- **Storage** – The symbols labeled “Disk1” and “Disk2” represent the storage specified in the model. The color of the symbols indicates whether the storage is physical or virtual. Physical storage symbols are yellow, and virtual storage symbols are purple. Clicking a storage symbol opens a dialog box containing certain properties of the storage. Storage properties cannot be changed in the platform designer.
- **Connections** –The lines between the symbols represent connections between those platform components.

The following sections explain how to display the platform designer and manage server group properties.

Displaying the Platform Designer

To display the Platform Designer, complete the following steps:

1. Display one of the platform screens by clicking either the [My Platforms](#) or [All Platforms](#) link that is on the **Platforms** tab in the HP CDA interface.
2. Click the name of the platform model that you wish to display in the platform designer from the list of platforms listed in the screen that you invoked above.
A screen containing details of the chosen platform model displays.
3. Click the **Designer** tab to open the platform designer.

The following sections describe how you can specify capabilities by tagging the server groups in the platform designer with capability tags. HP CDA contains several default tags from which you can select when you perform the following procedures. If you do not see a desired tag or tags, the HP CDA administrator can add them for you.

Assigning Software Capabilities

Software capabilities identify the software that will be installed on a platform that is provisioned from the platform model that is displayed in the platform designer screen. The application developer user can then subsequently utilize these assigned software capabilities to determine if server groups within a given platform have the necessary software capabilities required to support the application layers in an application model being designed. Software capabilities are assigned on a per server group basis.

To assign software capabilities, complete the following steps:

1. Click the symbol in the platform designer screen for the server group to which you want to assign software capabilities.
A server group “Properties” dialog box opens in the display.
2. Click on the **Software** tab in that dialog box if it is not selected.
The software tab contains two areas that show the pre-installed and add-on software capabilities that are currently specified for the server group.
3. Perform the following actions, as is appropriate:
 - To add “pre-installed” software capabilities:
 - a. Click **Add**.
 - b. Select the Pre-installed SW radio button in the “Add New Software” dialog box that appears and click the Next > button.
 - c. In the tree view that appears, select the software capability to be added, select the **Managed by Software Add-On** checkbox if necessary.
 - d. Click **Next >**.

The **Managed by Software Add-On** checkbox determines which screen displays next.

- **Checkbox Selected** – Select the add-on software that will manage the pre-installed software, click Next > and then perform the actions stated under **Checkbox Not Selected** below.

- Checkbox Not Selected – If the software being added has parameters associated with it, modify the parameter values if necessary and then click the Finish button. If the software being added does not have parameters associated with it, click the Finish button.

Note: You can double-click on any software capability in the list after it has been added in order to modify associated parameter values if necessary.

- e. Click **Save** to save the changes.
- To add “add-on” software capabilities:
 - a. Click the **Add** button.
 - b. Select the **Add-on SW** radio button in the “Add New Software” dialog box that appears and click the **Next >** button.
 - c. Select the add-on software from the list that appears, and click **Next >**.
 - d. If the software being added has software capabilities associated with it *and* the software represented by the capabilities have parameters associated with them, modify the parameter values if necessary and then click the **Finish** button. If the software being added does not have parameters associated with it, click the **Finish** button.
 - e. Click the **Save** button in the server group “Properties” dialog box to save the changes.
 - To remove software capabilities, click on the software capability to be removed from the list, click the **Remove** button, and then click the **Save** button in the server group “Properties” dialog box to save the changes.
 - To adjust the order of software capabilities in a software capability list:
 - a. Click on the software capability to be moved in either the “Pre-installed SW” or “Software Add-ons” list in the dialog box, and use the **UP** and **Down** buttons as is necessary to adjust the position of the capability in the list.
 - b. Repeat the above step as is necessary to reorder the list of capabilities as is desired.

Note: Changing the order of the capabilities in the “Pre-installed SW” list only affects the way the list is presented in the interface. Changing the order of the capabilities in the “Software Add-ons” list affects the way the list is presented in the interface, and also the order in which the add-on software is installed when the platform is provisioned.
4. After the capabilities list has been reordered as desired, click **Save** in the server group “Properties” dialog box to save the changes.

Assigning Hardware and Operating System Capabilities

Hardware and operating system capabilities identify the capabilities of compute resources of the platform that is to be provisioned from the platform model that is displayed in the platform designer screen. The application developer user can then subsequently utilize these assigned capabilities to determine if server groups within a given platform have the necessary physical/virtual hardware and OS capabilities required to support the application layers in an application model being designed. These capabilities are assigned on a per server group basis.

To assign capabilities, complete the following steps:

1. Click the symbol in the platform designer screen for the server group to which you want to assign hardware and OS capabilities.

A server group “Properties” dialog box opens in the display.

2. Click on the **HW & OS** tab in that dialog box if it is not selected.

The hardware and OS tab contains two areas. The “Available Capabilities” on the left side of the dialog box show the available hardware and OS capability tags, and the “Assigned Capabilities” on the right side of the dialog box show the currently assigned hardware and OS capabilities. Capabilities that were assigned to the platform at the infrastructure template level are shown in gray font under “Assigned Capabilities.” Such template-assigned capabilities cannot be changed in the server group “Properties” dialog box.

3. In the tree view under “Available Capabilities,” drill down to and select a capability to be assigned, and then click the > button to move it to the assigned capabilities list.

Note: Use the < (less than) button to remove capabilities from the assigned capabilities list.

4. Repeat the above step as is necessary to move the remaining necessary hardware and OS capabilities to the assigned capabilities list.
5. Click **Save** to save the changes.

Specifying Deployer Parameters

The platform model needs some information pertinent to the deployer that will be used during platform provisioning. That information is configured on the Deployer tab of the “Properties” dialog box.

1. Click the symbol in the platform designer screen for the server group to which you want to view or specify deployer parameters.

A server group “Properties” dialog box opens in the display.

2. Click on the **Deployer** tab in that dialog box if it is not selected.
3. Modify the parameters explained below as necessary:

Note: If you select HP SA as the deployer in the **Deployer** drop-down field, then the **User Name** and **Password** fields are not used.

- **User Name** – Enter the username of the user to be used to log into the deployer specified in the “Deployer” field into this field.

Note: When Microsoft Windows domain -based resources are involved, the user name needs to be in the format <Domain>\\<user>

When cloud-based resources are involved, the user name to enter into this field is provided by the cloud service provider when the service account is set up.

- **Password** – Enter the password of the user specified above into this field.

Note: When cloud-based resources are involved, the password field is inactive, as authentication is provided via a supplied key file rather than a password.

- **Deployer** – In this drop-down field, select the name of the plugin configuration for the deployer that will be used during platform provisioning. Deployers that have been integrated with HP CDA via plugin will appear in this field.

Note: When Chef is used as a deployer, a separate Chef server must be set up for deployment onto private cloud-based virtual resources, public EC2 cloud-based resources, and public HP Cloud Services cloud-based resources. For cloud-based deployment, the Chef servers must be set up in the cloud.

- **Remove Software when deprovisioning** – Select this checkbox to remove software from the server when the server is deprovisioned. Since virtual servers are deleted during deprovisioning, selecting this checkbox in such cases serves no purpose. Selecting this checkbox for a physical server can be useful.
 - **Unregister Server when deprovisioning** – Select this checkbox to unregister the server being deprovisioned from the deployer. Since virtual servers are deleted during deprovisioning, leaving this checkbox unchecked in such cases will leave a useless registration record of the server in the deployer database. Leaving a registration record of a physical server in the deployer database can be useful.
4. Click **Save** to save the changes.

Adding Connections

A platform model can contain descriptions of communication *endpoints* that are exposed by the platform software that is installed and run on the platform after provisioning. Endpoints typically consist of a protocol and port number that other software and external services can use to communicate with the platform software. For example, database platform software might expose as an endpoint the database port for MySQL through TCP on port 3306.

Connections describe an association to software and external service endpoints needed by platform software. For example, web server platform software might need to communicate with a database server in order to save and retrieve data. If the database server's host and port number are abstracted as an endpoint, a connection from the web server can be associated to that endpoint.

To add a connection, complete the following steps:

1. Invoke one of the platform screens by clicking either the [My Platforms](#) or [All Platforms](#) link that is on the **Platforms** tab in the HP CDA interface, as is appropriate.
2. Click the name of the platform model for which you want to add connections from the list of platforms listed in the screen that you invoked above.

A screen containing details of the chosen platform model is invoked.

3. Click the **Connections** tab in the screen that was invoked above.
4. Click the [New Endpoint](#) link to invoke the "New Endpoint" dialog box.
5. Enter the following parameters into the dialog box, as described. Parameters that are mandatory are followed by a red asterisk (*) in the screen:
 - **Software Instance** – In the drop-down field, select the platform software for which the endpoint is being created.
 - **Name** – Enter a name for the endpoint. This is the name that will identify the endpoint in user interface elements, and to which a connection can be specified.
 - **Description** – Enter a description for the endpoint, if desired, or leave blank.
 - **Path** – Optionally, specify what is exposed as part of this end point. On an application tier, for example, this could be a web-context exposed at the end point.
 - **Port** – Enter the port to be exposed.
 - **Protocol** – Select the protocol to be used for communications via the endpoint.
 - **Load Balanceable** – Select this checkbox to indicate that the layer represented by this end-point will be load balanced, and if a load balancer is part of the topology, the request will be distributed. If this checkbox is not selected, requests will go to the specific server hosting this tier.

Note: If you choose load balancing by selecting the **Load Balanceable** checkbox, you will need to enter load balancer parameter values in the deployment topology map. For details and instructions, see [Deployment Topology and Load Balancing](#) on page 68.

6. Click **OK** to save the endpoint and close the dialog box.

5 Monitoring Administration

Monitoring Administration Overview

You can configure monitoring to observe application availability and performance. Monitoring is deployed together with the infrastructure and application components.

HP Continuous Delivery Automation (HP CDA) integrates with industry-standard monitoring tools and receives events from them. The HP CDA user interface displays monitoring status based on the events received. You can view collected monitoring data by launching the monitoring tool UI from the status shown in HP CDA.

The HP CDA application model and the monitoring model are integrated for deployment purposes. Monitoring is deployed/un-deployed as the application is deployed/un-deployed.

Like the rest of HP CDA, HP designed monitoring to be open and extensible. HP CDA 1.00 provides integrations for HP Diagnostics, HP SiteScope, and Nagios. For integrations with additional monitoring applications, contact your HP representative.

Often, a variety of monitoring tools is employed to obtain sufficient performance and availability metrics and perspectives. Generally speaking, there are two broad approaches that monitoring tools use:

- Agent-based: tools that require you to install software (i.e., the agent) on the monitored system.
- Agent-less: tools that do not require you to install agents, but do require access to the monitored systems. This access may entail opening firewall ports between the tool and the monitored system, as well as configuring credentials (for example, usernames and passwords and/or certificates) for remote access. Agent-based solutions tend to require less configuration than agent-less solutions. This is a general categorization and some monitoring tools exhibit aspects of both agent-less and agent-based approaches.

Each approach has its strengths, and using a combination of monitoring tools is often desirable. Generally speaking, agent-less approaches (such as HP SiteScope) are ideal for monitoring availability from strategic locations in your computing environment. When you monitor availability close to your users' access point, you gain a more accurate assessment of what they are experiencing. Often, services that are up and running appear to be unavailable, due to network access issues. Generally speaking, agent-based approaches (such as HP Diagnostics) are ideal for monitoring performance. Performance data can be much more voluminous than availability data, and agents can help manage these higher data rates.

After you identify your monitoring tools, you perform three main steps to configure them:

- creating monitors
- organizing monitors into policies
- assigning policies to application deployment topologies

When you deploy an application, the monitors are set up according to the policies that were assigned to the application's deployment topology. As HP CDA receives threshold crossing events from monitoring tools, the corresponding monitoring status, displayed in the HP CDA user interface, is updated.

You can also update a policy adding a new monitor definition (or removing an existing monitor definition) and then re-deploy the policy to all locations where it is currently deployed.

Configuration and Deployment of Monitors and Monitoring Policies

To configure monitoring, complete the following steps:

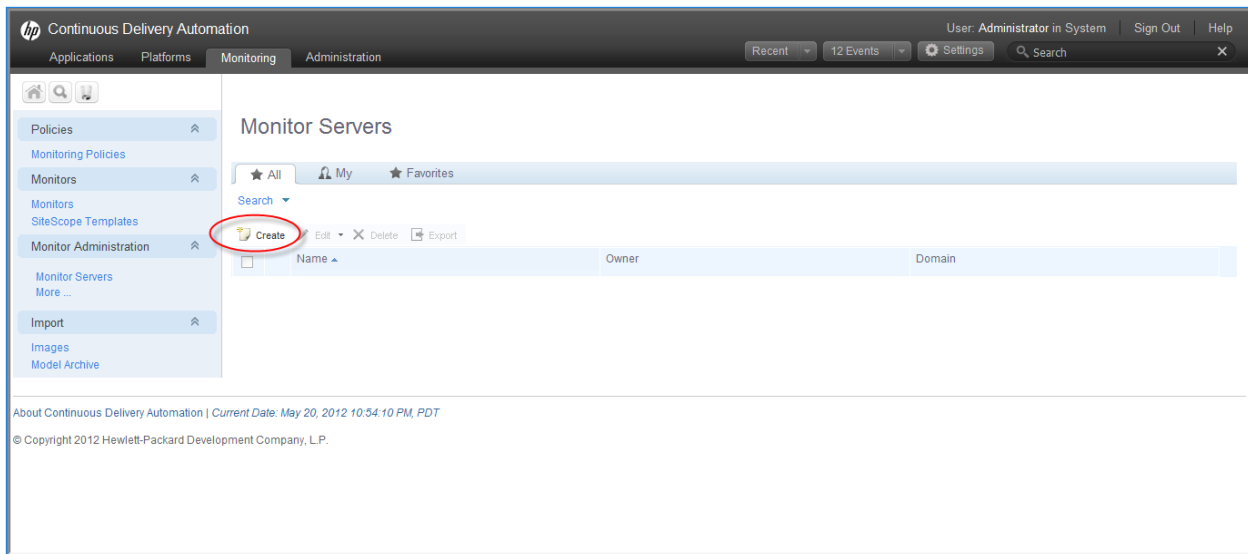
Identify the monitoring tools in your environment and create Monitor Servers.

1. Define monitors.
2. Define monitoring polices.
3. Assign policies to Application Deployment Topologies.
4. Review the state of monitored applications.

Step 1: Identify the monitoring tools in your environment and create Monitor Servers.

Monitor Servers contain parameters for connecting to a monitoring tool. To create a monitor server:

1. Go to the Monitoring tab.
2. Click **Monitor Servers** in the left panel.
3. Click **Create**.



4. Enter a name.
5. Click **Select**.
6. Select **Diagnostics** or **SiteScope** for Monitor Provider Definition (monitoring tool).
7. Click **Add**.
8. Click **Next**.
9. Enter the tool-specific connection information as indicated.
10. Click **Finish**.
11. Configure your monitoring tools to send events to HP CDA.
 - To configure HP Diagnostics, the following are required:
 - Script Execution on Alerts enabled.
 - Change the value of “Minimum time between events for a single rule” from default value 1800 to 0.

- A URL for posting events to CDA host:

http://<cda host>:<CDA port>/mon-sis-wer/sisreceiver

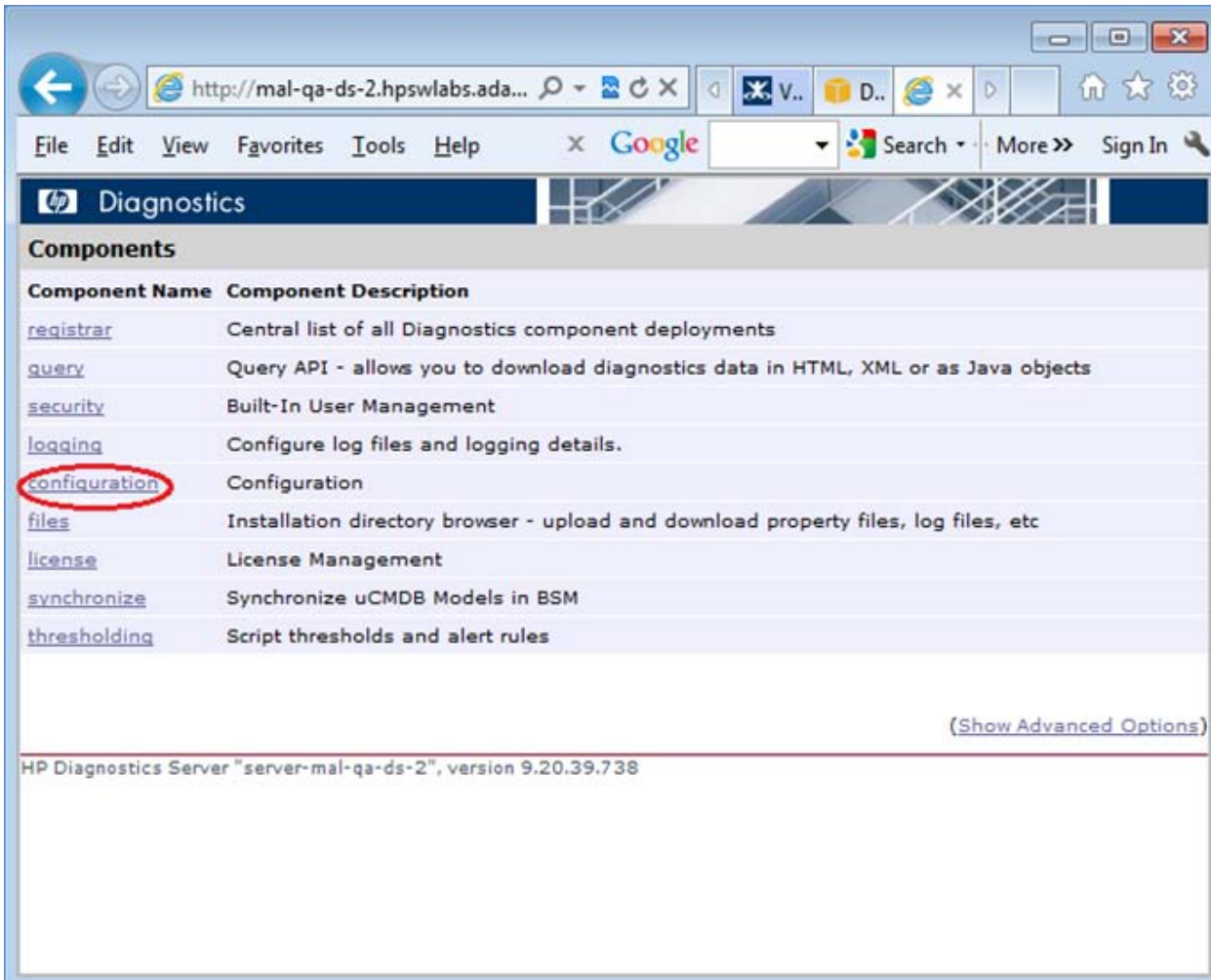
This URL can be tested from a browser in HP Diagnostics host to get a timestamp from HP CDA, thus verifying connectivity.

- Script status mapping

unavailable,unavailable,unavailable,warning,good,error

Note: The status mapping script cannot contain spaces or tabs.

HP Diagnostics has a configuration user interface, which is separate from its performance analysis user interface for performing these tasks.



The screenshot shows a web browser window displaying the HP Diagnostics Configuration page. The browser's address bar shows the URL: <http://mal-qa-ds-2.hpswlablads.ada...>. The page title is "HP Diagnostics". Below the title, there is a "Configuration" section with a table of settings. The "Alert Properties" link is circled in red.

Name	Description
Customer Information	Properties that are used to configure the customer information for the Server.
Alert Properties	Properties that are used to configure the alert settings for the Server. All changes take effect dynamically, without server restart.
Component Communications	Properties that are used to configure how this Diagnostics Component communicates with the other diagnostic components.
Memory Diagnostics	Properties that configure the way that the Server will handle memory diagnostics
Online Cache	Properties that are used to configure the Server's Online Cache.
logging	Configure log files and logging details.

[Show Advanced Options](#)

HP Diagnostics Server "server-mal-qa-ds-2", version 9.20.39.738

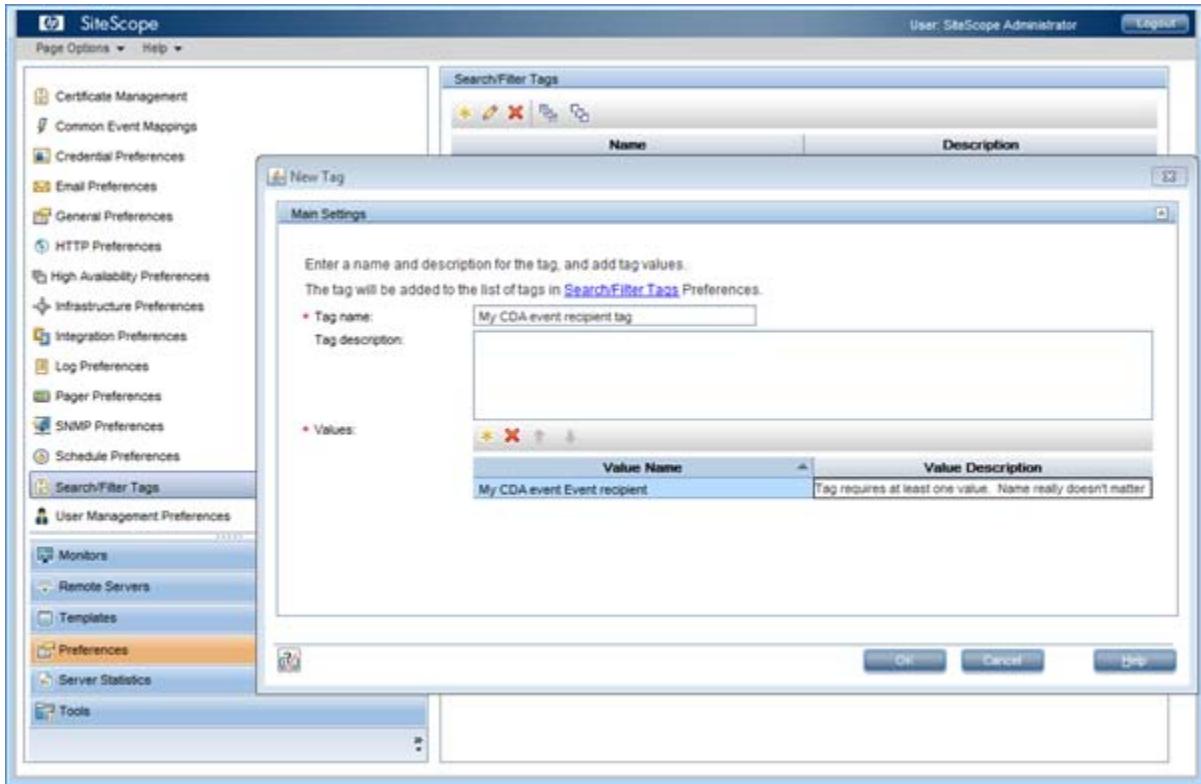
The screenshot shows a web browser window with the URL `http://mal-qa-ds-2.hpswlab.../Diagnostics`. The page title is "Diagnostics" and the main content area is titled "Alert Properties". It contains a table with configuration options for alerting and email notifications.

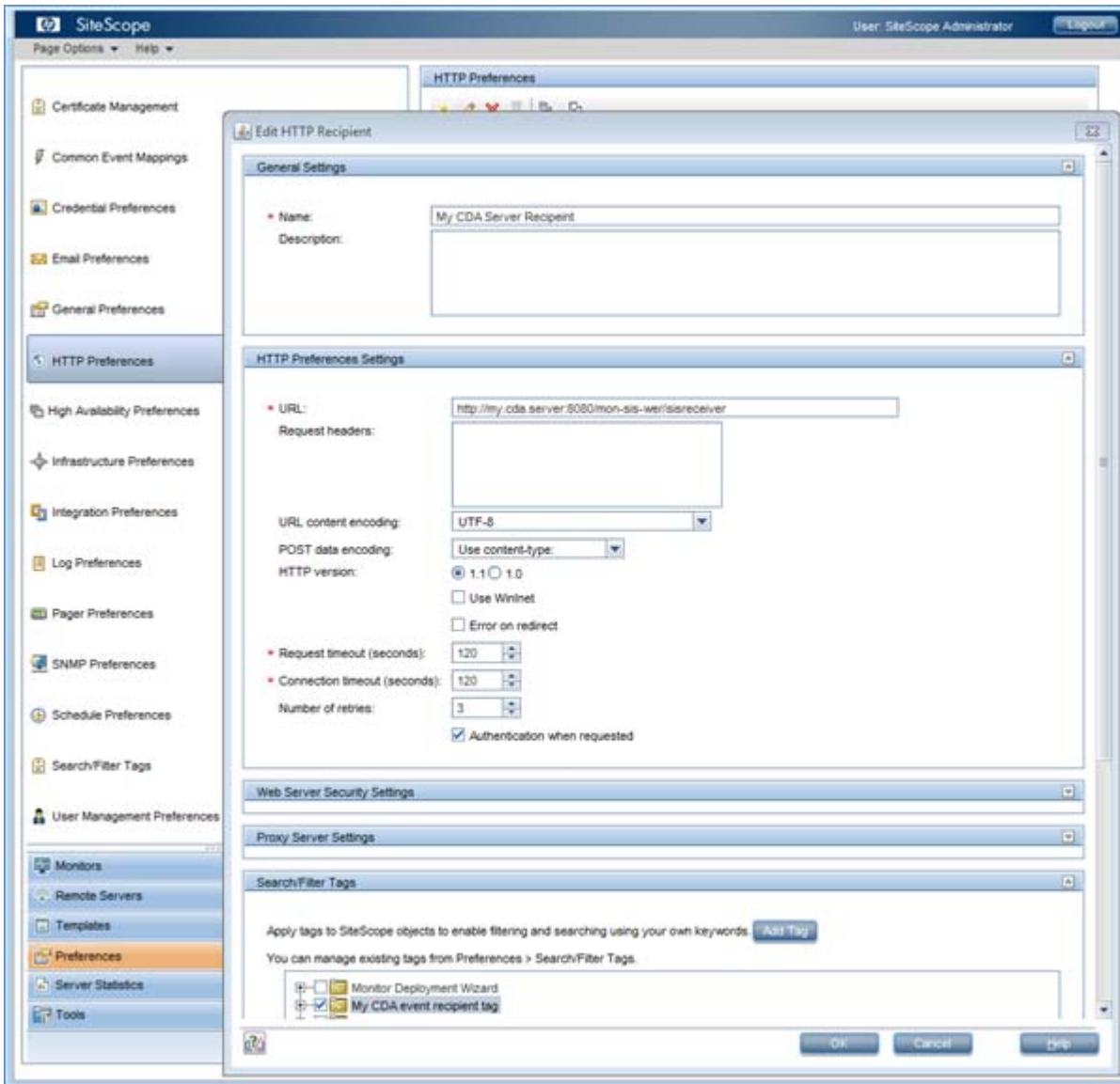
Name	Value	Description	Default Value
Alerting Enabled	<input checked="" type="checkbox"/>	This switches alerting on and off entirely for this server.	true
SNMP Server	<input type="text"/>	The IP address or fully-specified hostname to which the Diagnostics Server will send SNMP traps.	
SNMP Port	162	The port to which the Diagnostics Server will send SNMP traps.	162
SNMP Community Key	public	The SNMP community key which the Diagnostics Server will use when sending SNMP traps.	public
SMTP Server	<input type="text"/>	The IP address or fully-specified hostname to which the Diagnostics Server will send email via SMTP.	
SMTP Port	25	The port to which the Diagnostics Server will send email via SMTP.	25
SMTP From Address	<input type="text"/>	Email address from which alerts will be sent	
SMTP Default Email Addresses	<input type="text"/>	Default email addresses for SMTP alerts (comma-separated list)	
SMTP Admin Email Addresses	<input type="text"/>	Administrator email addresses for internal server issues (comma-separated list)	
Enable Script Execution on Alerts	<input checked="" type="checkbox"/>	This switches on or off whether script execution on alerts will be allowed.	false
Script Date Format	ISO 8601	The format to use for the TIME_CREATED environment variable. This can be "ISO 8601" or a format based on Java's SimpleDateFormat class.	ISO 8601

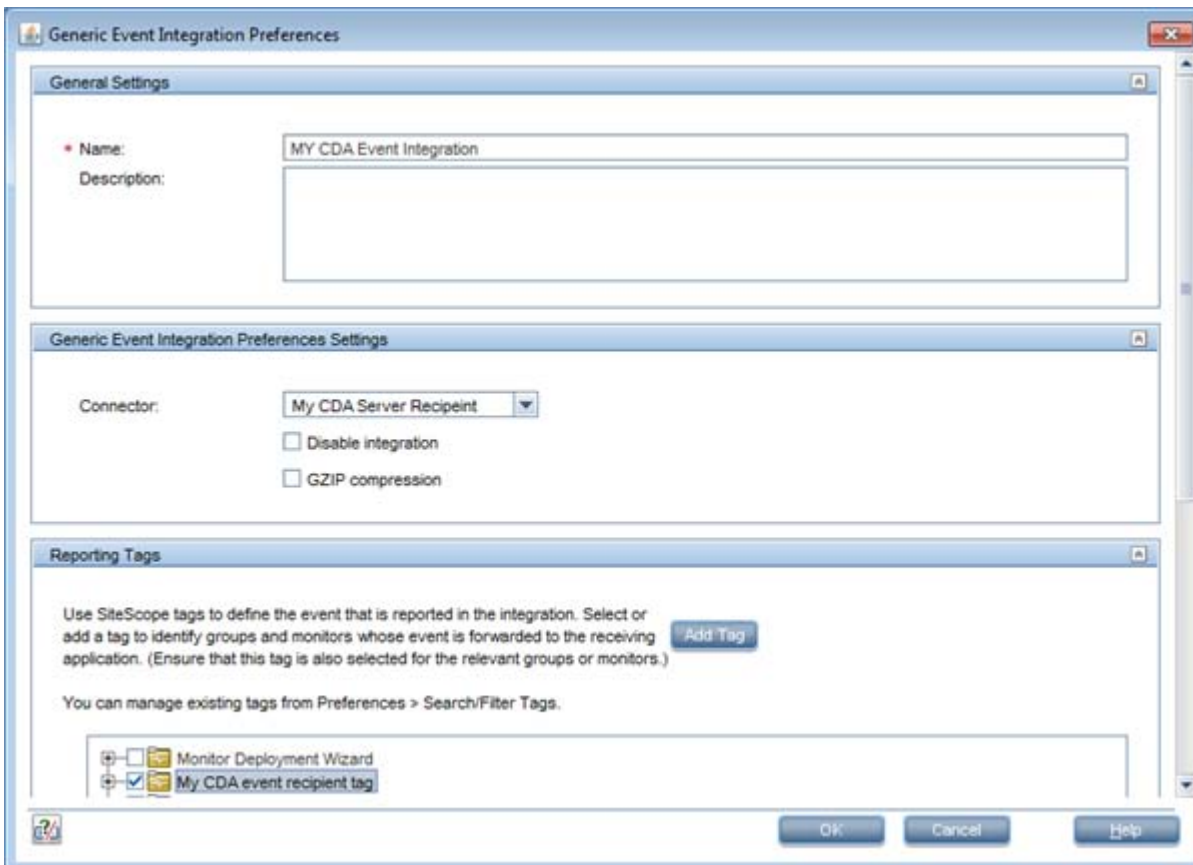
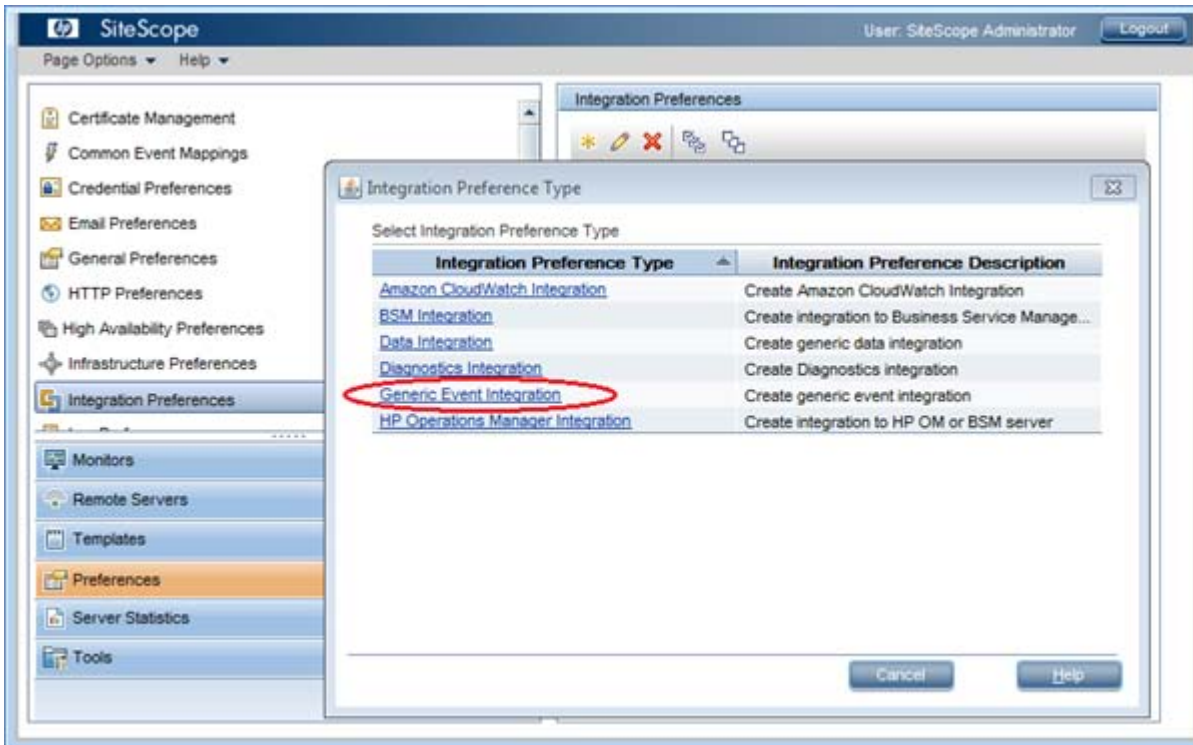
At the bottom of the configuration area, there are two buttons: "Submit" and "Reset All". Below the buttons is a link labeled "Show Advanced Options" which is circled in orange. At the very bottom of the page, the text reads "HP Diagnostics Server 'server-mal-qa-ds-2', version 9.20.39.738".



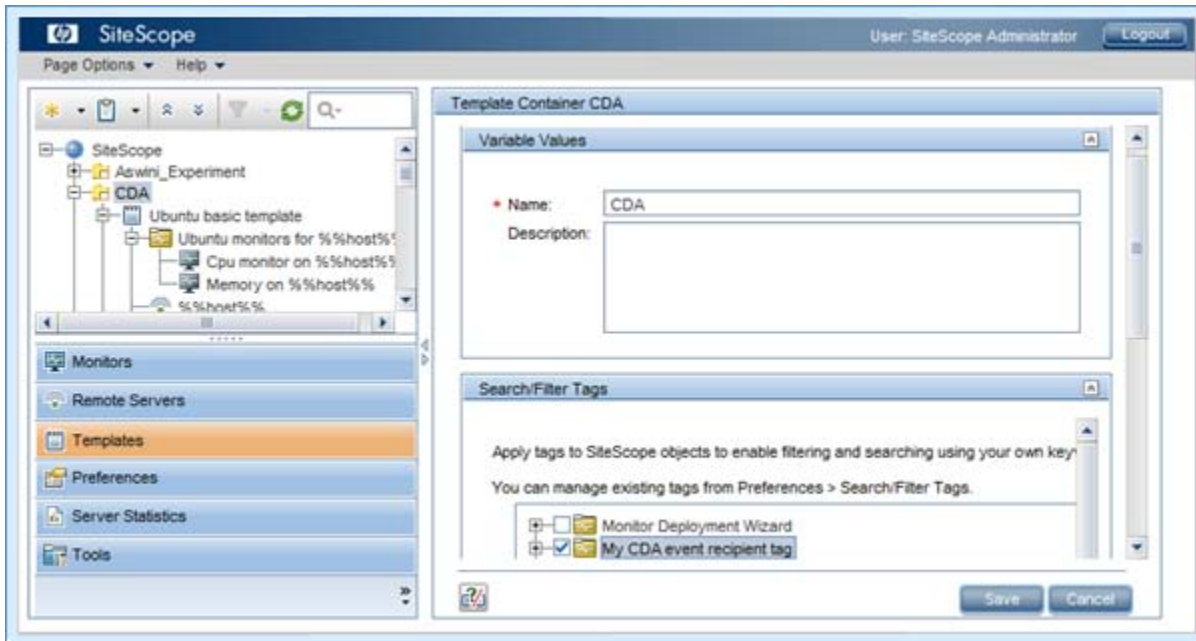
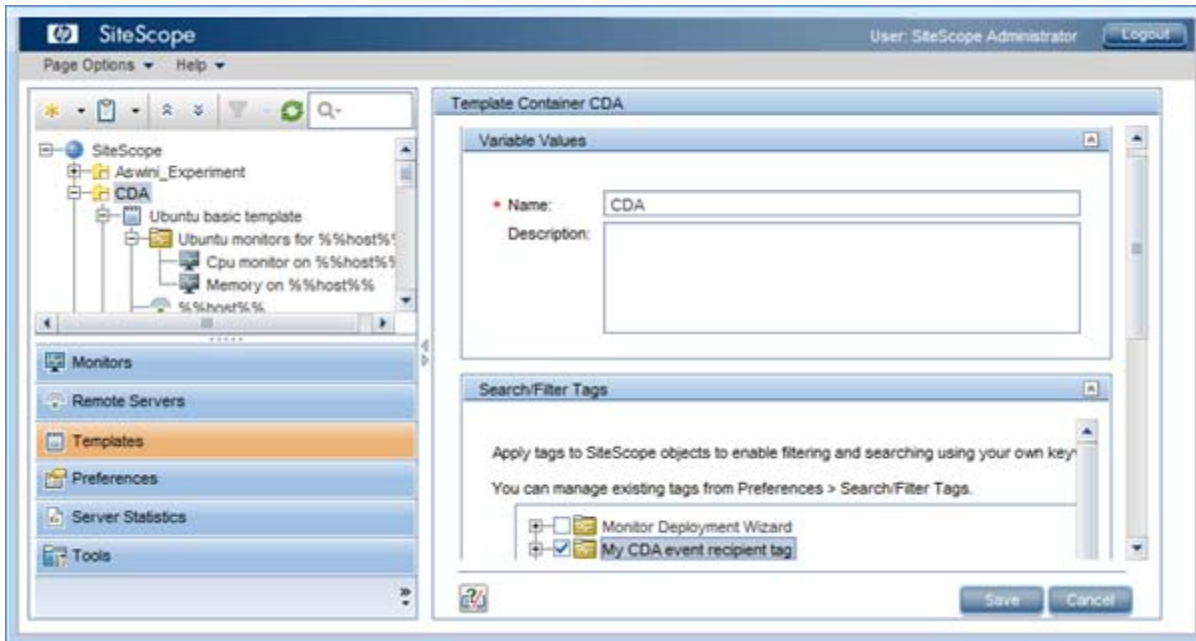
- To configure HP SiteScope, the following are required:
 - A Search/Filter Tag
 - An HTTP Recipient containing the URL for posting events to HP CDA host with the tag applied
 - http://<cda host>:<CDA port>/mon-sis-wer/sisreceiver
 - An event integration preference defined
 - Templates that:
 - Reference the HP CDA URL via a tag
 - Use the HP CDA Event Mapping.

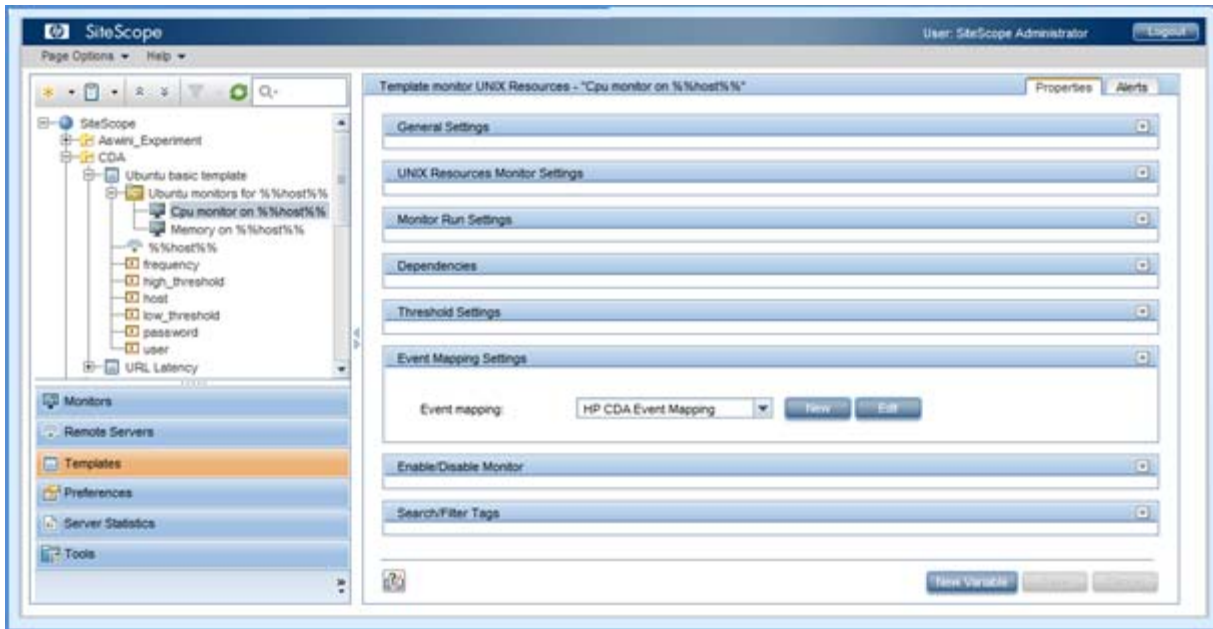






Note: When assigning Search/Filter tags to templates, any tag assigned at a level in the template path hierarchy applies to all templates below unless it is overridden at a lower level in the template path hierarchy. However, event mappings are defined on the individual monitors.





Step 2: Define Monitors.

HP CDA provides two basic monitor types:

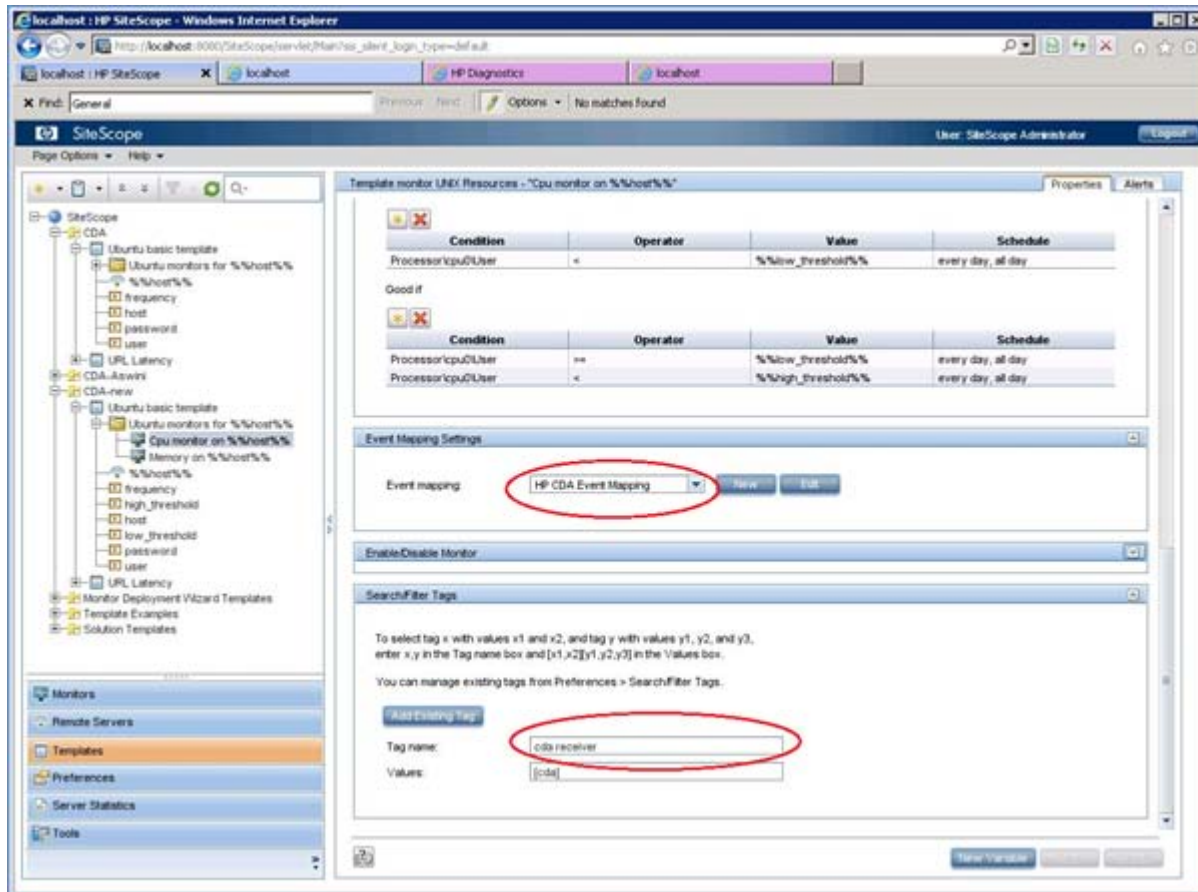
- SiteScope monitors, which are created by “importing” SiteScope templates
- Manually configured monitors, including HP Diagnostics and Nagios.

Importing SiteScope Monitors

The monitoring configuration model of HP CDA directly accommodates an existing SiteScope configuration model by leveraging your SiteScope monitoring template. HP CDA can automatically define monitors by “importing” your SiteScope templates. When the templates are “imported,” they are not physically moved or copied. The content of the template is scanned and corresponding monitor configuration data is generated within HP CDA. In environments with multiple SiteScope installations, you must ensure that corresponding templates exist on each SiteScope server to which HP CDA deploys the configured monitor(s).

Additional SiteScope templates are available from [HP LiveNetwork](#).

Note: Not all HP SiteScope templates have thresholds for generating events. Many are used for data collection, only. If your templates have thresholds for generating status events, they must have a tag set for sending events to HP CDA and they use HP CDA Event Mapping in order for HP CDA to receive status events from the HP SiteScope monitors.



To import monitoring templates from a SiteScope server:

1. Go to the Monitoring tab.
2. Click **Monitor Servers** in the left panel.
3. Click a SiteScope Monitor Server from the list of monitor servers.
4. Click **Import Templates** on right panel.
5. Click **Import**.
6. SiteScope templates are imported from the designated SiteScope Server. All templates are imported, but no existing templates are overwritten.

Note: To replace a template with an updated version, you must first delete the older template on HP CDA, and then import the new template.

7. To view the imported templates, click **SiteScope Templates** on the left pane.
8. Click an imported template to view its parameters. HP CDA parameters have been created for each SiteScope variable in the template. The syntax for SiteScope variable names has been altered to HP CDA format.

SiteScope templates can contain multiple monitor definitions and are always deployed as a unit. All monitors defined in the template are deployed together.

Individual monitors within a SiteScope template cannot be shared. Only the entire template can be shared. This follows the SiteScope configuration paradigm that templates “own” the monitors they contain, and other templates are not allowed to refer to or share them. A template contains parameters and all the monitors within the template share those common parameters.

Manually configuring monitors

Aside from the “HP SiteScope monitors” created by importing SiteScope templates, all monitor definitions include a single monitor. All monitor definitions (SiteScope templates and non-SiteScope monitors) can be shared by multiple monitoring policies.

To define an HP Diagnostics Monitor:

1. Go to the Monitoring tab.
2. Click **Monitors** in the left panel.
3. Click **Create**.
4. Enter Name.
5. Enter the Realized Monitor Name Template field. This is a template that controls how monitors are named within the monitoring tool when they are set up.

Note: HP Diagnostics uses icons (not names) to represent monitors, so this field is not used by HP Diagnostics. HP SiteScope, on the other hand, does name monitors, but in that case, the naming scheme is controlled by the import mechanism. This field may (or may not be) be used by additional monitoring tools you might be using (for example, Nagios). Refer to the integration specific documentation for the monitoring tool in those cases.

6. Enter the Contextual Launch URL Template field. This is a template for generating the URL to launch the monitoring tool when the status of the deployed monitor is clicked on in the HP CDA UI. If you leave this field blank, HP CDA will use a default URL launch template provided with the tool integration. All the parameters in the monitor definition, its related monitor server, and monitor deployer are available for creating this URL template.

For example: HP Diagnostics monitor servers have parameters such as *diag_server*, *diag_port*, and *diag_user*. The operating system deployer for HP Diagnostics has parameters such as *diag_metric_name*, *diag_high*, and *diag_low*. This monitor definition could also have additional parameters that could be used to create a URL launch template. With these parameters, construct a URL launch template such as:

http://@{diag_server}:@{diag_port}/topaz/appmon/tcAPI?action=showTxBd&customer_name=Default%20Client&view=virtual_machines_screen&select_entity=@{probeid}

For more information about URL launching, refer to the documentation of the specific monitoring tool. For HP Diagnostics, see the HP Diagnostics User’s Guide/Online Help entitled, “How to create a URL to Open any Diagnostics View.”

7. Click **Select** in the Monitor Deployer section. Every monitoring tool provides at least one monitoring deployer. For example, HP Diagnostics has two: one for Operating System monitors and another for monitoring requests.
8. Select the Monitor Deployer corresponding to the type of monitor you want to add and the monitoring tool you are using.
9. Click **Add**.
10. Click **Select** in the Metric Name section. Metric Names are used to provide consistent naming across monitoring tools. Metric names are used by HP CDA to convey the current state of deployed monitors. For example, one tool may monitor CPU utilization “% Processor Util” while another tool calls it “CPU Usage” and a third tool uses “Avg. CPU Utilization” to refer to this same basic measurement.
11. Select the Metric Name you want to add.

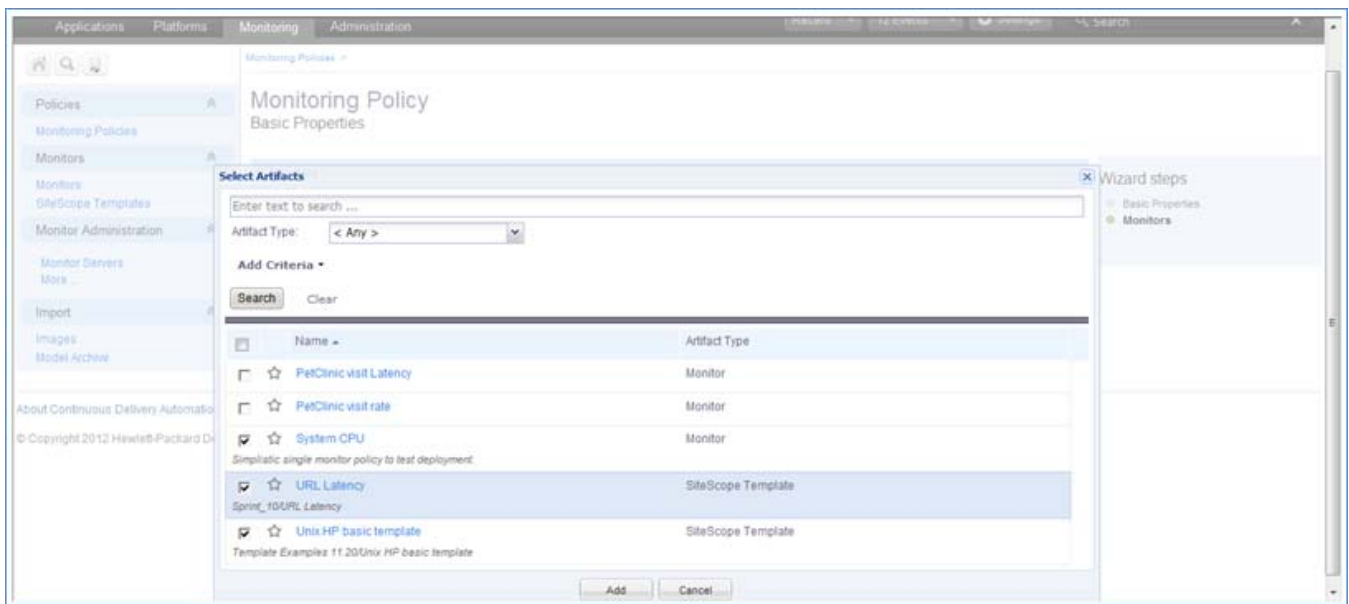
12. Click **Add**.
13. Click **Next**.
The Deployment Parameters display.
14. Enter values for the Deployment Parameters.
If the monitor will require addition parameters to be filled in when it is deployed, you can add them in the Additional Parameters section.
15. Click **Finish**.

Step 3: Define monitoring policies

Monitoring policies are a unit of monitoring deployment. You can think of a policy as an “envelope” with multiple monitors inside it. Before you can define a policy, you must define monitors for it to reference. After you define a policy, the monitors can be referred to (shared) by multiple monitoring policies.

To define monitoring policies:

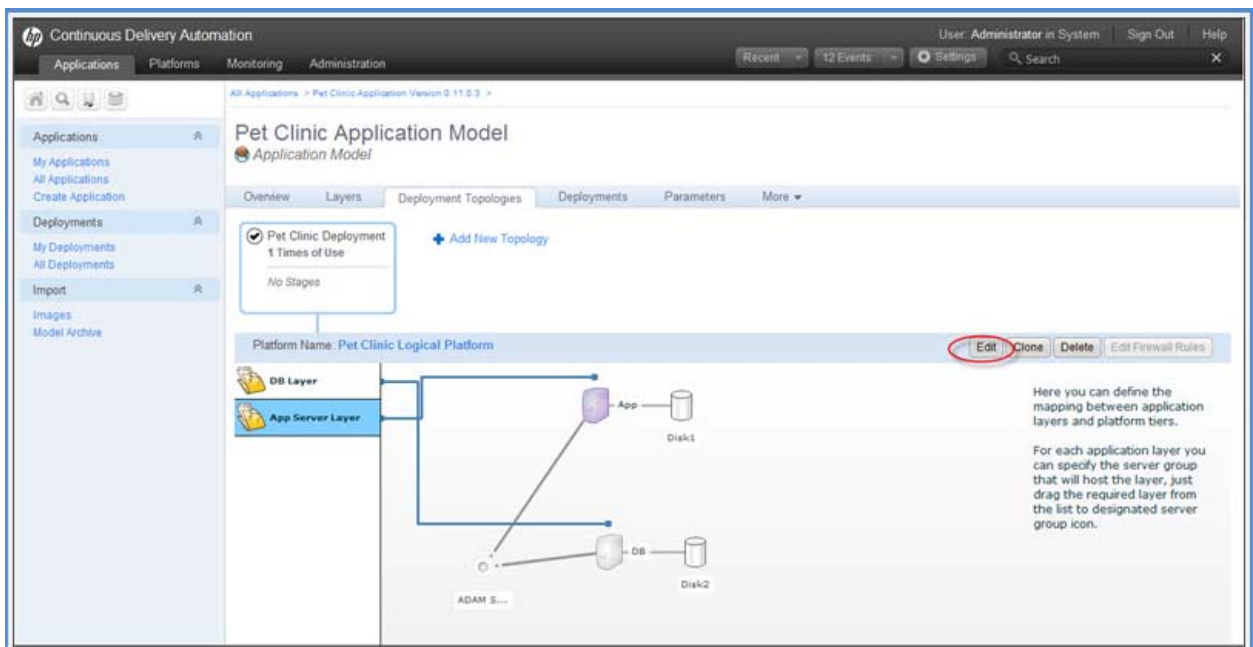
1. Go to the Monitoring tab.
2. Click **Monitoring Policies** in the left panel.
3. Click **Create**.
4. Enter a Name.
5. Click **Next**.
6. Click **Add** to add one or more monitors to the policy.
7. Check the boxes next to each monitor (or HP SiteScope Template) you want to add to the policy.
8. Click **Add**.
9. Click **Finish**.



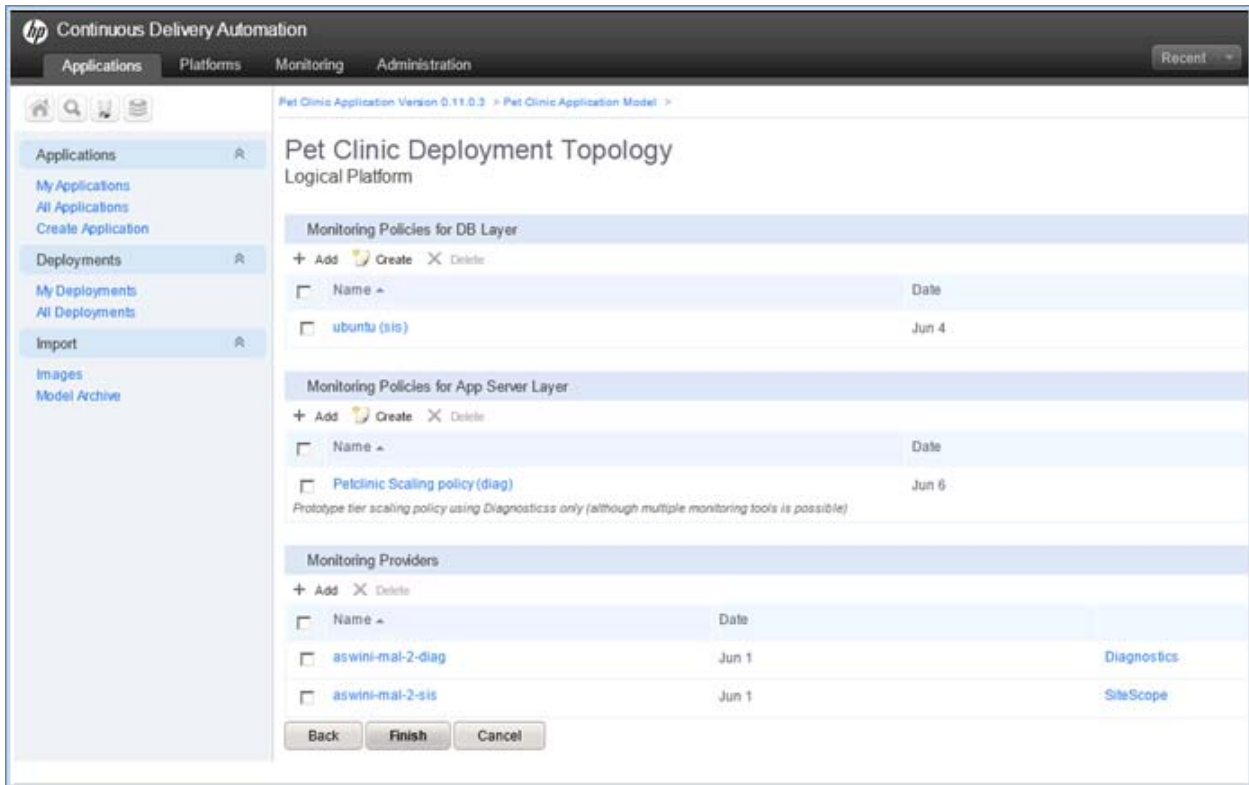
Step 4: Assign Monitoring Policies to Application Topologies.

Monitors are deployed in conjunction with application deployment. To assign monitoring policies to Application Topologies:

1. Go to the Applications tab.
2. Click **All Applications** in the left panel.
3. Select an Application version from the list.
4. Click **Application Models**.
5. Select the Application.
6. Click **Deployment Topologies**.
7. Select a Deployment topology.
8. Click **Edit**.



9. A wizard appears on the right panel.
10. Click **Next**.
11. Click **Next**.
12. For each application layer:
 - Click **Add** under the Monitoring Policies section.
13. Select the monitoring policies to deploy with the application layer.
14. In the Monitoring Providers section, click **Add** to add a monitor server corresponding to each of the Monitor Provider Policy types which have been selected for the layers.
15. Click **Finish**.

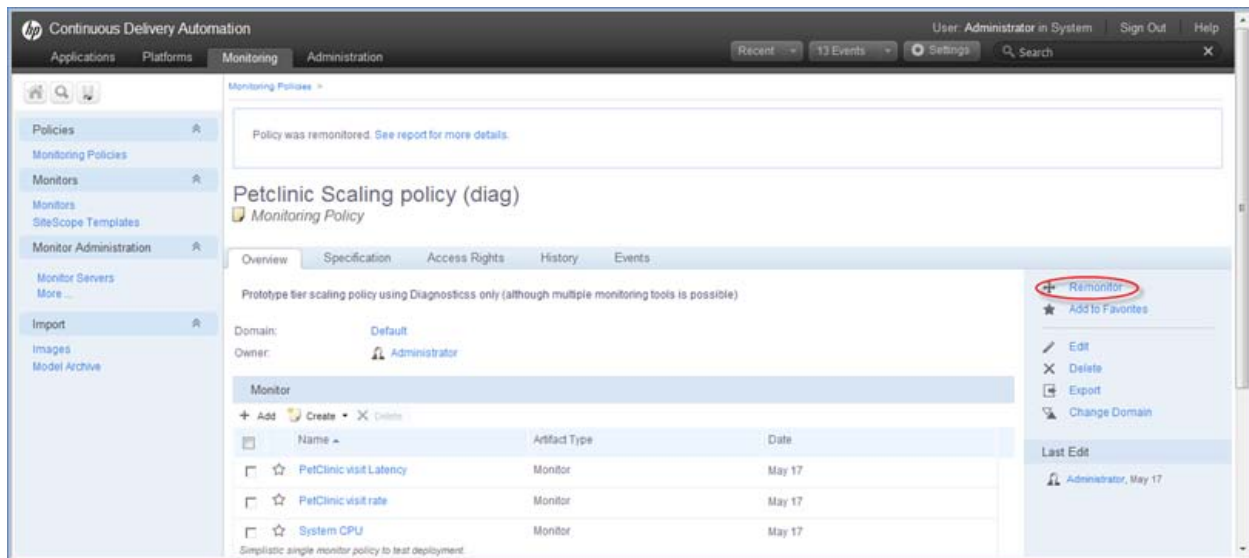


Remonitoring

If you need to add monitors to (or remove monitors from) a policy after it has already been deployed, you can edit and “Remonitor” the policy. The “Remonitor” function will update the policy everywhere it has already been deployed to contain the monitors you have added (or removed). Remonitoring was designed to completely succeed or cleanly fail. If there is a failure during a remonitor operation, none of the previously deployed policies will be modified.

To edit a monitoring policy and remonitor it:

1. Go to the Monitoring tab.
2. Click **Monitoring Policies** in the left panel.
3. Click on the policy you want to edit.
4. Add (or delete) monitors as needed, using the Add or Delete button to change the set of monitors the policy uses.
5. Click **Remonitor**.
A dialog appears allowing you to edit parameter for the monitors in the policy.
6. The policy will be redeployed everywhere it has previously been deployed.



Deploying and Undeploying Applications with Monitors

If an application layer is deployed or undeployed, the monitoring topology corresponding to the application layer at the time of deployment will also be deployed or undeployed.

HP CDA's monitoring subsystem is designed to completely succeed or cleanly fail when it processes a request to deploy or undeploy monitoring. This means there should be nothing to clean up if monitoring fails to deploy or undeploy monitors for an application layer. However, applications often consist of multiple layers, so it is possible for one layer of an application to contain successfully deployed monitoring, while another layer is left with no monitoring deployed because it cleanly failed. Rolling back the application deployment (or undeployment) should bring the layers of the application into a consistent state.

Step 5: Review the state of monitored applications.

To review the state of monitored applications, complete the following steps:

1. Go to the Applications tab.
2. Click **All Applications** in the left panel.
3. Select an application.
4. Select a Deployment (assuming that the application has been previously deployed).
5. Under the "Infrastructure Status" section, monitors display for each system the application is deployed upon. These systems are organized by the layer of the application. Monitor status (displayed as status change events) is received from the monitoring tools. Each monitor has one of four possible statuses.
 - Unknown: HP CDA has not received a status event so the status is not known.
 - High: the monitored value is above the high threshold value that was configured.
 - Low: the monitored value is below the low threshold value that was configured.
 - Normal: the monitored value is between the high and low threshold values that were configured.
6. (Optional) Click on a monitor to launch the user interface of the underlying tool. Depending on the LWSSO configuration, you may be asked to provide login credential to the underlying monitoring tool.

Monitoring Applications in Amazon Elastic Compute Cloud

Both HP Diagnostics and HP SiteScope can monitor Amazon Elastic Compute Cloud (Amazon EC2) environments independently of HP CDA. This section focuses on the connectivity between HP CDA and these tools when they are used to monitor Amazon EC2 environments. For more detailed information for how to use these tools to monitor Amazon EC2 environments, refer to the HP Diagnostics and/or HP SiteScope documentation.

HP Diagnostics and Amazon EC2

HP Diagnostics is a distributed application with three installable pieces:

- Commander: the central consolidation point for all diagnostics data
- Mediator: similar to the Commander, in that it also acts as a consolidation point for diagnostics data
- Various instrumentation probes: agents that observe and collect data from the running environment.

When using Diagnostics in hybrid cloud environments, a Mediator should be installed in the Amazon EC2 cloud and configured to communicate with an on-premise Commander. This will allow the Command to consolidate and view Amazon EC2 cloud information and local on-premise data center information. It also minimizes traffic between the two environments by allowing probes in Amazon EC2 to communicate with the Mediator that is also deployed in Amazon EC2. If SSH tunnels are used for communication, this approach has fewer tunnels to maintain than having the remote probes communicate with an on-premise Mediator or Commander.

The Mediator in Amazon EC2 should run on a system with a fixed IP-address or hostname, because the probes use this information to communicate with the Mediator. If this information changes when the Mediator system is restarted, the configuration of the various probes must be updated to reestablish communication.

When you use this recommended deployment of HP Diagnostics, HP CDA communicates directly with the on-premise HP Diagnostics Commander. The Commander communicates with the remote Mediator. From an HP CDA perspective, nothing special is required to deploy monitoring configuration. However, the Mediator must be enabled to send events back to HP CDA. This requires a VPN or SSH tunnel to be established (see below).

When a system is provisioned in Amazon EC2 with HP Diagnostics probe software, the probe configuration parameter for the Diagnostics Server should be the Amazon EC2 private network hostname or IP address of the HP Diagnostics Mediator.

HP SiteScope and Amazon EC2

HP SiteScope can be installed on-premise or in the remote Amazon EC2 cloud. When installed on-premise, CloudWatch monitors can be used for monitoring activity in Amazon EC2. Other (i.e., non CloudWatch) monitors require SiteScope communication to the Amazon EC2 systems being monitored. This communication typically requires a VPN solution or SSH tunneling.

HP CDA initiates communication with HP SiteScope to deploy monitoring configuration changes and HP SiteScope initiates communication with HP CDA to send monitor status events. To deploy monitoring configuration changes or to launch the HP SiteScope user interface, on-premise access to the HP SiteScope is required. This can be accomplished by adding HP SiteScope to an Amazon EC2 Security Group to prevent blocking by firewalls.

To send events from HP SiteScope to HP CDA, a VPN solution (recommended) or SSH tunnels are required.

Establishing SSH Tunnels

If the network connection associated with an SSH tunnel is dropped for any reason (e.g., one of the endpoints is restarted), the SSH tunnel will not be re-established automatically. Many home-grown solutions for persistent

SSH tunnels are described in various internet forums. Commercial solutions tend to be VPNs. VPNs are more flexible and more resilient than point-to-point SSH tunnels.

Despite these limitations, SSH tunnels may be applicable in very limited situations such as automated testing where the tunnels only need to be active for a limited period of time (i.e., while the tests are running). However, even in these situations, it is worth noting that HP CDA does not automate the establishment of SSH tunnels for monitoring as an out of the box capability.

Any ports on an Amazon EC2 server that will be accessed from an on-premise system must be opened in an Amazon EC2 Security Group, otherwise firewalls will block access. The default port for SSH is 22.

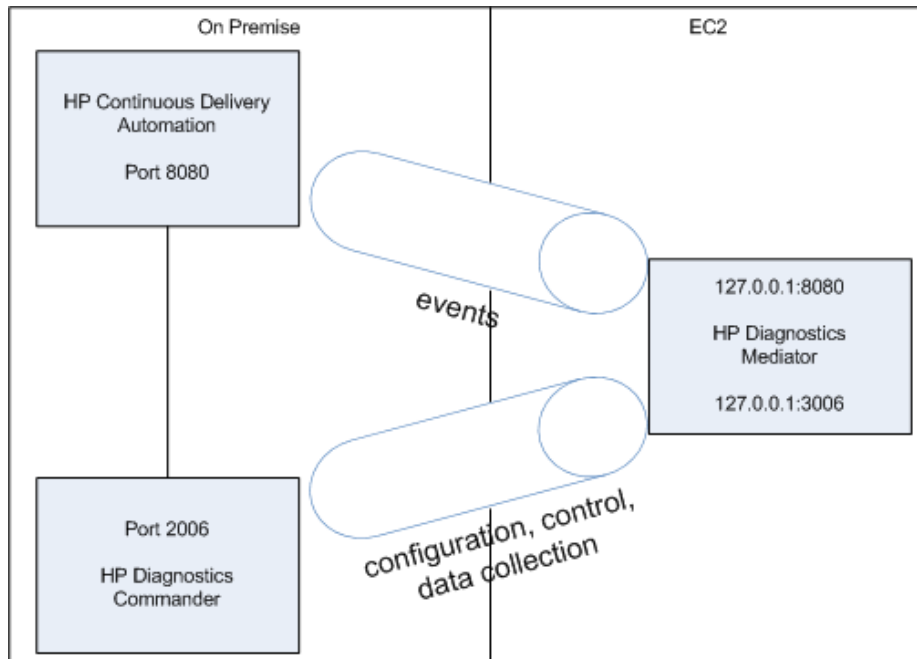
To open an SSH tunnel you need a tool such as *openssh* or *PuTTY* and a *keyfile* for authentication. The *keyfile* can be obtained from your Amazon Web Services (AWS) account. Refer to your AWS documentation for more information regarding key generation. The tunnels are established by running the SSH tool on-premise with the appropriate credentials to access the Amazon EC2 system.

Using SSH Tunnels with HP Diagnostics

HP Diagnostics requires two SSH tunnels to be established and maintained. One is for the on-premise Commander to communicate with the Mediator in Amazon EC2. The other is for the Mediator in Amazon EC2 to send events to the on-premise HP CDA server.

The following diagram depicts these tunnels using default port numbers. In this example, the HP Diagnostics Mediator is configured to send events to localhost (127.0.0.1) port 8080. This traffic is forwarded to port 8080 on the HP CDA system via the SSH tunnel. Similarly, the Commander URL is configured to localhost port 3006 and traffic is forwarded to port 2006 of the Commander which (like HP CDA) is also on premise. HP Diagnostic probes are not shown in the diagram, but those deployed in Amazon EC2 will communicate with the Mediator in Amazon EC2 and those deployed on premise will communicate with the on-premise Commander.

Figure 8 HP Diagnostics and SSH Tunnels



Using SSH Tunnels with HP SiteScope

You can deploy HP SiteScope on premise or in Amazon EC2. When used on premise, CloudWatch monitors can be used for Amazon EC2. CloudWatch monitors only require proxy settings for reaching Amazon Web

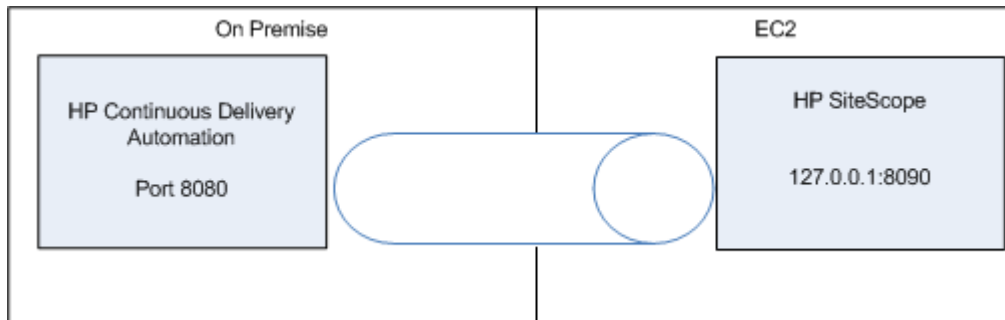
Services (AWS). Other, non-CloudWatch monitors can require modification to use an SSH *keyfile* instead of a username and password to access the Amazon EC2 environment.

In that case, you place the keyfile on the HP SiteScope system and refer to it by filename. If you make the keyfile name a variable within your HP SiteScope template, the HP CDA import mechanism creates monitor definitions with a corresponding parameter. This allows the keyfile name to be specified from the HP CDA user interface. Refer to your Amazon Web Services (AWS) and HP SiteScope documentation for more information regarding SSH key generation and management.

When HP SiteScope is deployed in Amazon EC2, it should be on a system with an elastic IP address, so the SSH tunnels can more easily be re-established (albeit manually) if/when the system is restarted.

The following diagram depicts HP SiteScope sending events to an HTTP Recipient at localhost (127.0.0.1) port 8090. This traffic is forwarded to port 8080 on the HP CDA system via the SSH tunnel.

Figure 9 HP SiteScope and SSH Tunnels

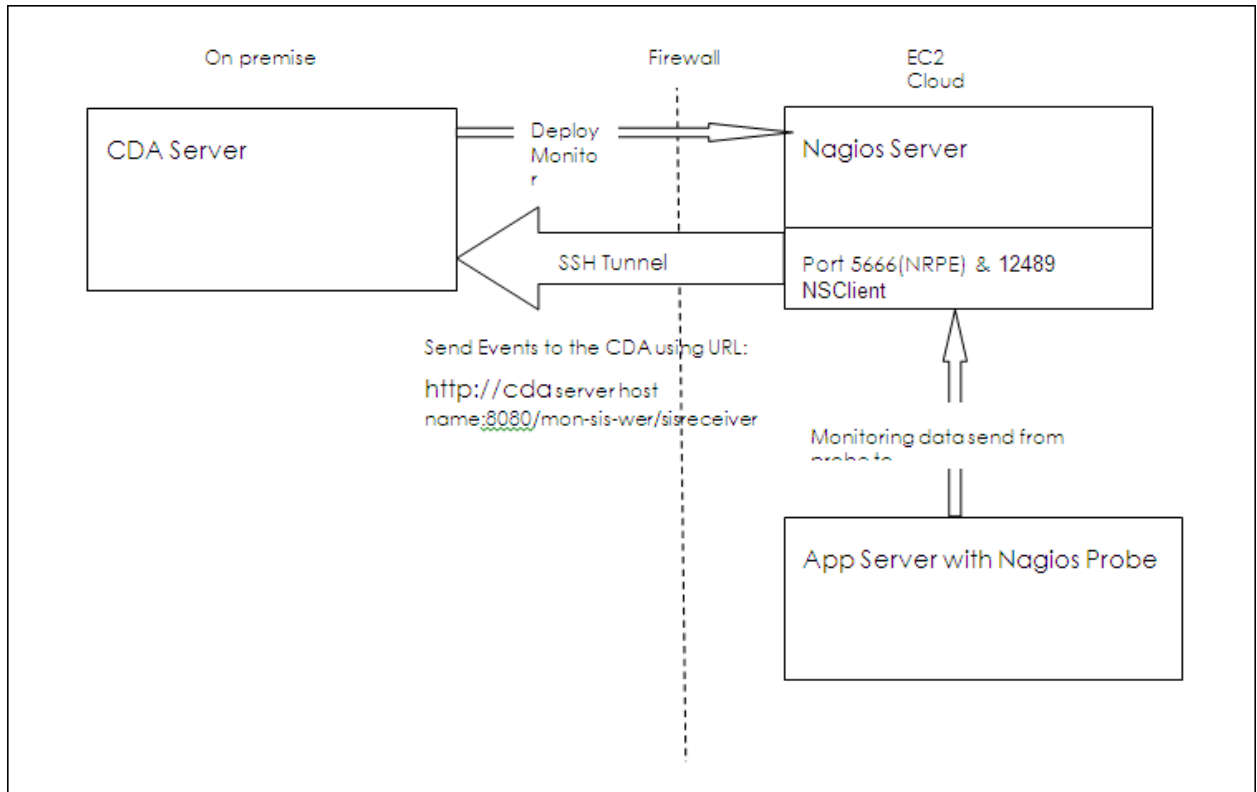


Using SSH Tunnels with Nagios

If you use Nagios server, it should be hosted in Amazon EC2 for easy communication with the agents, without needing SSH tunnels. The Nagios probe will be hosted on target machines in Amazon EC2 and communicate with Nagios server on Amazon EC2. Nagios server needs only one SSH tunnel to send the status events back to the HP CDA on port 8080. To deploy monitors, HP CDA requires http connectivity on port 80 with Nagios server in Amazon EC2, through http proxy.

The following diagram depicts the tunnel and default port communication on both ends. Amazon EC2 end security group (firewall) modification is required for communication. The Nagios server should be configured with an elastic IP address to facilitate disaster recovery.

Figure 10 Nagios and SSH Tunnels



6 HP CDA System Administration

Administration Overview

Administration in HP CDA can be broadly divided into the following areas:

- **Managing Content:** The most important content management concept in HP CDA is the use of *Domains*. You, as an administrator, can create a domain structure that represents your organizational structure. Each domain represents a working area with users assigned to specific roles within each domain and the content of the domain managed to restrict its visibility and access rights.

Administrators are also responsible for the day-to-day maintenance of the data content in the Catalog and reports about the Catalog's status. HP CDA provides a set of administration tasks that you can execute manually, or schedule to run at set times or periodically, to maintain and update Catalog data content.

- **Managing Users:** Usually, user management is delegated to an external user store, such as LDAP, where the management of people and groups who actually use HP CDA should take place. HP CDA represents users with *User* artifacts, which the particular user or an administrator can manage. Users can create additional *Contact* artifacts to represent external contacts who do not use the product –in order to associate external contacts with artifacts in the Catalog.

If necessary, you can add and manage additional local groups to organize users into groups that are not represented by the external user store.

Note: HP recommends using roles instead of creating local groups.

An important concept in HP CDA is the use of *Roles*. Roles are generic job descriptions that can apply to users and groups in specific domains. The use of roles enables the administrator to manage generic templates for lifecycle processes and security management in the top-level global domain which the resolve to specific users and groups within each working domain. Roles also control user access to functionality in the user interface.

Administrators within each domain are responsible for assigning users to roles within their domain.

- **Managing Security:** Administrators are responsible for managing and controlling user access to Catalog content. HP CDA uses Access Control Lists (ACL) to restrict access based on users, groups, or roles.
- **Managing Global Artifacts:** HP CDA uses domains to divide content into working areas, and users are assigned to specific roles within domains. A top-level domain contains all domains and global artifacts (which apply across all domains). HP CDA restricts access to these artifacts and their management to the top-level administrator.
- **Configuration and System Management:** Administrators are responsible for the configuration of each deployment of HP CDA.

The Administration tab provides access to the aspects of configuration that can be managed while HP CDA is running.

Security and Access Control

Most organizations restrict access to resources by user and group permissions. HP CDA extends this type of security by enabling the use of domain and role-based access rights.

HP CDA uses Access Control Lists (ACL) to define who can access particular resources and their permissions. Each ACL consists of a set of Access Control Elements (ACE) that define the following for a resource or collection of resources:

- **User Identification:** The user identification as a specified user, a group of users, or a role that resolves to users and groups in the domain that the artifact belongs to.
- **Granted Permission:**
 - Read Permission: Access to read the data and metadata of an artifact or resource, or a collection of artifacts.
 - Write Permission: Access to modify the data and metadata of an artifact or resource, or to create new artifacts, resources, and sub-collections, and update the metadata of a collection of artifacts. Users assigned as the owner of an artifact and administrators always have write permission.

ACLs apply in the following use cases:

- **Artifact Creation Rights:** Administrators can define which roles can create artifact types within a domain. Within the domain, the users in the allowed roles can access the artifact creation pages for the specified artifact types. The default creation rights are cumulative, so default rights given in the top-level domain apply in all other domains, and rights given to a group or role also apply in addition to rights given to each user in the group or role.
- **Governed Artifact Access Rights:** The access rights for artifacts in governance are determined by the lifecycle process applicable to the artifact. The administrator can assign rights and permissions to particular roles for each stage of a lifecycle process. Within a domain, these roles resolve to the assigned users and groups who have the specified access to the artifact at that lifecycle stage.
- **Ungoverned Default Artifact Access Rights:** In the cases where artifacts are not governed, the administrator can define which roles can read or write particular artifact types within a domain. Within the domain, the users in the allowed roles can access the artifact edit pages for the specified artifact types. In addition, you can extend this default access control functionality using particular values of categorization properties. For example, this enables you to define different access rights for services categorized as application services and for services categorized as infrastructure services. The default access rights are cumulative, so default rights given in the top-level domain apply in all other domains, and rights given to a group also apply to all the users of the group.
- **Specific Artifact Access Rights:** The owner, maintainers, and administrators of artifacts can set read or write access rights to artifacts for users, groups, or roles individually or in bulk.

Caution: HP recommends only setting specific artifact access rights for ungoverned artifacts.

Tenant Management

HP CDA can be configured for multi-tenancy which enables creation of Tenant structure in order to service multiple client organizations (tenants) by partitioning client specific data and configurations. This feature needs to be enabled at install time. Refer to “HP CDA Installation and Configuration Guide”.

Administrators can assign users specific roles in specific Tenants. This enables client organizations to compartmentalize their Catalog and restrict the access and visibility of data to only the authorized users.

A tenant can have multiple domains within itself to reflect a client’s organization structure.

To display and manage Tenants, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Tenants**.
A list of Tenants displays.
3. Click a Tenant name to view its details.

Domain Management

HP CDA enables the administrator to create a domain structure that reflects the organization of your business. Each artifact in the Catalog belongs specifically to one domain and the administrator can assign users specific roles in specific domains. This enables you to compartmentalize your Catalog and restrict the access and visibility of data to only the users who need it. For more details about the concept of domains, see [Domains Overview](#) on page 130.

To display domain information, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Domains**.
3. A list of all domains displays.
4. Click a domain name to view its details.

Responsibility for domain management is divided into the following parts:

- **Managing the Domain Structure:** Users with the Administrator role have responsibility for the overall domain structure. Only the Administrator can create and delete domains.
- **Managing a Domain:** The Administrator can assign a separate administrator for specific domains. Users with the administrator role within a domain are responsible for the following:
 - Editing the domain.
 - Setting the default role for the domain.
 - Assigning users to roles in the domain
 - Managing default permissions in the domain.
- **Setting Default Domains:** When users sign in, they access their default domain. The administrator can set default domains for users and groups and individual users can set their own default domain.

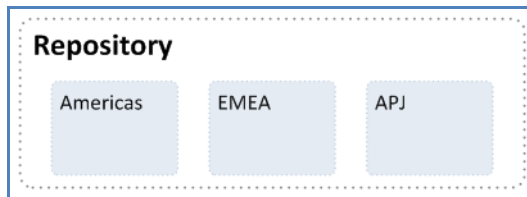
Note: By default, HP CDA contains a default domain which users who are not assigned to a specific working domain sign in to. To change which domain is the default, change the setting of the `platform.catalog.defaultUiAdapter.defaultDomain` property to the Domain ID of the required domain.

Domains Overview

Domains provide a logical separation of data within the Catalog. Each domain can represent a discrete working area for an individual department or organizational unit. This separation allows users to focus on the data that is most relevant to them and enables data to be structured by working area.

In this release, HP CDA provides support for a single layer of domains within a global top-level repository domain. For example, a domain structure representing organizational regions, Americas, EMEA, and APJ.

Figure 11 Repository Domains

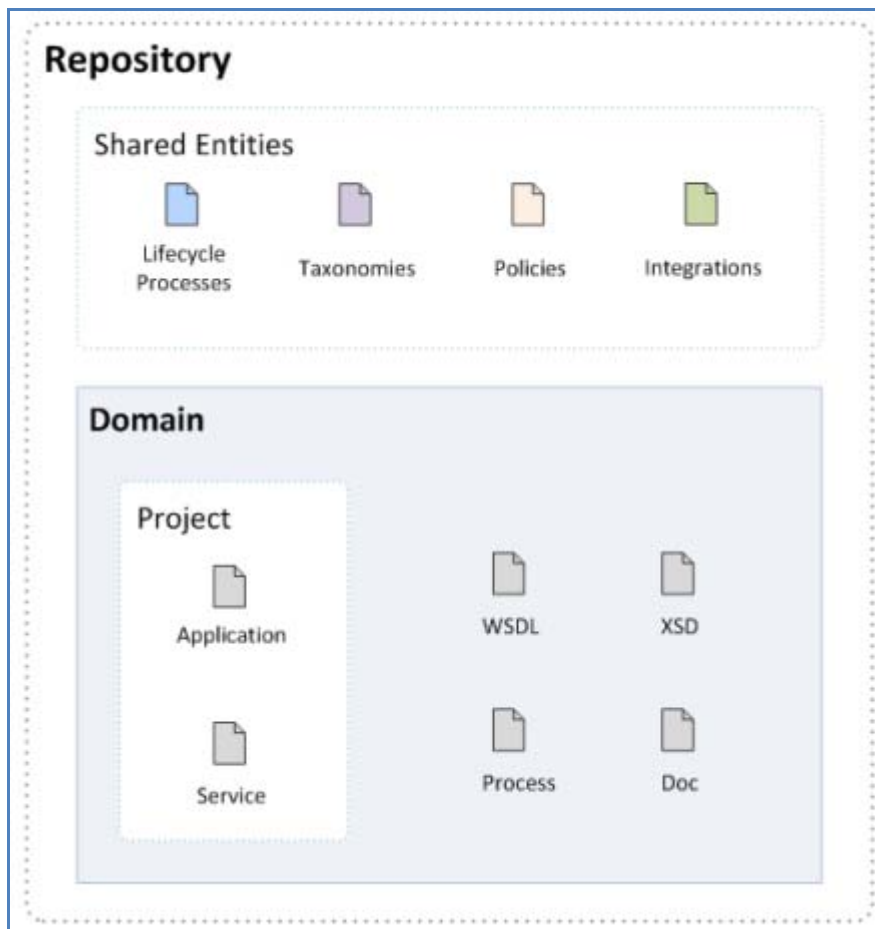


After installation, HP CDA consists of the top-level repository domain and a default domain. The default domain represents a default working area for all users until the administrator creates additional working area domains and assigns users to them.

Note: The administrator can disable default domain sign-in, so that users must be assigned to a specific domain before they can sign in.

The top-level domain is a special domain containing system and global data, such as lifecycle processes, policies, and taxonomies, which apply across all domains, and each working domain contains the specific data relevant to users of that domain.

Figure 12 System and Global Data in a Repository



The separation of data between the top-level and working domains is as follows:

- Top-Level Domain (Global Configuration):
 - Lifecycle Processes
 - Taxonomies
 - Policies
 - Roles Definition
 - System settings (including SDM and UI customizations)
- Working Domains
 - Artifacts (for example, Services, Applications, WSDLs, and Documents) that belong to the domain
 - User Role Assignments (for example, Joe is an architect in the EMEA domain)
 - Default Settings (for example, a default server folder)
 - UDDI Registry / STM Servers

Note: Working domains inherit all settings applied in the top-level repository domain. For example, access rights, roles, and lifecycle processes set in the top-level domain apply in all domains.

This domain structure creates a logical separation, not only between departments or organizational units, but also between global functions and working area domain functions.

Users in HP CDA perform specific functionality based on the roles assigned to them and the user interface restricts their access to functionality and artifacts based in these roles.

User roles are split into the following user types:

- **Top-Level Repository Administrators:** Global administration with responsibility for the following functional areas:
 - Domain Management for all domains.
 - Lifecycle process Administration.
 - User and Group Management.
 - Role Administration.
 - Server Configuration Management.
 - Policy Administration.

Note: Top-Level Repository Administrators can also access all the functionality of Domain Administrators.
- **Domain Administrators:** Users assigned to the administrator role in a specific domain with responsibility for the following functional areas:
 - Domain Management for their domain.
 - Management of administrative tasks within their domain.

Note: This role should be used to grant Tenant Administrator permissions if multi-tenancy option is enabled
- **Domain Users:** Users assigned to a specific role within a domain with specific functionality associated with that role. The same user can access different domains in different roles.

Each artifact belongs to exactly one domain. This domain is set to the current domain when a user creates an artifact. Typically, the domain does not change during the artifact lifecycle, but if required it is possible to transfer single artifact or multiple artifacts from one domain to another. By default, artifacts are only visible in the owning domain, but they can be explicitly shared by all users across all domains.

Creating and Deleting Domains

Administrators have responsibility for creating and deleting domains within the Catalog.

To display and manage domains, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Domains**.

The Domains window displays and provides Add and Delete Domain functionality for Administrators.

To add domains, complete the following steps:

1. Go to the Administration tab.
2. Click **Domains**.

The Domains window displays.

3. Click **Add Domain**.

The Create Domain dialog displays.

4. Enter a Name and Description for the domain.
5. *(Optional)* Enter a Default Domain Location to be the default location for attached data content.
6. Click **Save** to create the new domain.

All working domains exist in a single layer as sub-domains of the top-level global domain. Administrators of the top-level domain have administrative rights in all domains and can assign users and groups to be administrators of specific domains.

To delete a domain, complete the following steps:

1. Go to the Administration tab.
2. Click **Domains**.
The Domains window displays.
3. Select the domains to delete.
4. Click **Delete**, and confirm your choice to delete the selected domains.

Note: You cannot delete a domain if it contains any artifacts.

The administrator of a domain is responsible for assigning users to roles in the domain.

Managing roles in a domain consists of setting a default role, and assigning users and groups to specific roles in the domain.

If the administrator assigns a role to a user or a group in the top-level domain, the role assignment applies to all domains.

HP CDA provides a Default Role that it assigns to any user who signs into a domain, but who does not have a role assignment for the domain.

To set a Default Role for a domain, complete the following steps:

1. Go to the Administration tab.
2. Click **Domains**.
The Domains window displays.
3. Select the **Overview** tab.
4. In the Default Role section, click **(Change)**.
The Choose Role dialog displays.
5. Select the role from the list. Optionally, use the filter to find a particular role.
6. Click **Select** to set the selected role as the default role for the domain.

To Assign Users and Groups to Roles for a Domain:

1. Go to the Administration tab.
2. Click **Domains**.
The Domains window displays.
3. Select the **Roles** tab.
4. Select the role that you want to add users or groups to.
5. Click **Add Member**.
The Add Member dialog displays.

6. Do one of the following:
 - Select the Users tab, and select the users to add. Optionally, use the filter to locate a particular user.
 - Select the Groups tab, and select the groups to add. Optionally, use the filter to locate a particular group.
7. Click **Select** to add the selected users and groups to the selected role in the domain.

To Assign Roles to Users and Groups for a Domain:

1. Go to the Administration tab.
2. Click **Domains**.
The Domains window displays.
3. Select the **Members** tab.
4. Click the **Edit** icon in the Roles column for the user or group you want to assign roles to.
The Change Roles dialog displays.
5. Select the roles that you want to add to the user or group. Optionally use the input dialog to search for a specific role.
6. Click **OK** to add the roles to user or group in the domain.

Note: After users or groups are assigned to roles within a domain, administrators can extend their role memberships within specific domains.

Managing Default Access Rights

A domain's administrator is responsible for defining who can create artifacts within the domain. Administrators also define who has read and write access to artifact types.

To display and manage default access rights, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Domains**.
The Domains browse view displays.
3. Select the domain that you administer to open its details page.
4. Select the **Default Access Rights** tab.
The Default Access Rights tab displays the current Artifact Creation and Read / Write Access permissions for the domain.

Note: Domains inherit all default access rules from the top-level global domain and display them as non-editable rules for the domain. Any permission rules set within the scope of the domain are additive to those global rules. The administrator of the top-level domain can edit these global rules from the detail page of the top-level domain.

To Manage Artifact Creation Rights, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Domains**.
The Domains browse view displays.

3. Click the name of the domain you want to manage to open its details page.
4. Select the **Default Access Rights** tab to view the current permissions for the domain.
5. In the Artifact Creation table, do one of the following:
 - To Add Artifact Creation Rules:
 - a. Click Add Rule to open the Add Artifact Creation Rule window.
 - b. Select the artifact type to add creation rights for from the To Artifact Type drop-down list.
 - c. In the Granted To table, click Add Member to open the Add Member dialog.
 - d. Click Browse Address Book and select the roles to add creation rights for the selected artifact type. Optionally, use the dialog search input to find a particular role.
 - e. Click Select to add the selected roles to the Granted To table.
 - f. Click Save to add the rule for the selected artifact type to the Artifact Creation table.
 - To Edit Artifact Creation Rules:
 - a. Click the [Edit](#) link for the rule to open the Edit Artifact Creation Rule window.
 - b. Use the Granted To table to add and remove roles from the rule.
 - c. Click Save to apply the changes to the rule to the Artifact Creation table.
 - To Remove Artifact Creation Rules:
 - a. Select the rules to remove.
 - b. Click Remove Selected, and confirm your decision.

To Manage Default Read / Write Access, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Domains**.
The Domains browse view displays.
3. Click the name of the domain you want to manage to open its details page.
4. Select the **Default Access Rights** tab to view the current permissions for the domain.
5. In the Read / Write Access table, do one of the following:
 - To Add Read / Write Access Rules:
 - a. Click **Add Rule** to open the Add Read / Write Access Rule page.
 - b. Select to add Read only or Read / Write access.
 - c. Select the artifact type to add read / write access for from the To Artifact Type drop-down list.
 - d. In the Granted To table, click **Add Member** to open the Add Member dialog box.
 - e. Select the roles, users, or groups to add read / write access for the selected artifact type. Optionally, use the dialog search input to find a particular role, user, or group.
 - f. Click **Select** to add the selected roles, users, and groups to the Granted To table.
 - g. (*Optional*) Use the Condition table to add conditions to the access rule. These conditions enable you to further restrict access rights based on artifact type specific rules.
 - h. Click **Save** to add the rule for the selected artifact type to the Read / Write Access table.

- To Edit Read / Write Access Rules:
 - a. Click the [Edit](#) link for the rule you want to edit, to open the Edit Read / Write Access Rule page.
 - b. Select **Read only** or **Read / Write access**.
 - c. Use the Granted To table to add and remove roles, users, and groups from the rule.
 - d. (Optional) Use the Condition table to edit conditions for the access rule. These conditions enable you to further restrict access rights based on artifact type specific rules.
 - e. Click **Save** to apply the changes to the rule to the Read / Write Access table.
- To Remove Read / Write Access Rules:
 - a. Select the rules to remove.
 - b. Click **Remove Selected**, and confirm your decision.

Caution: Do not remove write permission from the Contacts artifact type for the system#registered group. This permission is required for the registration of new users.

Note: The default read and write permissions only apply to artifacts that are not governed by a lifecycle process or where the initial stage of the governing lifecycle process does not define access rights.

Exporting Domain Content

The administrator can export the content of an entire domain. Using this functionality for the top-level domain enables you to export the entire content of the Catalog.

To Export a Domain, complete the following steps:

1. In the Domain Details page Overview tab, click the **Export** context action to open the Export dialog box.
2. (Optional) Change the name of the archive.
3. (Optional) Expand **Advanced Options**, and select from the following options:

Advanced Option	Description
Data	Select to export all artifacts that are assigned to the domain.
System Settings	Select which domain settings to export.

4. Click **Export** to create a .ZIP archive containing the domain artifacts.

Note: This process executes as a bulk operation. An information dialog opens. It informs you that the operation is in progress and displays a progress bar with options to **Stop** the operation or to **Notify Me** when the operation is complete.

The operation executes asynchronously, so you can navigate away and perform other tasks while the operation completes.

5. When the export archive is complete you are prompted for a download location. Alternatively, open the export report and click **Download Content** to save the archive.
6. The archive is available for import using the Import Repository Archive functionality.

Disabling Default Domain Sign-In

When HP CDA installs there is a default domain which users sign-in to if they are not assigned to specific working domains.

The Administration UI enables you to change which domain is the default domain but in certain circumstances (for example, where you have thousands of LDAP users) you may want to disable default domain functionality so that users cannot sign-in unless they have been specifically assigned to a domain.

To disable the default domain, you must manually modify the configuration properties file.

To disable default domain sign-in, perform the following steps:

1. Stop the application server.
2. Export the HP CDA configuration with the following command:

```
CDA_HOME/bin/export --image IMAGE --components configuration
```

3. Open *IMAGE*/configuration/configuration-properties.xml in a text editor.
4. Change the property platform.catalog.defaultUiAdapter.defaultDomain, setting the value to an empty string.
5. Save configuration-properties.xml.
6. Upload the new configuration with the following command:

```
CDA_HOME/bin/import --image IMAGE --components configuration --configuration-application --configuration-system
```

7. Restart the application server.

Note: When default domain sign-in is disabled, the administrator must enable users by creating user artifacts or importing them from the user store, and then assigning them to a specific domain.

User Management

HP CDA delegates user management to LDAP or an application server user store. Users are represented in HP CDA by user artifacts for artifact ownership, notification, and contact purposes.

User artifacts represent users in the user store and contact artifacts represent external contacts. A contact does not have a corresponding LDAP or application user store account and cannot sign in.

When users first sign in, they are authenticated against the external user store and HP CDA creates a user artifact based on their external account.

Note: HP CDA also checks new users against existing contacts. If a matching contact artifact exists, the login name is attached and the contact becomes a user artifact.

Responsibility for user management is divided into the following parts:

- Each user can manage a user artifact.
- Users with appropriate permissions can create new contacts in the Catalog tab

Caution: Only create contacts for people who need to be represented in the Catalog but who do not need to use HP CDA. Use LDAP or your application server user store to create users who use HP CDA.

Administrators are responsible for the following aspects of user management:

- Disabling and enabling users
- Assigning users to groups and roles
- Setting default domains for users)
- Setting new artifact ownership)
- Managing Groups
- Managing Roles

Disabling and Enabling Users

Administrators can block access to HP CDA by disabling users.

To disable multiple users, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Users** to open the Users browse page.
3. Select the users to disable.
4. Click **Disable** and confirm your selection to block the selected users from accessing HP CDA.

To disable or enable a single user, complete the following step:

1. Go to the Administration tab.
2. In the Administration menu, click **Users** to open the Users browse page.
3. Click the name of the user you want to Disable or Enable to open their User details page.
4. In the User details page Overview tab, click the **Disable** or **Enable** context action.

Assigning Users to Groups and Roles

The administrator can assign a user to be a member of multiple groups or roles within specific domains.

To assign users to groups, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Users** to open the Users browse page.
3. Click the name of the User to open its detail page.
4. Select the **Groups and Roles** tab to view the group details for the user.
5. In the Groups table, do one of the following:
 - To remove the users from groups:
 - a. Select the groups to remove the user from.
 - b. Click Remove and confirm your decision.
 - To add the users to groups:
 - a. Click Add to Group to open the Add Groups to User dialog box.
 - b. Select the groups to add the user to.
 - c. Click **Select** to add the user to the selected groups.

Note: The administrator can also manage the membership of groups from the group perspective.

Assigning Users to Roles:

To assign users to roles, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Users** to open the Users browse page.
3. Click the name of the User to open its detail page.
4. Select the **Groups and Roles** tab to view the role details for the user.
5. In the Roles by Domain section, click **Change** for the required domain to open the Change Role Membership dialog box.
6. Select the additional roles to assign to the user for the domain and click **Set** to apply the changes.

Note: This functionality is only available if the user is already assigned to roles in a domain and can only be used to extend the roles set there.

Setting Default Domains for Users

Administrators can set the default domain that new users sign in to.

To set the default domain for a single user:

1. Go to the Administration tab.
2. In the Administration menu, click **Users** to open the Users browse page.
3. Click a user name to open their details page.
4. In the Overview tab, Access Security section, click the Default Domain [Change](#) link to open the Set Default Domain dialog.

5. Select a default domain and click **Select**.

When users sign in for the first time they access the selected domain.

To set the default domain for multiple users:

1. Go to the Administration tab.
2. In the Administration menu, click **Users** to open the Users browse page.
3. Expand Manage Members, and select **Set Default Domain** to open the Set Default Domain dialog box.
4. Select a domain from the list and click **Select** to set the domain as the default for new users.
5. When the selected users sign in for the first time they access the selected domain.

Note: Default domains can also be set for groups and individual users can set their own default domain.

Setting New Artifact Ownership

By default, users who create artifacts become the *owner* of the artifact. The owner of an artifact has default read / write permission for the new artifact.

Administrators can configure new artifact ownership for users, so artifact ownership is not assigned to the user – it is instead assigned to a specified role or group.

To set new artifact ownership for a single user, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Users** to open the Users browse page.
3. Click a user name to open a details page.
4. In the Overview tab Access Security section, click the New Artifact Ownership [Change](#) link to open the Select New Artifacts Ownership dialog box.
5. Click **Change** to select a new owner.
6. Input a role or group search term or click **Browse Address Book** and select from the list of roles or groups.
Note: The user must be members of the selected group or assigned to the selected role in at least one domain.
7. Click **Select** to apply the new artifact ownership assignment to the selected user.

To set new artifact ownership for multiple users:

1. Go to the Administration tab.
2. In the Administration menu, click **Users** to open the Users browse page.
3. Select the users to apply new artifact ownership assignment to.
4. Expand Manage Members, and select **Set New Artifacts Ownership** to open the Set New Artifacts Ownership dialog.
5. Click **Change** to open the dialog.
6. Enter a role or group search term or click **Browse Address Book** and select from the list of roles or groups.
Note: The users must be members of the selected group or assigned to the selected role in at least one domain.

7. Click **Select** to apply the new artifact ownership assignment to the selected users.

Managing Groups

HP CDA Administrators manage two types of groups:

- groups managed by the external user store
- local groups

Note: HP Software recommends using roles instead of local groups.

Administrators are responsible for the following aspects of group management:

- Creating groups
- Managing group membership
- Assigning groups to additional domain roles
- Setting default domains for groups
- Retiring and deleting groups
- Managing group membership
- The administrator is responsible for managing the membership of local groups.

Creating Groups

To create groups, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Groups**.
3. Click **Create Group** to open the New Group page.
4. Enter a name and description.
5. (Optional) Add group members as described in "Managing Group Membership" (on page).

Managing Group Membership

Administrators can use HP CDA to manage groups. Management includes adding and removing members from the group. To manage group membership, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Groups** to open the Groups browse page.
3. Click the name of the group to open its details page.
4. Click **Edit** to open the Edit Group page.
5. In the Members table, do one of the following:
 - To remove users from the group:
 - a. Select the users to remove from the group.
 - b. Click **Remove** and confirm your decision.
 - To add users to the group:
 - a. Click **Add Member** to open the Add Users to Group dialog box.

- b. Select the users to add to the group.
- c. Click **Select** to add the selected users to the group.
- d. Click **Save** to apply your group membership changes.

Note: Administrators can also manage the membership of groups from the user perspective.

Assigning Groups to Additional Domain Roles

Administrators can assign a group to have member roles within specific domains.

To assign groups to additional domain roles, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Groups** to open the Groups browse page.
3. Click the name of the Group to open its detail page.
4. In the Roles by Domain section, click **Change** for the required domain to open the Change Role Membership dialog box.
5. Select the additional roles to assign to the group for the domain and click **Set** to apply the changes.

Note: This functionality is only available if the group is already assigned to roles in a domain and can only be used to extend the roles set there.

Setting Default Domains for Groups

Administrators can set the default domain that members of groups sign in to.

To set the default domain for groups, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Groups** to open the Groups browse page.
3. Click the name of the group to open its details page.
4. Click **Set Default Domain** to open the Set Default Domain dialog box.
5. Select a domain from the list and click **Select** to set the domain as the default for the group.
6. When members of the selected group sign in, they access the selected domain.

Note: Default domains can also be set for users and individual users can set their own default domain.

Retiring and Deleting Groups

Administrators can retire and then delete internal groups. However, these processes do not retire or delete the users who are members of these groups.

To retire multiple groups, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Groups** to open the Groups browse page.
3. Select the groups to retire.
4. Click **Retire** and confirm your decision.

To retire a single group, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Groups** to open the Groups browse page.
3. In the Group details page, click the **Retire Group** context action to open the Retire Group dialog box.
4. Click **(Change)** to select a new owner for any artifacts owned by members of the group.

To delete multiple groups, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Groups** to open the Groups browse page.
3. In the Groups browse page, select the groups to delete.
4. Click **Delete** and confirm your decision.

To delete a single group complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Groups** to open the Groups browse page.
3. In the Group details page, click the **Delete Group** context action.

Note: You must retire the group before you can delete it.

Managing Roles

Administrators use HP CDA to assign users to specific roles related to the users' job functions. These roles restrict their access to artifacts, and limit their functionality only to that appropriate to their role.

Administrator responsibility for role management is divided into the following parts:

- **Managing Roles:** Administrators are responsible for the roles available in HP CDA.
- **Assigning Users to Roles:** Administrators who have a role within a domain are responsible assigning users to specific roles within their domain and setting a default role for new users of the domain.

Roles Overview

HP CDA offers functionality across the entire service development lifecycle. In most organizations, these functions are performed by many individuals and teams with specific permissions. HP CDA uses *Roles* to enable you to define and assign these permissions, and use these assignments to focus each user or group on specific functionality and tasks and restrict their access to artifacts appropriate to their role.

Administrators define roles in the top-level domain, but user assignment to roles can be global or to different roles in different domains. For example, in the following diagram, Joe is assigned a global architect role in the top-level repository domain, Fernando is assigned the architect role in Domain A, Pam is assigned the manager role in Domain A and the architect role in Domain B, and Eric is assigned to the manager role in Domain B.

Figure 13 Roles Example



These assignments mean that in Domain A, Joe and Fernando access functionality and artifacts relevant to the architect role, whereas Pam accesses functionality relevant to the manager role.

Pam has a different role in Domain B, so along with Joe, accesses architect functionality, whereas Eric accesses functionality relevant to the manager role.

The following topics describe user role functionality in more detail:

- **User Interface Roles:** HP CDA uses roles to restrict the availability of functionality to users in appropriate roles.
- **Lifecycle Roles:** You can create Lifecycle templates with specified tasks and actions assigned to specific roles)
- **Security and Access Control Roles:** HP CDA restricts access to artifact types using ACLs, which can use roles as well as users and groups.

Some HP CDA roles are “default” roles. Administrators can extend the default roles by adding additional customized roles.

HP CDA also includes a special role, Sharing Principal, specifically associated with sharing artifacts. By default, this role is associated with the system#registered group that represents all users who access HP CDA. For more details, see [Changing the Sharing Principal](#) on page 146.

User Interface Roles

HP CDA restricts access to UI functionality according to your role. Following are roles you see in the user interface (although not all roles have user interface access):

- **Application Developer:** (default role) Application Developers have access to the Applications tab, so they can model and maintain the applications which are managed by HP CDA throughout the continuous deployment cycle.
- **Infrastructure Developer:** (default role) Infrastructure Developers have access to the Platforms tab, so they can design and maintain Infrastructure to host the Application being modeled in HP CDA.

- **Monitoring Administrator:** (default role) Monitoring Administrators have access to the Monitoring tab, so they can manage monitoring of application and infrastructure components that are developed in HP CDA.
- **Operations Manager:** (default role) Operations Managers have access to the Applications and Platforms tabs, so they can manage applications and platforms from the production perspective.
- **Provider:** (default role) Providers create, develop and manage artifacts in the repository. This role has no UI access.
- **Sharing Principal:** (default role) Sharing Principals determine who can read shared artifacts. This role has no UI access.
- **Administrator:** (default role) Administrators can use all the tabs and functionality accessible by users in the roles listed above. Additionally, they can access an Administration tab, so they can manage users, groups, roles, domains, and other system artifacts.

Lifecycle Roles

Lifecycle management makes use of roles to enable global lifecycle process management with role-based assignments in a lifecycle process template.

The administrator of the top-level domain manages lifecycle processes and uses roles to define the following:

- The user role responsible for approving particular lifecycle stages.
- The user roles with read-only, write access, and ownership permissions for artifacts at particular lifecycle stages.
- The user roles responsible for performing tasks associated with a lifecycle stage.
- The user roles that are automatically notified as a result of specific events within the lifecycle.

Note: For more details about lifecycle governance, see [Lifecycle Process Management](#) on page 157.

Within specific domains, these role assignments resolve to the users and groups assigned to the role in that domain. For example, consider the development stage of a lifecycle process for services.

The stage might consist of the following role assignments:

- A task to build and test the service assigned to the QA Engineer role.
- A stage approver in the Provider role.
- An automatic action to notify the users in the Operations Manager role when a service is approved at the development stage.

Following is information about role responsibilities within the life cycle:

- **Administrator:** responsibility for managing users, groups, roles, and system artifacts. The responsibilities of administrators vary according to the domain that they manage:
- **Top-Level Repository Administrators:** Global administration with responsibility for the following functional areas:
 - Domain Management for all domains.
 - Lifecycle Process Administration.
 - User and Group Management.
 - Role Administration.
 - Server Configuration Management.

- Policy Administration.

Note: Administrators can also access all the functionality of Domain Administrators.

- Domain Administrators: Users assigned to the administrator role in a specific domain with responsibility for the following functional areas:
 - Domain Management for their domain.
 - Management of administrative tasks within their domain.

Managing Roles

Administrators have responsibility for managing roles in HP CDA.

To view and manage roles, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Roles** to open the Roles browse page.
3. The Roles browse page provides the following functionality:
 - Delete
 - a. Select the roles to delete.
 - b. Click **Delete**.

Note: You cannot delete the default roles.
 - Create Role
 - a. Click **Create Role** to open the "Create Role Page".
 - b. After you create a role, it is available for user and group assignment, for use in lifecycle processes, and for default access rights.
 - c. Click a role name to open its details page, showing its properties and UI Access details.
 - d. Click **Edit** to change the details for a role. For system roles, you can only change the UI Access details. For details of the Edit Role page parameters, see [Create Role Page](#) on page 147.

Changing the Sharing Principal

The Sharing Principal is a special role associated with sharing artifacts to make them visible to more users. The default functionality is to use the system#registered group which represents all users who access HP CDA.

To change the sharing principal, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, select **Domains** to open the Domains page.
3. Select the Top-Level domain to open the Domains details page.
4. Select the Roles tab to view the roles and their membership for the domain.
5. Select the Sharing Principal role to view its membership.
6. Use **Remove Selected** and **Add Member** to edit the membership of the role.

Create Role Page

The Create and Edit Role pages contain the following parameters:

Parameter	Definition
Name	Name of the Role.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property.
Sub-Roles	Select from the set of existing roles to inherit their functionality and artifact access for the new role.
Additional UI Access	Sub-roles configuration can be used to provide additional UI access.
Allows Login	Allow users in the new role to access the UI.

Policy and Assertion Management

HP CDA enables you to validate your Catalog content against published policies to ensure its consistency and conformance to your business policy.

Policy management and validation uses the following artifact types:

- **Technical Policy:** A technical policy consists of a set of assertions and references to other technical policies, and serves as the central point of reference for validation. References to other technical policies enable you to collect a set of technical policies together into a larger policy enabling you to validate them collectively.
- **Assertion:** An assertion is a validation check for a single piece of data which can either pass or fail. For example, an assertion can check the following attributes of an artifact:
- **Property Values:** An assertion can check that a particular property has a value and what the value is. Typical examples include verifying that keywords are set when an artifact is created and verifying that an artifact is consumable before it enters the production lifecycle stage.
- **Related Artifacts:** An assertion can check whether a particular artifact type is associated with the artifact being verified and check property values of the related artifacts. A typical example is to verify that a service has an attached document which is categorized as a business specification document before the service is approved at the candidate lifecycle stage.
- **Data Content:** Some artifact types are expected to contain attached content. An assertion can verify that the attachment exists.
- **Artifact State:** An assertion can check various status attributes of an artifact, For example, its lifecycle status or its compliance status.
- **Metrics from Integrated Products:** An assertion can check values generated by integrated products. For example, whether there are any open defects in *HP Service Test Manager (STM)*.

Create and edit assertions using Assertion Editor. For details, see the *Assertion Editor Guide*.

HP CDA enables you to use technical policies in the following ways:

- **Lifecycle Stage Validation:** Lifecycle stages in a lifecycle process can have a set of policies associated with them that serve as validation checks for artifacts at that lifecycle stage. Policies can be optional or required and can also validate that tasks associated with the stage are complete. These policies are automatically validated when a user requests stage approval or can be manually validated by users. The collective status of these policies gives a Compliance Status for an artifact which measures the current percentage of required policies which pass validation. The administrator assigns policies to lifecycle stages during the creation of lifecycle processes.
- HP CDA includes a default Lifecycle Validation Task, scheduled to run once a day, which automatically validates all artifacts in governance against the policies that apply to their current lifecycle stage.
- **Manual Validation:** You can use the Policy Report feature of the Reports tab to perform ad-hoc manual validation. You can set up a policy report which validates a selected set of artifacts against a set of selected technical policies.
- **Artifact Form Validation:** The administrator can configure a set of technical policies that validate an artifact whenever it is created or modified. You can use this type of validation to ensure that new and amended artifacts contain a minimum set of descriptive data such as keywords and impact type, and enforce rules such as the versioning schema. Each artifact type can have its own set of form validation.

The administrator of the top-level domain is responsible for policy management.

Managing Technical Policies

To view and manage technical policies, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Technical Policies** to view the list of policies in the Catalog.
3. The Technical Policies page provides the following management functions:
 - **Create Technical Policy:**
 - a. Click **Create Technical Policy**.
 - **Delete Technical Policies:**
 - a. Select the technical policies to delete, and click Delete.
 - b. The Applicable To column enables you to filter the policies table according to the artifact type that it is applicable to.
 - c. Click the name of a technical policy to view its details.
4. The technical policy detail page provides access to individual technical policy management tasks. From this page you can perform the following functions:
 - **Add Technical Policy:**

Collect a set of related policies together under a single technical policy. This enables you to use the parent policy to perform validation against the set of policies.

 - a. Click **Add Technical Policy** to add the first technical policy, and then the standard table functionality for related artifacts.

- *Add Assertion*

Manage the assertions related to the technical policy using standard table functionality. After selecting an assertion to add, you must set its scope within the technical policy. Set the severity, and any parameters defined by the assertion.

Note: Parameters inherit any default values set in the assertion template and any non-optional parameters must be set. For more details, see "Assertion Schema" in the *Reference Guide*.

5. (Optional) You can change the severity and parameters for an assertion using the [Edit](#) link for the assertion in the table of assertions.
6. In addition, the Technical Policy details page provides the standard edit, delete, and export artifact management tasks.

Default Technical Policies

HP CDA includes a prepackaged set of technical polices.

These technical polices are described in the following tables organized by the type of validation they perform:

- **Related Artifact Validity:** Policies that verify that relevant related artifacts are present for a service, such as documentation

Technical Policy	Definition
Has Business Requirements	Verifies that an artifact has a business requirements document.
Has Approved Documentation	Has approved all document references in the given stage.
Has At Least One Approved Documentation	An artifact has at least one documentation approved.

Create Technical Policy

The Create and Edit Technical Policy pages contain the following parameters:

Parameter	Definition
Name	Name of the technical policy.
Description	A description including rich text and HTML support. This description displays in the Artifact Details page Overview tab properties area. This description is limited by the display length available in the Overview tab. If you need a longer description or an artifact specification edit the Specification property.
Technical Policies	Use the Add Technical Policy functionality of the table to manage referenced technical policies.
Assertions	Use the Add Assertion functionality of the table to manage referenced assertions.

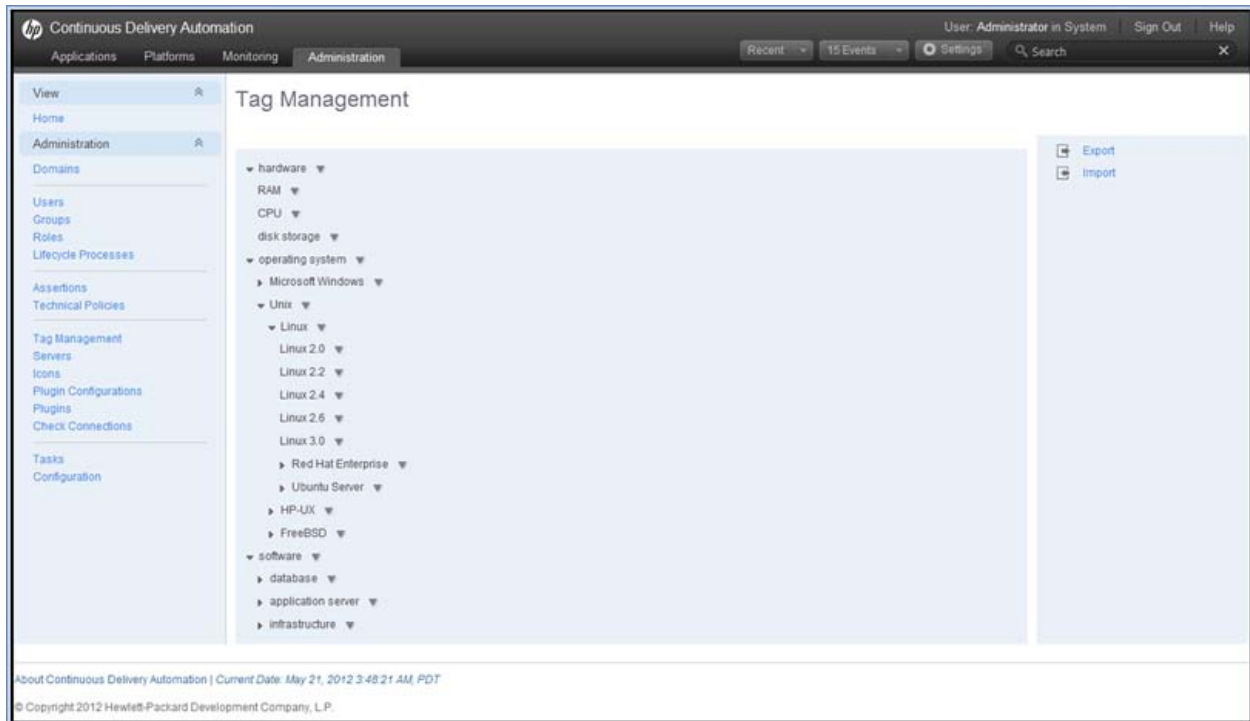
Tag Management

Tags are used to describe and define requirements and capabilities of Applications and Platforms within HP CDA. Tags are predefined classifications of these requirements and capabilities which can be used to characterize applications and platforms.

An Application Model will require specific types of servers, storage, networking, etc. on the platforms upon which it is deployed. A specific platform will have various capabilities, such as CPU, memory, operating system type and version, and other middleware, such as Tomcat or MySQL.

Tags can be added to Application Models and Platforms in order to associate these requirements and capabilities in a uniform manner. For example, when an application is deployed, only those servers which contain the tagged requirements will be offered up as available servers for the application.

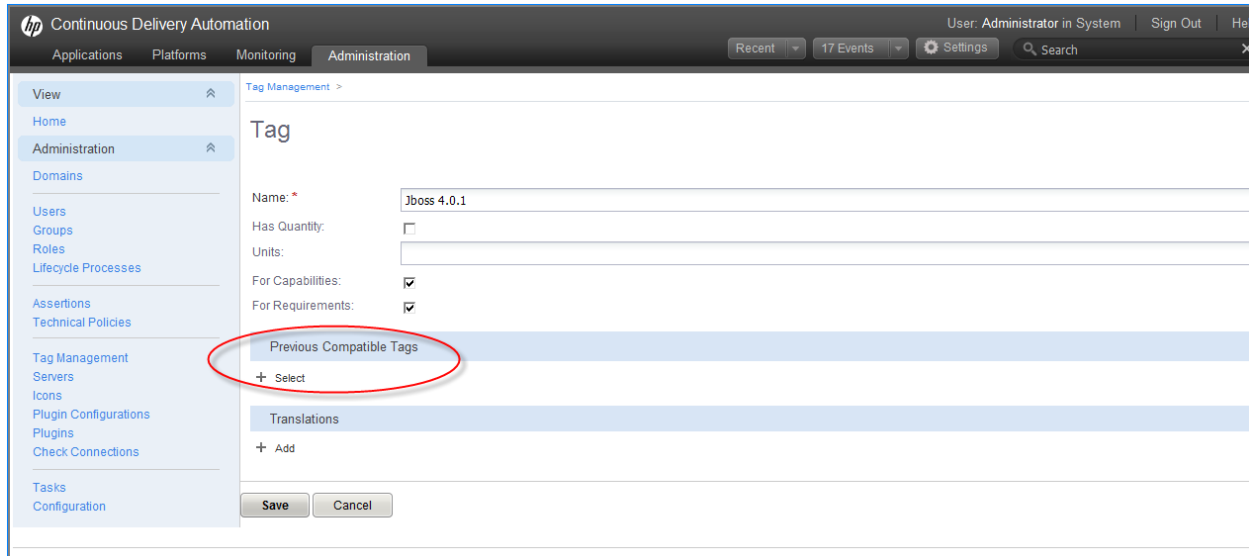
Tags can be grouped and described in a hierarchical manner by creating sibling and child tags. For example, the parent tag “operating system” might include the following child tags: “Microsoft Windows” and “Unix.” The tag “Unix” might contain the child tags “Linux,” “HP-UX,” and “FreeBSD.” “Linux” and “HP-UX” are sibling tags.



Adding, Deleting, or Modifying a Tag

To display and manage tags, complete the following steps:

1. Go to the Administration tab.
2. Click Tag Management in the left panel.
3. You will see the top level tag categories, “hardware,” “operating system,” and “software.”
4. Click on the carat to the left of each tag category to expand the category description.
5. Click on the carat to the right of a tag in order to add a child tag, add a sibling tag, move the tag up or down in the hierarchy, edit the item, or remove the item and/or its sub-tree.
6. When adding a tag, click **Select** and check the boxes next to the tags representing previous compatible tags.



Note: Tags appearing in the list are available for selection from Requirements and Capabilities sections when creating and defining software and server platforms in the “Applications” and “Platforms” tabs.

Provider Integration

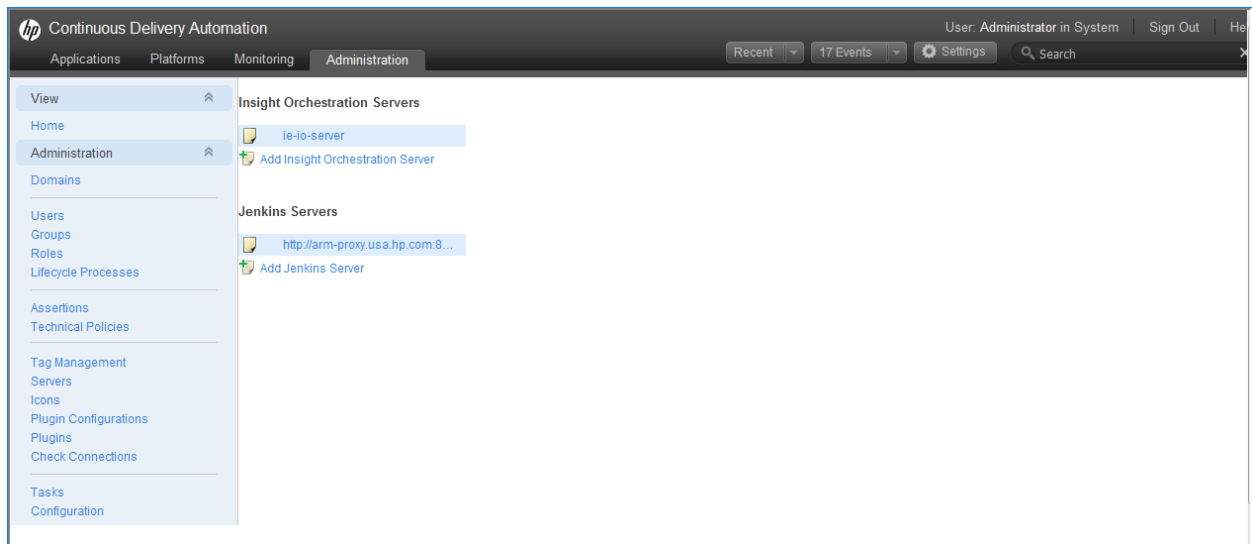
Two predefined server types are provided with HP CDA: HP Insight Orchestration (HP IO) servers and Jenkins Servers. The following information about provider integration is covered in this section:

- Defining Servers
- Integrating Providers Using Plugins
- Testing a Plugin Prior to Updating
- Predefined Deployer Types for Plugins
- Updating Plugins
- Uploading New Plugins
- Defining a Deployer Plugin

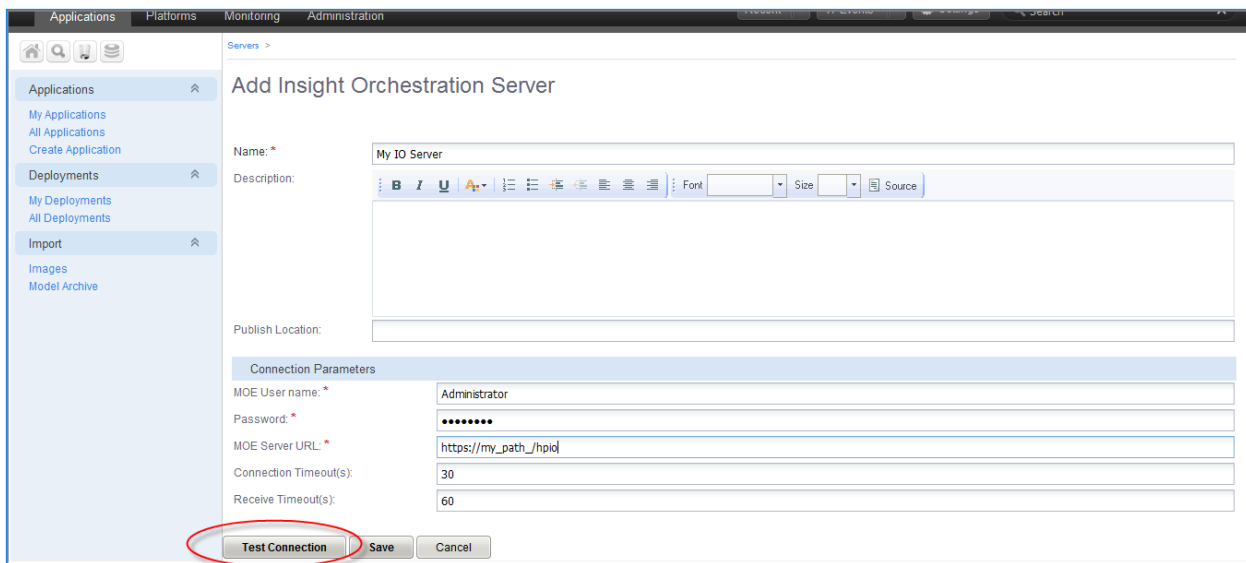
Defining Servers

To define an HP IO or Jenkins Server:

1. Go to the Administration Tab.
2. Click **Servers** in the left panel.
3. Click **Add Insight Orchestration Server** or **Add Jenkins Server**.



4. Enter the Name and Connection Parameters in order to access the Server.
5. Click **Test Connection** to make sure that the parameters are entered correctly.
6. Click **Save**. The Server now appears in the list of available Servers.



Integrating Providers Using Plugins

HP CDA uses *plugins* to load information about external services. A plugin is a way to communicate with a remote system.

Plugins can be updated while the HP CDA system is running without having to restart the server.

Plugins provide a robust way to manage your environment. The plugin environment can support various upgrade models based on the exposed risks in the environment. The capability is provided for immediate, complete replacement of a plugin in order to perform hot fixing of critical defects. Controlled integration of new plugin allows the plugin to be tested in a limited environment. Phased rollouts can be accomplished by repointing various models to the new plugin in order to limit risk. These various rollout options are described in more detail below.

Plugins can be updated in place, to allow immediate replacement of the plugin. For information on updating a plugin, see [Updating Plugins](#) on page 153.

Testing a Plugin Prior to Updating

To first test a plugin (prior to updating it) see [Uploading New Plugins](#) on page 154. Add configurations and providers, and then create a binding / test platform and application, or update an application to use the new plugin configuration. This will allow you to test the plugin with this new configuration without affecting the binding to the rest of the models.

After the plugin is tested, the models can be updated by performing one of the following actions:

- Moving existing models to the new plugin configurations in a phased manner
- Repointing the plugin back to the original configuration, and then following the procedure in the “Uploading New Plugins” section to upgrade all environments to the new plugin.

Predefined Deployer Types for Plugins

In HP CDA 1.00, plugins for the following predefined deployer types are provided:

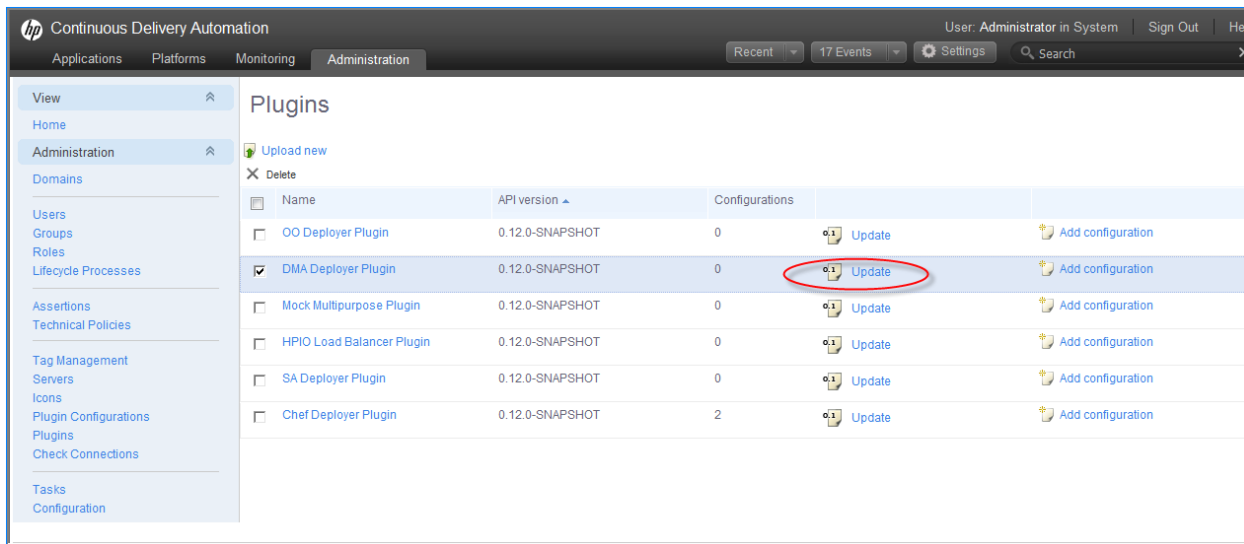
- OO Deployer
- DMA Deployer
- HPIO Load Balancer
- SA Deployer
- Chef Deployer
- A generic, “Mock” Multipurpose Plugin

Updating Plugins

To update a plugin:

1. Go to the Administration Tab.
2. Click **Plugins**.
3. Select a Plugin from the list by checking its box.
4. Click **Update**.

HP Continuous Delivery Automation User Guide

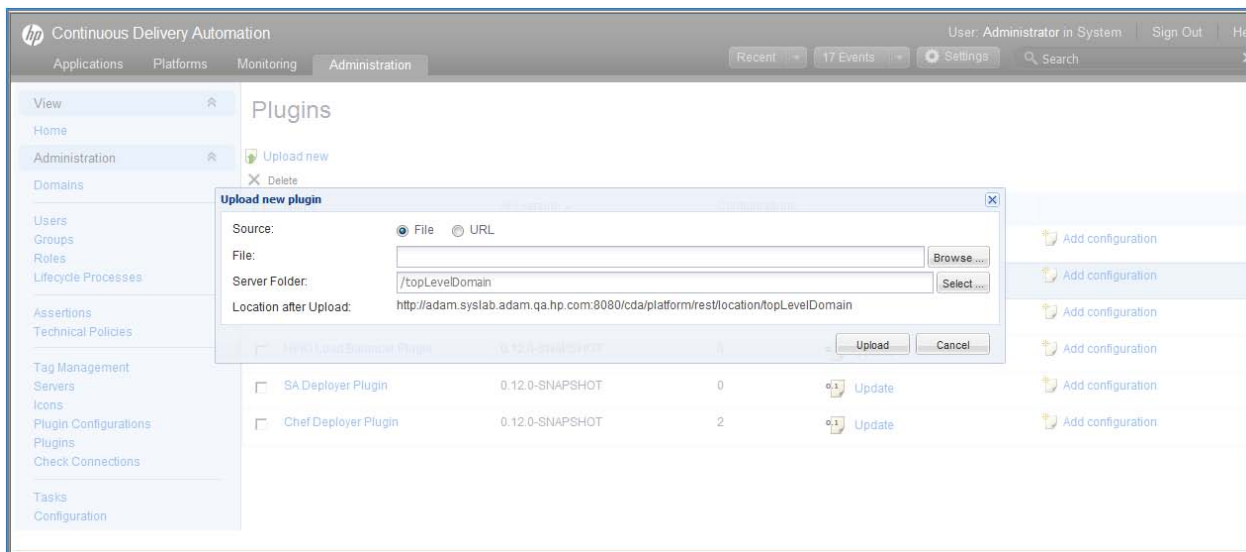


5. Select either **File** or **URL**.
6. Browse and select the File and Server Folder for the Plugin.
7. Click **Update**.

Uploading New Plugins

To upload a new Plugin:

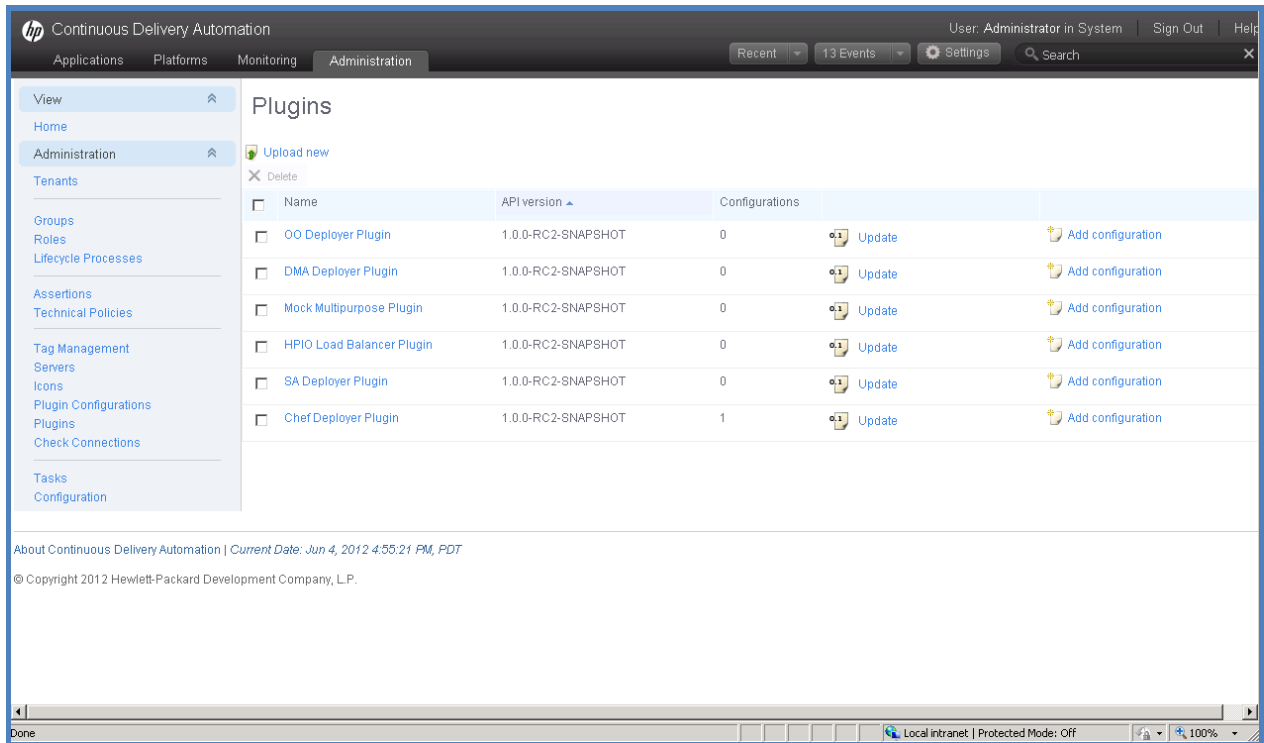
1. Go to the Administration Tab.
2. Click **Plugins**.
3. Click **Upload new**.
4. Select either **File** or **URL**.
5. Browse and select the File and Server Folder for the Plugin.
6. Click **Upload**.



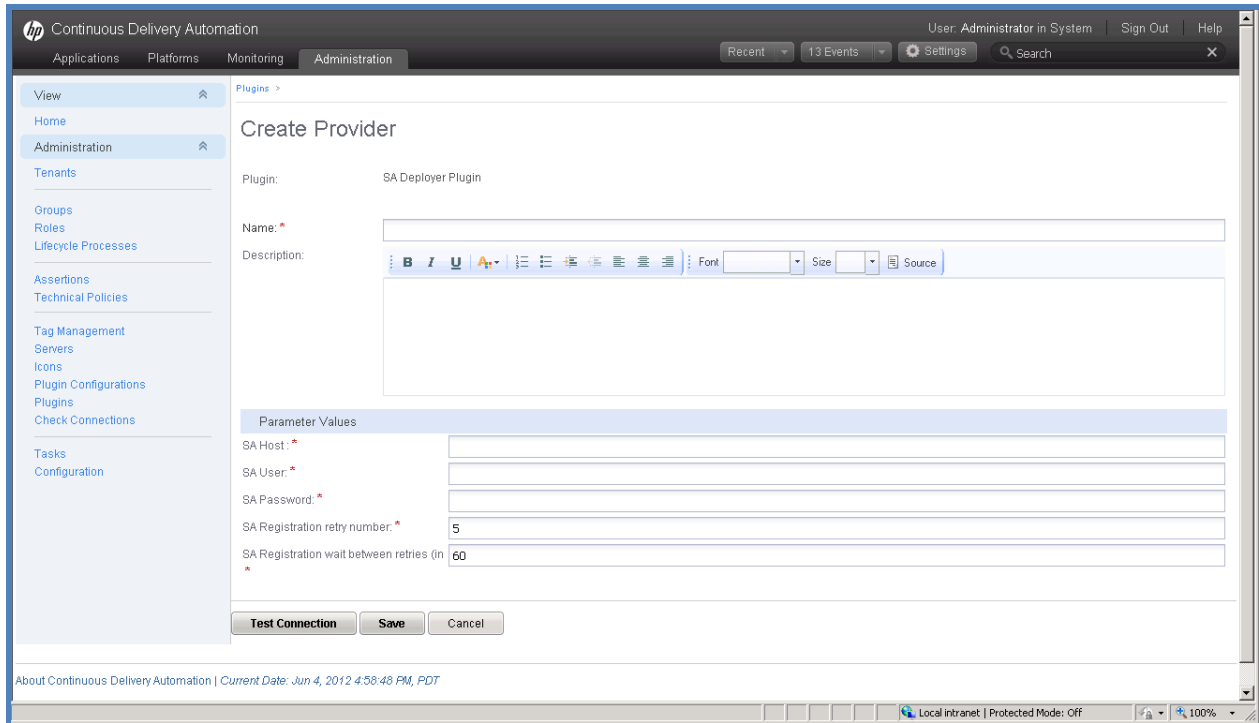
Defining a Deployer Plugin

To define a Deployer Plugin:

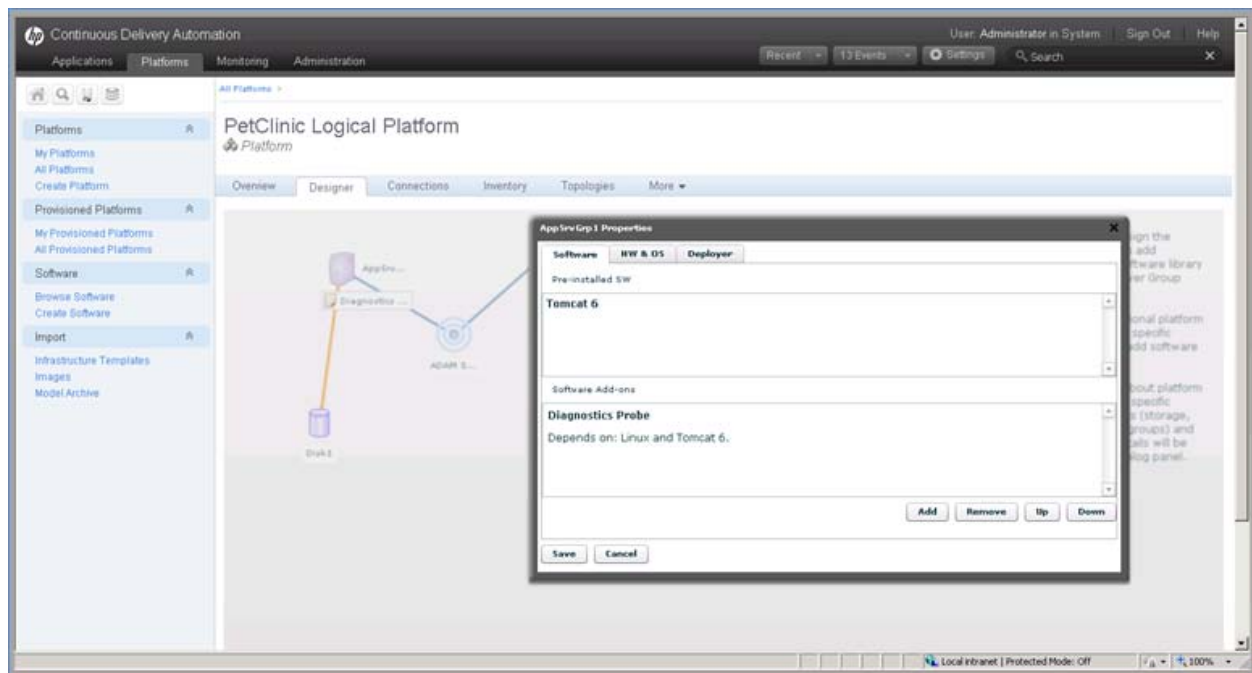
1. Go to the Administration Tab.
2. Click **Plugins** in the left panel.
3. Select the desired Plugin from the predefined list "OO, DMA, HPIO Load Balancer, SA, Chef or Mock Multipurpose".



4. Click **Add configuration** for that Plugin.



5. Add the Name and Connection Parameters in order to access the Server.
6. Click **Test Connection** to make sure that the parameters are entered correctly.
7. Click **Save**. The Server will now appear in the list of available Plugin Configurations.
8. In Platform Designer, for each serverGrp on Properties Screen this plugin will appear in the drop-down list under the Deployer tab.



9. Select the desired deployer from the deployer list in the Platform Designer.

Lifecycle Process Management

Before a Lifecycle Process can be used, it must be first defined and published by a user with Administrator rights.

A Lifecycle Process can also be copied and modified so as to maintain certain values rather than having to be constructed from the beginning.

Lifecycle Processes are defined by clicking **Lifecycle Processes** under the Administration tab and either selecting an existing process for modification or clicking **Create** to build a new one.

In the Lifecycle Process Overview or Stages tab it is possible to edit, publish, clone, delete or export the current process by clicking on the appropriate action.

The following tasks may be required when defining a new, or editing an existing Lifecycle Process:

- Creating Lifecycle Processes
- Defining Lifecycle Stages
- Defining Lifecycle Tasks"
- Define Lifecycle Transitions"
- Defining Lifecycle Approvers
- Defining Lifecycle Policies
- Defining Lifecycle Permissions
- Defining Lifecycle Automatic Actions
- Publishing a Lifecycle Process
- Exporting Lifecycle Processes

Creating Lifecycle Processes

The first step in artifact lifecycle management is the creation of a lifecycle process.

To create a new lifecycle process, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Lifecycle Processes** to open the Lifecycle Processes page, and then click **Create**.
3. In the upper section, enter a **Name** and **Description** for the lifecycle process.
4. Define the lifecycle stages for the lifecycle process as follows:
5. Select the Root Artifact Type from the drop down list.
6. Optionally select the Sub-Artifact Type(s) by selecting the check box of the required types
7. If the new process is to be automatically assigned to artifacts, click the check box and add categories for **Categorized as** and **Not Categorized as** to apply taxonomic association rules for the automatic assignment of the lifecycle process to artifacts based on their categorizations
8. **In Categories:** Artifacts must contain these categories to be automatically assigned to the lifecycle process.
9. **Not In Categories:** Artifacts must not contain these categories to be automatically assigned to the lifecycle process.

10. **In Domains:** Select which domains lifecycle process will be used in.
11. **Note:** In cases where two or more automatically assigned lifecycle processes apply for an artifact type, HP CDA uses these association rules to assign the most appropriate lifecycle process. See "Automatic Assignment" in the Concepts Guide for more information about Automatic Assignment.
12. Click **Save**.

The status of a newly created lifecycle process is set to **Draft** and the user is redirected to the Overview tab of the process page in which lifecycle stages can be defined.

Defining Lifecycle Stages

Lifecycle stages represent important milestones in the lifecycle process. Lifecycle stages and their order are defined in the lifecycle process definition. New lifecycle processes have no stages assigned to them.

To define lifecycle stages, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, click **Lifecycle Processes** to open the Lifecycle Processes page, and then click the name of the required process.

The Lifecycle Processes page opens in the **Overview** tab.

3. To set an initial stage definition for artifacts click **Add Stage**. The Add Initial Stage dialog opens.
4. Select Stage name from dropdown list or the user may enter a custom name.
5. Click **Share Artifacts** and select when the artifact will be shared if required (this can also be altered in the **Permissions** tab). The *After Approval* option is not available if the transition to the next stage is automatic.
6. Click **OK**.
7. (Optional) Additional stages can be added by clicking **Add Stage**, selecting a Stage name and a transition type.

Any process can be edited, cloned, exported, deprecated or deleted by clicking the appropriate button

It is also possible to add the same stage into a process multiple times. These are called referral stages and are visually different from regular stages. Attempting to edit a referral stage will automatically send the user back the properties of the initial version of that stage.

8. Before publishing the process, go to the Stages tab and do the following:

Note: A valid stage definition must contain at least one of the following conditions.

- Set transitions: For details, see [Defining Lifecycle Transitions](#) on page 159.
- Set voters for approval: For details, see [Defining Lifecycle Approvers](#) on page 160.
- Define tasks to complete before stage approval: For details, see [Defining Lifecycle Tasks](#) on page 159.
- Set policies to comply with before stage approval: For details, see [Defining Lifecycle Policies](#) on page 160.
- Define automatic actions to execute when a lifecycle transition occurs: For details, see [Defining Lifecycle Automatic Actions](#) on page 161.

- Specify whether to automatically promote on stage approval: For details, see [Defining Lifecycle Permissions](#) on page 161.
9. Click **Save** to save the Stage Definition and return to the Lifecycle Process page.
All stage details are listed. You can edit or copy a stage definition using **Edit** or **Copy**. Stages can also be deleted by clicking the [Delete](#) link.
 10. When the process is complete, click **Publish** to make active.
Note: Stage layout cannot be changed once the process is published.

Defining Lifecycle Transitions

When you add a stage, you must also add a transition type. Choose from one of three following options:

- **Automatically, When stage_name is Approved**
Stage will transition to the next stage as soon as approved by designated approvers.
- **Manually, After stage_name is Approved**
Stage will require approval from all designated approvers before it can be manually advanced to the next stage.
- **Manually, Anytime**
Stage does not require approval from all designated users before being manually advanced
Hovering over the stage or the transition between stages and clicking the pencil icon allows you to edit a transition.

Caution If a stage has several next stages, only one of the transitions may be automatic and the rest must be set as **Manually, Anytime**. In case of conflict the user will be notified about the automatic change.

Note: If a stage shares several common previous stages, the transitions types are presented together when editing the stage.

Defining Lifecycle Tasks

You can define a set of manual tasks to be performed as part of a lifecycle stage.

To add a task, complete the following steps:

1. Click the **Stages** tab to enter the **Stage Definition** window and click on the stage that you would like to add a task to.
2. Click **Add Task** to open the Add Task dialog box.
3. Enter a name and optional description for the task.
4. A Task may be assigned to a Role, a User or a Group by clicking the **Select** next to **Assigned To:** label. Clicking the **Browse Addressbook** button will reveal lists of Roles, Users and Groups that can have assigned tasks.
5. A Task may be verified by policy by clicking the **Select** next to **Verified by Policy** label which will provide a list of existing policies that may be applied.

Note: Only one policy may be associated with a task at a time. Only policies not used within the same stage can be selected (either in tasks or policies).

6. Click **Ok**.

The new task is added to the Tasks field.

Defining Lifecycle Policies

You can define a set of policies to validate artifact compliance as part of a lifecycle stage.

To add a policy:

1. Click **Stages** to open the **Stage Definition** window and click on the stage that requires a policy.
2. Click **Add Policy** to open the **Add Policy** dialog.
3. You may use either the search **function** or select a policy from the presented list using the radio button.

Note: Only policies not used within the same stage can be selected (either in tasks or policies).

3. Click **Select**.

The next window allows you to change your choice of policy, the artifact type it is applicable to and whether the policy is required to approve the stage.

4. Click **Ok** to apply the policy.

Defining Lifecycle Approvers

For each lifecycle stage in a lifecycle process you can define a unique set of approvers. The approvers determine whether the artifacts governed by the process are complete for the current stage.

To define voters for approval:

1. Click the **Stages** tab on the **Lifecycle Processes** page and then select the stage you require an approval
2. To add an individual, group or role of voter(s):
 - a. Click the **Browse Addressbook** button.
 - b. Principals can be found in the lists or can be found by entering a name in the Search field and clicking **Search** to show matching search results.
 - c. Click the radio button next to the Role/User/Group required and click **Select** to add selection as an approver.
 - d. If the selected approver is not an individual, but a role or group, select the number of votes required by that role or group.
 - e. The **Passive Approval** option can now be selected with a specified number of days before automatic approval.
 - f. Click **OK** to add the Approver
3. To remove voters, click on the **Delete** button next to their name
4. To edit the Passive Approval option, click the **Edit** button next to the name of the approver.

Note Approvers can also be viewed, edited or added under the **Permissions** tab

Defining Lifecycle Automatic Actions

To define automatic actions, click on the [Add Automatic Actions](#) link at the bottom of the stage details window on the stages tab.

The following actions may be available dependant on artifact type:

- **Parent Request Approval** - Creates approval request for parent artifact. Parent Artifact Type and Parent Stage must be selected in the pop-up dialog. Click **OK** to add automatic action. The approval on parent's artifact will be requested automatically.
- **Enable Consumption** - Allows the creation of new contracts.
- **Disable Consumption** - Prevent the creation of new contracts.
- **Remove Comments** - Once artifact goes into production, all comments will be removed.
- **Delete from Registries** - Will perform delete operation from registry with the possibility of setting when the action is applied. Possible values are Entering stage, Leaving stage and Approval succeeded.
- **Export to Registries** - Will perform export function to registry with the possibility of setting when the action is applied. Possible values are Entering stage, Leaving stage and Approval succeeded.
- **Notify** - HP CDA enables you to send e-mail notifications to artifact stakeholders. You can send set up automatic notifications to be sent as a result of lifecycle changes. Notifications require a subject, and may include Lifecycle Status and an attached text.

The possible recipients of the notification are described in the following table:

Notification Recipients

Recipient	Description
Owner	The user, group, or role that owns the artifact.
Maintainers	Users, groups, and roles groups with write permission for the artifact.
Contacts	Users and organizational units associated with the artifact by the contact relationship or listed in the artifact stakeholder property.

Automatic actions can be deleted by clicking the **Delete** icon and the Notifications and Parent Request Approval actions can be edited by clicking the **Edit** icon.

Defining Lifecycle Permissions

For each lifecycle stage in a lifecycle process you can define a set of permissions. These permissions determine which Role, Group or User has ownership, access rights, approver status and whether artifacts are shared or not.

Note: Permission settings are not carried forward to the next stage.

To define a lifecycle permission, complete the following steps:

1. To set permissions for a stage, select the **Permissions** tab in the **Lifecycle Process** window.
2. Click the select box for any stage that requires a change in **Permissions** setting.
 - Read or Write Access can be assigned to individual Users, Roles or Groups by clicking on the Add Read Access or Add Write Access buttons or the corresponding value in the permissions matrix.

- The owner of the Stage can be assigned by clicking the **Set Owner** button or the **Ownership** value in the matrix.
- Artifacts can be shared by selecting a value from the **Share Artifacts** drop-down list.
- Permissions for a stage can be removed by clicking the **Clear** button.
- Defined Stage Permissions can be copied to another stage by clicking the **Copy** button and selecting which stages and/or roles require the copies values.

Publishing a Lifecycle Process

After creating a lifecycle process, the next step is to make it available for the governance of artifacts.

To publish a lifecycle process, complete the following steps:

1. From the Lifecycle Processes page, select the processes to be published by clicking on the check box next to the process name and click **Publish**.
2. Processes can also be published from the Overview or Stages tab of a selected process.
3. Processes can be deprecated by clicking the **Deprecate** button in either the processes page or the Overview or Stages tab of a selected process.

This lifecycle process is now available to be used in the governance of root and sub-artifacts.

Exporting a Lifecycle Processes

Any Lifecycle Process can be exported from either the Overview or Stages tab.

To export a lifecycle process, complete the following steps:

1. Click **Export** from the right hand menu.
2. Provide a name for the exported file and click **Export** in the dialog window.
3. Either open or provide a location for the exported file.

Lifecycle Best Practices

The following list enumerates basic best practices.

- **Govern all artifacts:** HP recommends governing artifacts whenever possible. We are aware that in some situations (e.g. for system artifacts) there may be additional overhead, but the benefits gained are worth it. For example, versioning works properly only for governed artifacts.
- **Assign lifecycle processes automatically:** We recommend marking lifecycle processes as automatically assignable. In our opinion, common users should not select a lifecycle process for their artifact(s). Instead, the Administrator should create an appropriate lifecycle process that is assigned to the artifact automatically after artifact creation.

Action Set Lifecycle Process / Stage provides users with the possibility to set any process to the artifact. Administrators can even set a specific stage and mark it as approved. This action is intended mainly for administrators who import data to the repository. If the data is suitable, Administrators can use this action to set for example, Production stage.

- **Separate lifecycle processes:** HP recommends separating lifecycle processes and defining different lifecycle process for different artifact types. These processes can be connected together via policies but it is crucial to not govern all artifacts in one process.

As an example, you can review the default lifecycle processes include in this topic. There is a special lifecycle process for Project artifacts containing only Project artifacts and their documents. And similarly there are separated lifecycle processes for Applications, Services, and Processes.

Configuration Management

HP CDA provides a customizable configuration that enables the administrator to control nearly every aspect of the behavior of HP CDA.

Administrators can modify most parts of the configuration of HP CDA from the Administration tab.

To manage configuration settings, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, select **Configuration** to open the Configuration page.
3. The Configuration page is split into the following tabs:
 - **Basic Settings:** Settings that affect the operation of HP CDA repository.
 - **System Settings:** The detail level settings of the configuration. The administrator can use this tab to modify individual settings at a global, or domain level. This tab also enables the export and import of the configuration as a whole.
 - **License:** A summary page detailing the terms of the current license with an option to enter a new license key and manage licensed users.
 - **Self-Test:** Details the status of HP CDA with an option to disable self-test.
 - **Artifact Validation:** Configure policies to validate artifacts when users create or edit them.
 - **Report Cleanup:** Set the maximum age of reports and events to be retained when the Report Cleaner Task executes. You can set a different age by report and event type. Manage the execution and scheduling of the Report Cleaner Task using the Tasks page.

Managing Basic Configuration Options

HP CDA enables the administrator to control some basic aspects of the configuration from the Administration tab.

To configure basic settings, complete the following steps:

1. Go to the Administration tab.
2. In the Administration menu, select **Configuration** to open the Configuration page in the Basic Settings tab.
3. In the Basic Settings tab, set any of the following options:
4. Full Text Search
5. Select this option to enable full text search in the HP CDA UI.

Note: Full-text search must also be enabled on the database as described in the following sections of the *Installation and Deployment Guide*:

- "Enable Full-Text Search in MSSQL"
- "Enable Full-Text Search in Oracle"

By default, HP CDA appends a % to search terms. To disable this functionality after installation, set the configuration property `shared.db.fulltextsearch.appendpercentage` to `FALSE`.

Create Empty Data Artifacts

Data artifacts are normally associated with data content, such as a WSDL or document, uploaded from an external source.

By default, HP CDA does not offer the option to directly create new data artifacts (for example, documentation or WSDLs) and only creates artifacts of these types when data content is uploaded. For example, a documentation artifact can only be created by uploading a document.

Set the **Create Empty Data Artifacts** option if you want to enable the creation of new data artifacts without requiring the upload of associated data content.

SSL Customization

To customize SSL, complete the following steps:

1. Select the authentication method to apply to SSL certificates.

Certificate Trust and Customization Categories

Certificate Trust	Customization
Java/JSSE default key/trust stores...	default
Server certificates are always trusted...	skipped
Database key/trust stores...	database
Composition of database and default...	composite

2. Click **Save** to make your changes or **Reset to Defaults** to restore these settings to their defaults.

License Management

HP CDA employs a licensing system to control the following aspects of an installation:

- Usage limits, including the following:
 - Time limited licenses
 - Number of Application models limited licenses
- Feature limits, including the following:
 - Integration with Opscode Chef
 - Integration with Nagios

Access the license information from the Administration tab. In the Administration menu, click Configuration to open the Configuration page, and select the License tab to view the current license.

Changing the License

The administrator can change the HP CDA license.

To change the license, complete the following steps:

1. Contact your sales representative for a new license key.
2. In the Administration tab Administration menu, click **Configuration** to open the Configuration page.
3. Select the **License** tab to view the current license details.

4. Click **Enter License Key** to open the Enter License Key page.
5. Click **Browse** to select the license file on the local disk.
6. Click **Finish**.

The Configuration page opens in the License tab showing the details of the new license.

HP CDA Self-Test

HP CDA includes a Self-Test application that runs a series of checks on the installation and configuration.

The administrator can access the Self-Test from the Administration tab. In the Administration menu, select **Configuration** to open the Configuration page, and then select the **Self-Test** tab.

The Self-Test tab displays a summary and then the details of the self-test checks.

To turn Self-Test on or off:

Click the **Disable** / **Enable** button.

The self-tester performs the following checks:

Self-Test	Description
Product configuration checks	Checks product configuration, versions, and libraries.
Product runtime checks	Checks logging configuration, and outputs product base URLs.
Application server checks	Checks application server and JVM settings.
JNDI checks	Checks required JNDI resources.
Datasource checks	Checks the data source connection.
JMS checks	Checks the sending of JMS messages to required JMS destinations.
LDAP checks	Checks LDAP connectivity, if configured during installation or setup.
Performance	Basic HP CDA performance checks.

In the default configuration, the output includes only information about the groups of checks that are run and any errors that occur. The full self-test output is stored in the application server log folder, CDA_self_test.log.

A machine-readable output of the self-test is available at <http://hostname:port/context/self-test/xml>.

Self-Test is also available as a stand-alone deployment which you can use during installation and deployment. For details, see "HP CDA Self-Test" in the *Installation and Deployment Guide*.

Managing System Settings

In the Configuration page System Settings tab, the administrator can add, edit, and remove system settings.

System settings can apply globally, or within a specific domain. If a domain setting exists, it takes precedence over a global setting.

To add system settings, complete the following steps:

1. In the Configuration page System Settings tab, click **Add Property** to open the Add Property dialog box.
2. *(Optional)* Click **Change** to alter the domain that the setting applies to.
3. Set a name and value for the property, and click **OK** to add the property to the system settings for the selected domain.

To add system settings from a file, complete the following steps:

1. In the System Settings tab, click **Add File Property** to open the Add File Property dialog box.
2. *Optional:* Click **Change** to alter the domain that the setting applies to.
3. Set a name for the property, browse for the file on your local file system, and click **OK** to add the property to the system settings for the selected domain.

To edit system settings, complete the following steps:

1. In the System Settings tab, use the Name filter to locate the setting you want to edit.
2. Click **Edit** for the setting to open the Edit Property dialog box.
3. Do one of the following:
 - Input a new value and click **Save** to set a new value.
 - Click **Load Default Value** to reset the value to its default installation value.

To delete system settings, complete the following steps:

1. In the System Settings tab, select the properties you want to delete.
2. Click **Remove** and confirm your decision.

Managing Artifact Form Validation

The Administrator can assign policies to validate artifacts when a user creates or edits them—for example, to ensure that keywords are set when a user creates a service.

To Assign Artifact Validation Policies:

1. In the Administration tab Administration menu, click **Configuration** to open the Configuration page.
2. Select the **Artifact Validation** tab to view the list of artifact types.

Note: Numbers next to artifact type names indicate validation policies in use.

3. Select the artifact type that you want to validate for creation and edit. The Validated By table refreshes to show the policies used to validate the selected artifact type.

Do one of the following:

- To Add Policies:
 - a. Click **Add Policy** to open the Add Policy dialog.
 - b. Use the search criteria to locate the technical policies you want to use.
 - c. Select the policies to use and click **Select** to add the policies to the Validate By table.
- To remove policies, complete the following steps:
 - a. Select the policies to remove.
 - b. Click **Remove Selected** and confirm your decision to remove the selected policies from the Validated By table.

Note: You can extend form validation to also validate data attachments to artifacts. Set property `platform.autovalidation.validateData` to true.

Administration Task Management

HP CDA includes a number of default administration tasks to help administrators to manage their deployment and the data it contains. In addition, the administrator can configure additional tasks to perform change management or custom tasks included in an extension.

To display and manage administration tasks:

1. Go to the Administration tab.

In the Administration menu, click **Tasks** to open the Tasks page.

2. The initial view of the Administrator Task page provides a simplified grid interface for dealing with the administration of Daily, Weekly, and unscheduled tasks. Any running tasks will also be visible with options to either stop them independently or to stop all running tasks.
3. HP CDA provides the following default administration tasks:
 - **Activity Report Update:** Updates the artifact activity data used to generate the Activity Report shown in the Reports tab Homepage.
 - **Lifecycle Passive Approval:** Approves Lifecycle stages that have been set to automatic approval after a set number of days.
 - **Lifecycle Validation:** Performs a validation of all artifacts in governance against the policies that apply to their current lifecycle stage.
 - **Policy Report Validation:** Performs an update of any policy reports.
 - **Recycle Bin Cleaner:** Permanently removes deleted artifacts from the recycle bin as the defined schedule.
 - **Report Cleanup:** Removes old reports and events based on their age against properties set in the Configuration page Report Cleanup tab.
 - **Build Cleaner Task:** Deletes old builds based on the configuration in "Build Cleanup" tab for each Domain.
 - **Data File Cleaner Task:** Runs data file cleanup job which removes file artifacts.
 - **Delegation Updater Task:** Runs Delegation Update job.
 - **Send Delayed Notifications Task:** Sends all expired delayed notifications.

The administrator can execute tasks immediately or schedule them to run on a periodic basis. In addition to the default tasks, the administrator can add change management tasks and custom tasks.

Running Tasks

You may want to immediately execute a task. If you have sufficient permissions, there is a **Run** option in the initial Tasks overview as well as in the detail view of the task.

To manually execute tasks, complete the following steps:

1. Click **Run** in either the Task overview or in the Task detail page.
2. Scheduling Tasks
3. HP CDA enables you to execute tasks on a timed or periodic basis.

Caution: HP CDA converts and stores any input time to GMT. If you import scheduled tasks from a data image, review the scheduling to ensure that the tasks execute at the local time that you require.

To schedule tasks, complete the following steps:

1. In the task window or detail view of the task click **Edit Schedule** to open the Edit Schedule dialog.
2. Select a start date and start time for the task
3. Set the frequency that the task will run
4. Either set the task to run Indefinitely, set a finish date and time or enter a value representing the number of times the task should run
5. Advanced Options allow you to select whether the schedule should be followed during downtime or to wait until HP CDA start running.
6. Click **OK** to set the schedule for the task.

Adding Change Management Tasks

Sync Tasks enable the administrator to synchronize artifacts in the repository with those imported from an external source

To Create Sync Tasks, complete the following steps:

1. Click **Add Artifact Sync Task** to open the dialog.
2. Add a name and optional description for the task.
Artifacts can be selected through using either standard search methods or they can be selected from a filtered list.
3. Click **Save**.
4. Clicking **Edit Schedule** will allow you to set a schedule for the running of the new task or the task can be run immediately by clicking **Run**.
5. The task may be edited by clicking on **Edit**.

Adding Custom Tasks

If HP CDA includes an extension which contains custom task implementations, you can add and schedule them in the Tasks page.

To create custom tasks, complete the following steps:

1. Click **Add Custom Task** to open the dialog.
2. Select a Task Implementation and add a name for the task.
Task Parameters are defined and variable according to how the task component was composed.
3. Click **Save**.
4. (Optional) Clicking **Edit Schedule** will allow you to set a schedule for the running of the new task or the task can be run immediately by clicking **Run**.
5. (Optional) The task can be edited by clicking on **Edit**.

External System Integration Using the Command Line Interface

HP CDA provides the ability to integrate using a Command Line Interface. HP CDA provides a command line interface which enables users to perform activities such as the following:

- List application model related artifacts.
- Invoke jobs related to the following:
 - Platform provisioning and de-provisioning
 - Application deployment and undeployment
 - Application start and stop
 - Job progress
- Create and manage platform to use existing infrastructure available in the environment.

For more information about available commands, see Appendix A, [The HP CDA Command Line Interface](#).

CLI Parameters and Options

The parameters and options which govern the usage of this tool are described below.

Model Commands

Purpose

To list artifacts which are part of an application model

Usage

`cdaexec m list <options>` [The various options and their usage are provided in Appendix A, [The HP CDA Command Line Interface.](#)]

Details

This parameter allows the user to list artifacts which are part of an application model in HP CDA. The results of these commands contain the “Name” and the respective “UUID” (Universally Unique Identified) of the artifact, which can be used with the “Job” parameter covered in the next section.

Job Commands

Purpose:

To invoke specific jobs available at Platform and Application level

Usage:

`cdaexec j <options>` [The various options and their usage are provided in Appendix A, [The HP CDA Command Line Interface.](#)]

Details

This parameter allows the user to invoke jobs related to platform provisioning\de-provisioning, application deployment\un-deployment, application start\stop, check job progress as well as invoking any custom operation defined in the application model.

Existing Infrastructure Commands

Purpose:

To create and manage platform comprising of existing infrastructure available in the environment

Usage

`cdaexec xi <verb> <artifact>` [Commands are provided in Appendix A, [The HP CDA Command Line Interface.](#)]

Supported verbs – add, delete, set, get & list

Supported artifacts – server, serverpool & template

Details:

HP CDA provides a mechanism to use existing infrastructure (servers) available in the environment to be configured as target platform for application deployment.

This unique feature allows the user to:

HP Continuous Delivery Automation User Guide

- create a “server” artifact for every existing server available in the network and ready to be used with HP CDA
- group these servers to be part of “serverpool” artifact(s)
- create Existing Infrastructure Template(s) which is a collection of serverpools

The Existing Infrastructure Templates(s) can be viewed by going to Platforms tab > Infrastructure Templates page. These templates can now be used to create a platform in the same way as the HP IO based templates are used.

A The HP CDA Command Line Interface

HP CDA Command Line Integration Commands Summary

Configuration Commands

configuration encryptproperties

Model Commands

model list application [-d *domain*]

model list applicationversion -a *application*

model list applicationmodel [-a *application*] [-n *name*] [-av *applicationversion*] [-d *domain*]

model list domain [-cwd *currentf*]

model list logicaltopology [-am *applicationmodelf*] [-d *domain*] [-n *name*]

model list logicalplatform [-d *domain*] [-n *name*] [-lt *logicaltopology*]

model list monitoringtool

model list realizedtopologymonitors -rt *realizedtopology* [-mt *monitoringtool*]

model list realizedtopology [-am *applicationmodelf*] [-n *name*] [-lt *logicaltopology*] [-d *domain*]

model list realizedplatform [-d *domain*] [-lt *logicaltopology*] [-lp *logicalplatform*] [-n *name*]

model list storedparameters [-lt *logicaltopology*] [-lp *logicalplatform*] [-rt *realizedtopology*] [-rp *realizedplatform*]

model list deploymentset [-lt *logicaltopology*] [-av *applicationversion*]

Job Commands

`job provision -n name -lp logicalplatform -lpsp logicalplatformstoredparameters [-rb rollbackOnError]`

`job deploy -n name -lt logicaltopology -ltsp logicaltopologystoredparameters [-lpsp logicalplatformstoredparameters] [-rp realizedplatform] [-ds deploymentset] [-rb rollbackOnError]`

`job undeploy -rt realizedtopology`

`job deprovision -rp realizedplatform`

`job checkprogress -jr jobreport`

`job start -rt realizedtopology`

`job stop -rt realizedtopology`

`job customoperation -rt realizedtopology -co customoperation`

Existing Infrastructure Commands

```
existinginfrastructure add server -hn hostname -ip ip [-down] [-d domainId]
```

```
existinginfrastructure set server <Hostname> [-hn hostname] [-ip ip] [-d domainId]
```

```
existinginfrastructure list server [-d domainId]
```

```
existinginfrastructure delete server <Hostname> [-d domainId]
```

```
existinginfrastructure get server <Hostname> [-d domainId]
```

```
existinginfrastructure add serverpool -nm name -os ostype [-min min] [-max max] [-cpu cpucount] [-maxcpu maxcpucount] [-mem memsize] [-maxmem maxmemsize] [-memType memsizeType] [-svr attachserver] [-d domainId]
```

```
existinginfrastructure set serverpool <Name> [-nm name] [-min min] [-max max] [-os ostype] [-cpu cpucount] [-maxcpu maxcpucount] [-mem memsize] [-maxmem maxmemsize] [-memType memsizeType] [-svr attachserver] [-d domainId]
```

```
existinginfrastructure list serverpool [-d domainId]
```

```
existinginfrastructure delete serverpool <Name> [-d domainId]
```

```
existinginfrastructure get serverpool <Name> [-d domainId]
```

```
existinginfrastructure add template -nm name [-atchsvrpl attachserverpool] [-d domainId]
```

```
existinginfrastructure set template <Name> [-nm name] [-atchsvrpl attachserverpool] [-d domainId]
```

```
existinginfrastructure list template [-d domainId]
```

```
existinginfrastructure delete template <Name> [-d domainId]
```

```
existinginfrastructure get template <Name> [-d domainId]
```

NAME configuration encryptproperties

ALIAS config encryptproperties

SYNOPSIS configuration encryptproperties

DESCRIPTION Encrypts the password in the properties file.

OPTIONS	<p>-s, --server</p> <p>-u, --user</p> <p>-p, --password</p> <p>-c, --config</p>	<p>The CDA server (url).*</p> <p>The CDA username.*</p> <p>The password for the CDA user.*</p> <p>A file containing default general option values in the format "<code><name>=<value></code>", where <code><name></code> is "server", "user", or "password" .</p> <p>If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*</p>
----------------	---	---

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME model list application

ALIAS m list app

SYNOPSIS model list application [-d *domain*]

DESCRIPTION Lists application(s). Applications can be listed by domain. If domain is not specified, all applications are returned.

OPTIONS	<p>-d, --domain</p> <p>-s, --server</p> <p>-u, --user</p> <p>-p, --password</p> <p>-c, --config</p>	<p>A domain UUID. If provided, all applications related to the specified to domain will be listed.</p> <p>The CDA server (url).*</p> <p>The CDA username.*</p> <p>The password for the CDA user.*</p> <p>A file containing default general option values in the format "<code><name>=<value></code>", where <code><name></code> is "server", "user", or "password".</p> <p>If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*</p>
----------------	---	---

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME model list applicationversion

ALIAS m list av

SYNOPSIS model list applicationversion -a *application*

DESCRIPTION Lists application version(s) by application UUID.

OPTIONS	<p>-a, --application</p> <p>-s, --server</p> <p>-u, --user</p> <p>-p, --password</p> <p>-c, --config</p>	<p>An application UUID</p> <p>The CDA server (url).*</p> <p>The CDA username.*</p> <p>The password for the CDA user.*</p> <p>A file containing default general option values in the format "<code><name>=<value></code>", where <code><name></code> is "server", "user", or "password".</p> <p>If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*</p>
----------------	--	---

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	model list applicationmodel	
ALIAS	m list am	
SYNOPSIS	model list applicationmodel [-a <i>application</i>] [-n <i>name</i>] [-av <i>applicationversion</i>] [-d <i>domain</i>]	
DESCRIPTION	Lists application model(s). Application models can be listed by either name, application id, application version id or domain. If no argument is specified, all application models are returned. No more than one argument can be supplied to this operation.	
OPTIONS	-a, --application	An application UUID. If provided, only the application models for this application will be listed.
	-n, --name	An application model name. If more than one application model shares the same name then the UUID for each application model of that name will be returned.
	-av, --applicationversion	An application version UUID. If provided only the application models for this application version will be listed.
	-d, --domain	A domain UUID. If provided, all applications related to the specified to domain will be listed.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <i><name>=<value></i> ", where <i><name></i> is "server", "user", or "password". If not specified, the <i>cliforcda.properties</i> file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	model list domain	
ALIAS	m list domain	
SYNOPSIS	model list domain [-cwd <i>current</i>]	
DESCRIPTION	Lists domains. If "--current" is specified will only list the current working domain for the user defined in the "--user" general parameter.	
OPTIONS	-cwd, --current	If specified, return only the working domain of the current user.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME model list logicaltopology

ALIAS m list lt

SYNOPSIS model list logicaltopology [-am *applicationmodel*] [-d *domain*] [-n *name*]

DESCRIPTION Lists logical topologies. If an application model UUID is supplied only the logical topologies attached to that application model will be listed. Select only one option from: -am, -d, or -n.

OPTIONS	-am, --applicationmodel	An application model UUID. If specified, only logical topologies attached to this application model will be listed.
	-d, --domain	A domain UUID. If specified, only the logical topologies related to this domain will be listed.
	-n, --name	A logical topology name. If more than one logical topology shares the same name then the UUID for each logical topology of that name will be returned.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME model list logicalplatform

ALIAS m list lp

SYNOPSIS model list logicalplatform [-d *domain*] [-n *name*] [-lt *logicaltopology*]

DESCRIPTION Lists all logical platforms. Select only one option from: -d, -n, or -lt.

OPTIONS	-d, --domain	A domain UUID. If specified, only the logical platforms related to this domain will be listed.
	-n, --name	A logical platform name. If more than one logical platform shares the same name then the UUID for each logical platform of that name will be returned.
	-lt, --logicaltopology	A logical topology UUID. If specified, only logical platforms related to this logical topology will be listed.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password". If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME model list monitoringtool

ALIAS m list mt

SYNOPSIS model list monitoringtool

DESCRIPTION Lists available monitoring tools.

OPTIONS

-s, --server	The CDA server (url).*
-u, --user	The CDA username.*
-p, --password	The password for the CDA user.*
-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".

If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	model list realizedtopologymonitors	
ALIAS	m list rtmn	
SYNOPSIS	model list realizedtopologymonitors -rt realizedtopology [-mt monitoringtool]	
DESCRIPTION	Lists monitors deployed in the given realizedtopology. If monitoringtool is specified, results are filtered to only include monitors deployed for given monitoring tool.	
OPTIONS	-rt, --realizedtopology	A realizedtopology UUID. Identifies the realizedtopology to search.
	-mt, --monitoringtool	A monitoringtool UUID. If provided, the result will be filtered to only include monitors for the given tool.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password". If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	model list realizedtopology	
ALIAS	m list rt	
SYNOPSIS	model list realizedtopology [-am <i>applicationmodel</i>] [-n <i>name</i>] [-lt <i>logicaltopology</i>] [-d <i>domain</i>]	
DESCRIPTION	Lists all realized topologies. Select only one option from: -am, -n, -lt, or -d.	
OPTIONS	-am, --applicationmodel	An application model UUID. If provided, only the realized topologies for this application model will be listed.
	-n, --name	A realized topology name. If more than one realized topology shares the same name then the UUID for each realized topology of that name will be returned.
	-lt, --logicaltopology	A logical topology UUID. If provided only the realized topologies for this logical topology will be listed.
	-d, --domain	A domain UUID. If provided, all realized topologies related to the specified to domain will be listed.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <i><name>=<value></i> ", where <i><name></i> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	model list realizedplatform	
ALIAS	m list rp	
SYNOPSIS	model list realizedplatform [-d <i>domain</i>] [-lt <i>logicaltopology</i>] [-lp <i>logicalplatform</i>] [-n <i>name</i>]	
DESCRIPTION	Lists all realized platforms. Select only one option from: -d, -lt, -lp, or -n.	
OPTIONS	-d, --domain	A domain UUID. If specified, only the realized platforms related to this domain will be listed.
	-lt, --logicaltopology	A logical topology UUID. If specified, only realized platforms related to this logical topology will be listed.
	-lp, --logicalplatform	A logical platform UUID. If specified, only realized platforms related to this logical platform will be listed.
	-n, --name	A realized platform name. If more than one realized platform shares the same name then the UUID for each realized platform of that name will be returned.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <i><name>=<value></i> ", where <i><name></i> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	model list storedparameters	
ALIAS	m list storedparams m list sp	
SYNOPSIS	model list storedparameters [-lt <i>logicaltopology</i>] [-lp <i>logicalplatform</i>] [-rt <i>realizedtopology</i>] [-rp <i>realizedplatform</i>]	
DESCRIPTION	Lists the stored parameters for a given model object specified by its UUID. Only one model object UUID may be specified.	
OPTIONS	<ul style="list-style-type: none"> -lt, --logicaltopology -lp, --logicalplatform -rt, --realizedtopology -rp, --realizedplatform -s, --server -u, --user -p, --password -c, --config 	<p>A logical topology UUID.</p> <p>A logical platform UUID.</p> <p>A realized topology UUID.</p> <p>A realized platform UUID.</p> <p>The CDA server (url).*</p> <p>The CDA username.*</p> <p>The password for the CDA user.*</p> <p>A file containing default general option values in the format "<code><name>=<value></code>", where <code><name></code> is "server", "user", or "password".</p> <p>If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*</p>

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME model list deploymentset

ALIAS m list ds

SYNOPSIS model list deploymentset [-lt *logicaltopology*] [-av *applicationversion*]

DESCRIPTION Lists deployment sets for a given logical topology or application version. Only one model object UUID may be specified.

OPTIONS

-lt, --logicaltopology	A logical topology UUID.
-av, --applicationversion	An application version UUID.
-s, --server	The CDA server (url).*
-u, --user	The CDA username.*
-p, --password	The password for the CDA user.*
-c, --config	A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password".

If not specified, the `cliforcda.properties` file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	job provision	
ALIAS	j provision	
SYNOPSIS	job provision -n name -lp <i>logicalplatform</i> -lpsp <i>logicalplatformstoredparameters</i> [-rb <i>rollbackOnError</i>]	
DESCRIPTION	Provisions a realized platform from a logical platform and logical platform stored parameters. If the deploy request is submitted successfully, a job report ID is returned. The status of the request can be checked by using this ID in a 'job checkprogress' operation.	
OPTIONS	-n, --name	A name from which the name of the realized platform will be derived.
	-lp, --logicalplatform	A logical platform UUID from which the realized platform will be provisioned.
	-lpsp, --logicalplatformstoredparameters	A logical platform stored parameters UUID.
	-rb, --rollbackOnError	Rollback the operation when an error occurs. Defaults to true.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password". If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME job deploy

ALIAS j deploy

SYNOPSIS job deploy -n *name* -lt *logicaltopology* -ltsp *logicaltopologystoredparameters*
 [-lpsp *logicalplatformstoredparameters*] [-rp *realizedplatform*]
 [-ds *deploymentset*] [-rb *rollbackOnError*]

DESCRIPTION Deploys a logical topology to an already realized platform or provisions a realized platform from a logical platform stored configuration and deploys a logical topology to the newly realized platform. If the deploy request is submitted successfully, a job report ID is returned. The status of the request can be checked by using this ID in a 'job checkprogress' operation.

OPTIONS	-n, --name	A name from which the names of the realized topology and realized platform will be derived.
	-lt, --logicaltopology	A logical topology UUID.
	-ltsp, --logicaltopologystoredparameters	A logical topology stored parameters UUID.
	-lpsp, --logicalplatformstoredparameters	A logical platform stored parameters UUID. If provided, the "--realizedplatform" option cannot be also provided.
	-rp, --realizedplatform	A realized platform UUID. If provided, the "--logicalplatformstoredconfiguration" argument must not also be provided.
	-ds, --deploymentset	A deployment set UUID.
	-rb, --rollbackOnError	Rollback the operation when an error occurs. Defaults to true.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password" . If not specified, the cliforcda.properties file in the current user's home directory is used if it

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

exists.*

NAME job undeploy

ALIAS j undeploy

SYNOPSIS job undeploy -rt *realizedtopology*

DESCRIPTION Undeploy a realized topology. If the undeploy request is submitted successfully, a job report ID is returned. The status of the request can be checked by using this ID in a 'job checkprogress' operation.

OPTIONS

-rt, --realizedtopology	A realized topology UUID.
-s, --server	The CDA server (url).*
-u, --user	The CDA username.*
-p, --password	The password for the CDA user.*
-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".

If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME job deprovision

ALIAS j deprovision

SYNOPSIS job deprovision -rp *realizedplatform*

DESCRIPTION Deprovisions a specified realized platform. If the deprovision request is submitted successfully, a job report ID is returned. The status of the request can be checked by using this ID in a 'job checkprogress' operation.

OPTIONS

-rp, --realizedplatform	A realized platform UUID.
-s, --server	The CDA server (url).*
-u, --user	The CDA username.*
-p, --password	The password for the CDA user.*
-c, --config	A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password".

If not specified, the `cliforcda.properties` file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME job checkprogress

ALIAS j checkprogress

SYNOPSIS job checkprogress -jr *jobreport*

DESCRIPTION Displays the progress information for a specified CDA job report.

OPTIONS	-jr, -jobreport	A job report UUID.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password".
		If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	job start	
ALIAS	j start	
SYNOPSIS	job start -rt <i>realizedtopology</i>	
DESCRIPTION	Executes the start operation on the specified realized topology. If the job start request is submitted successfully, a job report ID is returned. The status of the request can be checked by using this job report ID in a 'job checkprogress' operation.	
OPTIONS	-rt, --realizedtopology	A realized topology UUID.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password".
		If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	job stop	
ALIAS	j stop	
SYNOPSIS	job stop -rt <i>realizedtopology</i>	
DESCRIPTION	Executes the stop operation on the specified realized topology. If the job stop request is submitted successfully, a job report ID is returned. The status of the request can be checked by using this job report ID in a 'job checkprogress' operation.	
OPTIONS	-rt, --realizedtopology	A realized topology UUID.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password".
		If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME job customoperation

ALIAS j customoperation

SYNOPSIS job customoperation -rt *realizedtopology* -co *customoperation*

DESCRIPTION Executes the named custom operation on the specified realized topology. If the job customoperation request is submitted successfully, a job report ID is returned. The status of the request can be checked by using this job report ID in a 'job checkprogress' operation.

OPTIONS	-rt, --realizedtopology	A realized topology UUID.
	-co, --customoperation	A custom operation name.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	existinginfrastructure add server	
ALIAS	xinfra add svr xi add svr	
SYNOPSIS	existinginfrastructure add server -hn <i>hostname</i> -ip <i>ip</i> [-down] [-d <i>domainId</i>]	
DESCRIPTION	<p>Adds a server.</p> <p>Specify “-down” to indicate a server is DOWN temporarily.</p> <p>Server is assumed to be in UP state if “-down” is not specified.</p> <p>Specify “-domainId” to indicate which domain the specified server should be added to.</p>	
OPTIONS	<p>-hn, --hostname</p> <p>-ip, --ip</p> <p>-down, --inActive</p> <p>-d, --domainID</p> <p>-s, --server</p> <p>-u, --user</p> <p>-p, --password</p> <p>-c, --config</p>	<p>A Server hostname.</p> <p>A Server IP address.</p> <p>To indicate that server is temporarily down and cannot be used.</p> <p>Domain Identifier</p> <p>The CDA server (url).*</p> <p>The CDA username.*</p> <p>The password for the CDA user.*</p> <p>A file containing default general option values in the format “<name>=<value>”, where <name> is “server”, “user”, or “password”.</p> <p>If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*</p>

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure set server

ALIAS xinfra set svr

xi set svr

SYNOPSIS existinginfrastructure set server <Hostname> [-hn *hostname*] [-ip *ip*]
[-d *domainId*]

DESCRIPTION Edits server information.

Specify “-hostname” to change the hostname of server.

Specify “-down” to indicate a server is DOWN temporarily.

Specify “-up” to indicate a server is UP and ready to use.

OPTIONS

Hostname	A Server hostname.
-hn, --hostname	A Server new hostname.
-ip, --ip	A Server IP address.
-down, --inactive	To indicate that server is temporarily down and cannot be used.
-up, --active	To indicate that server is up and ready to use.
-d, --domainId	Domain Identifier.
-s, --server	The CDA server (url).*
-u, --user	The CDA username.*
-p, --password	The password for the CDA user.*
-c, --config	A file containing default general option values in the format “<name>=<value>”, where <name> is “server”, “user”, or “password”.
	If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure list server

ALIAS xinfra list svr

xi list svr

SYNOPSIS existinginfrastructure list server [-d *domainId*]

DESCRIPTION Lists all available servers.

OPTIONS	<p>-d, --domainId</p> <p>-s, --server</p> <p>-u, --user</p> <p>-p, --password</p> <p>-c, --config</p>	<p>Domain Identifier.</p> <p>The CDA server (url).*</p> <p>The CDA username.*</p> <p>The password for the CDA user.*</p> <p>A file containing default general option values in the format "<code><name>=<value></code>", where <code><name></code> is "server", "user", or "password".</p> <p>If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*</p>
----------------	---	--

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure delete server

ALIAS xinfra delete srvr
xi delete srvr

SYNOPSIS existinginfrastructure delete server <Hostname> [-d domainId]

DESCRIPTION Deletes a server. Specify server hostname.

OPTIONS	<p>Hostname</p> <p>-d, --domainId</p> <p>-s, --server</p> <p>-u, --user</p> <p>-p, --password</p> <p>-c, --config</p>	<p>A Server hostname.</p> <p>Domain Identifier.</p> <p>The CDA server (url).*</p> <p>The CDA username.*</p> <p>The password for the CDA user.*</p> <p>A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".</p> <p>If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*</p>
----------------	---	---

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure get server

ALIAS xinfra get srvr
xi get srvr

SYNOPSIS existinginfrastructure get server <Hostname> [-d domainId]

DESCRIPTION Fetches server information.

OPTIONS	Hostname -d, --domainId -s, --server -u, --user -p, --password -c, --config	A Server hostname. Domain Identifier. The CDA server (url).* The CDA username.* The password for the CDA user.* A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password". If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*
----------------	--	---

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure add serverpool

ALIAS xinfra add svrpool
xi add svrpool

SYNOPSIS existinginfrastructure add serverpool -nm *name* -os *ostype* [-min *min*]
[-max *max*] [-cpu *cpucount*] [-maxcpu *maxcpucount*] [-mem *memsize*]
[-maxmem *maxmemsize*] [-memType *memsizeType*] [-srvr *attachserver*] [-d *domainId*]

DESCRIPTION Adds a server pool.

Specify type of operating system (os) of servers to be allocated to the server pool. Specify "Windows", "Linux" or "Unknown", if the os is undetermined at this stage.

Remember to set operating system using the set serverpool command before using this serverpool for provisioning. Expect failures if this is not done!

To attach more than one server to server pool, specify "--attachserver" multiple times.

Specify "--domainId" to indicate which domain the specified serverpool should be added to.

OPTIONS	-nm, --name	A Server Pool name.
	-os, --ostype	Common Operating System of Servers in Server Pool.
	-min, --min	Minimum server count.
	-max, --max	Maximum server count.
	-cpu, --cpucount	Common Processor Count of Servers in Server Pool.
	-maxcpu, --maxcpucount	Common Maximum Processor Count of Servers in Server Pool.
	-mem, --memsize	Memory of Servers in Server Pool.
	-maxmem, --maxmemsize	Maximum Memory of Servers in Server Pool.
	-memType, --memsizeType	Unit of Memory of Servers in Server Pool.
	-srvr, --attachserver	A server hostname.
	-d, --domainId	Domain Identifier

HP Continuous Delivery Automation User Guide

-s, --server	The CDA server (url).*
-u, --user	The CDA username.*
-p, --password	The password for the CDA user.*
-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password". If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure set serverpool

ALIAS xinfra set svrpool
xi set svrpool

SYNOPSIS existinginfrastructure set serverpool <Name> [-nm name] [-min min]
[-max max] [-os ostype] [-cpu cpucount] [-maxcpu maxcpucount]
[-mem memsize] [-maxmem maxmemsize] [-memType memsizeType]
[-svr attachserver] [-d domainId]

DESCRIPTION Edits server pool information. Specify server pool name.

Specify "--name" to change the name of server pool.

Specify "--ostype" to change the operating system type of the server pool. Specify "Windows", "Linux" or "Unknown", if the os is undetermined at this stage.

Remember to set operating system using the set serverpool command before using this serverpool for provisioning. Expect failures if this is not done!

Specify "--attachserver" to attach servers to serverpool. To attach more than one server to server pool, specify "--attachserver" multiple times.

OPTIONS	Name	A Serverpool name.
	-nm, --name	A Server Pool new name.
	-os, --ostype	Common Operating System of Servers in Server Pool.
	-min, --min	Minimum server count.
	-max, --max	Maximum server count.
	-cpu, --cpucount	Common Processor Count of Servers in Server Pool.
	-maxcpu, --maxcpucount	Common Maximum Processor Count of Servers in Server Pool.
	-mem, --memsize	Memory of Servers in Server Pool.
	-maxmem, --maxmemsize	Maximum Memory of Servers in Server Pool.

HP Continuous Delivery Automation User Guide

-memType, -memsizeType	Unit of Memory of Servers in Server Pool.
-srvr, --attachserver	A server hostname.
-d, --domainId	Domain Identifier
-s, --server	The CDA server (url).*
-u, --user	The CDA username.*
-p, --password	The password for the CDA user.*
-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password". If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	existinginfrastructure list serverpool	
ALIAS	xinfra list svrpool xi list svrpool	
SYNOPSIS	existinginfrastructure list serverpool [-d <i>domainId</i>]	
DESCRIPTION	Lists all available server pools.	
OPTIONS	-d, --domainId	Domain Identifier.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <i><name>=<value></i> ", where <i><name></i> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure delete serverpool

ALIAS xinfra delete svrpool
xi delete svrpool

SYNOPSIS existinginfrastructure delete serverpool <Name> [-d *domainId*]

DESCRIPTION Deletes a server pool. Specify server pool name.

OPTIONS	Name	A Server Pool name.
	-d, --domainId	Domain Identifier.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password".
		If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure get serverpool

ALIAS xinfras get svrpool
xi get svrpool

SYNOPSIS existinginfrastructure get serverpool <Name> [-d domainId]

DESCRIPTION Fetches server pool information.

Specify server pool name.

Specify "--domainId" to indicate which domain the specified serverpool belongs to.

OPTIONS	Name	A Server Pool name.
	-d, --domainId	Domain Identifier.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	existinginfrastructure add template	
ALIAS	xinfra add tmplt xi add tmplt	
SYNOPSIS	existinginfrastructure add template -nm <i>name</i> [-atchsvrpl <i>attachserverpool</i>] [-d <i>domainId</i>]	
DESCRIPTION	Adds a template. To attach more than one server pool to template, specify "--attachserverpool" multiple times.	
OPTIONS	-nm, --name -atchsvrpl, --attachserverpool -d, --domainId -s, --server -u, --user -p, --password -c, --config	A Template name. A Server Pool name. Domain Identifier. The CDA server (url).* The CDA username.* The password for the CDA user.* A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password". If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure set template

ALIAS xinfra set tmplt
xi set tmplt

SYNOPSIS existinginfrastructure set template <Name> [-nm *name*]
[-atchsvrpl *attachserverpool*] [-d *domainId*]

DESCRIPTION Edits template information.
Specify template name.

Specify "--name" to change the name of template.

Specify "--attachserverpool" to attach serverpools to template.

To attach more than one server pool to template, specify "--attachserverpool" multiple times."

OPTIONS	Name	A Template name.
	-nm, --name	A Template new name.
	-atchsvrpl, --attachserverpool	A Server Pool name.
	-d, --domainId	Domain Identifier.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME	existinginfrastructure list template	
ALIAS	xinfra list tmplt xi list tmplt	
SYNOPSIS	existinginfrastructure list template [-d <i>domainId</i>]	
DESCRIPTION	Lists all available templates. Server pool names are displayed for any server pools attached to a template.	
OPTIONS	-d, --domainId	Domain Identifier.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password".
		If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. `--server(-s)` will default to `localhost` value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure delete template

ALIAS xinfra delete tmplt
xi delete tmplt

SYNOPSIS existinginfrastructure delete template <Name> [-d *domainId*]

DESCRIPTION Deletes a template. Specify template name.

OPTIONS	Name	A Template name.
	-d, --domainId	Domain Identifier.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format " <code><name>=<value></code> ", where <code><name></code> is "server", "user", or "password".
		If not specified, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

NAME existinginfrastructure get template

ALIAS xinfra get tmplt
xi get | tmplt

SYNOPSIS existinginfrastructure get template <Name> [-d *domainId*]

DESCRIPTION Fetches template information.

Specify template name.

OPTIONS	Name	A Template name.
	-d, --domainId	Domain Identifier.
	-s, --server	The CDA server (url).*
	-u, --user	The CDA username.*
	-p, --password	The password for the CDA user.*
	-c, --config	A file containing default general option values in the format "<name>=<value>", where <name> is "server", "user", or "password".
		If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.*

* Must be pre-defined in a configuration file or defined in the command line. --server(-s) will default to localhost value(s) if not pre-defined in a configuration file.

B HP Diagnostics Probe Deployment

HP Diagnostics is an agent-based monitoring tool that specializes in performance monitoring and analysis. It contains three installable components: the Commander, the Mediator, and various instrumentation probes. The Commander and Mediator act as a central collector of data from the instrumentation probes that are installed in target servers or applications. These probes are monitoring agents that collect performance data from the application servers and send it to the Commander or the Mediator which display the data to users. This requires the probes be installed as platform software on the target systems when a platform is provisioned. This section explains how to configure HP CDA platform software to install HP Diagnostics probes on provisioned systems. The following steps will be explained:

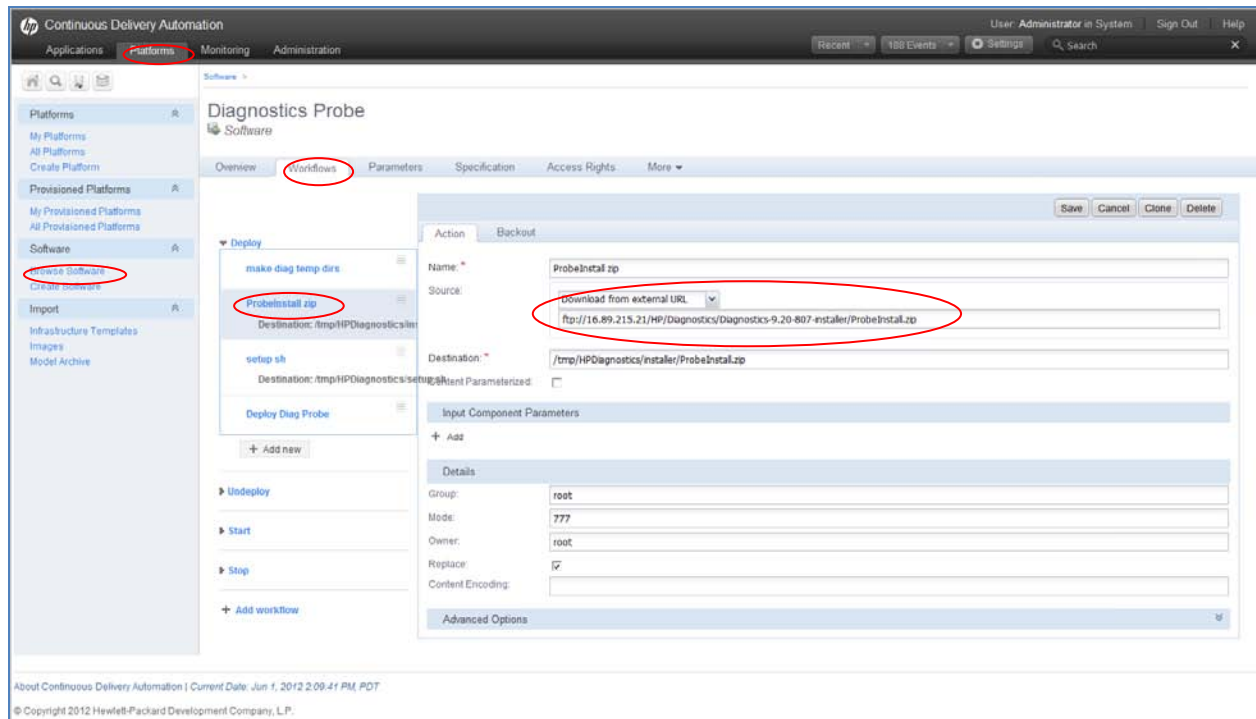
1. Import platform software for installing Diagnostics probes
2. Configure platform software to point to the Diagnostics distribution
3. Configure logical platform to attached the platform software

Import the platform software for installing Diagnostics probes

1. Click the **Platforms** tab in the HP CDA interface.
2. Click the [Model Archive](#) link that is in the navigation pane at the left side of the screen.
3. Select the **File** radio button.
4. Click the **Browse...** button and select the model Diagnostics probe platform software archive file.
5. Click the **Import** button to start the archive import process.

Configure the platform software to point to the HP Diagnostics distribution

1. Click the **Platforms** tab in the HP CDA interface.
2. Click the [Browse Software](#) link that is in the navigation pane.
3. Select **Diagnostics Probe** in the list of platform software that is displayed.
4. Click the **Workflows** tab.
5. Expand the **Deploy** workflow and click the **ProbelInstall.zip** programming action.
6. Select **Download from external URL** in the "Source" field and enter the FTP location of the Diagnostics distribution See **Error! Reference source not found..**



Configure the logical platform to attach the platform software

1. Click the **Platforms** tab in the HP CDA interface.
2. Click the **My Platforms** link that is in the navigation pane.
3. Select your platform in the list of logical platforms that are displayed.
4. Click the **Designer** tab.
5. Click the App server icon in the graphical display of the platform.
6. Click the **Add** button that is on the **Software** tab in the popup window.
7. Select the **Add-on SW** radio button in the “Add New Software” dialog box that appears, and click **Next**.
8. Select **Diagnostics Probe** in the left pane of the “Add New Software” dialog box, click **Next**, and then click **Finish**.
9. Click the **Save** button.

C For More Information

HP Continuous Delivery Automation (HP CDA) provides the following information resources available from the HP Software Product Manuals Web site at <http://support.openview.hp.com/selfsolve/manuals>:

- *HP CDA Information Resources List*: list of all HP CDA information resources and publication dates.
- *HP CDA Installation and Configuration Guide*: instructions about how to install and configure HP CDA.
- *HP CDA Release Notes*: list of release contents, installation notes, and any known problems and workarounds.
- *HP CDA Solution and Software Support Matrix*: information about platform support requirements for HP CDA core functions with links to requirements for component products.
- *HP CDA Concepts Guide*: high-level overview of the HP CDA solution.
- *HP CDA Troubleshooting Guide*: troubleshooting information including basic triage of common problems and use of integration log files.

Note: To check for recent updates or to verify that you are using the most recent edition of a document, go to:

<http://h20230.www2.hp.com/selfsolve/manuals>

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

<http://h20229.www2.hp.com/passport-registration.html>

You can also click the [New users - please register](#) link on the HP Passport login page. You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.