

# HP Continuous Delivery Automation

For the Windows<sup>®</sup> and Linux operating systems

Software Version: 1.30

## HP Continuous Delivery Automation Help

Document Release Date: December 2013

Software Release Date: August 2013



## 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 2012-2013 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.

UNIX® is a registered trademark of The Open Group.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

## 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:

**<http://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 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](http://h20230.www2.hp.com/new_access_levels.jsp)**

## Disclaimer for PDF Version of Online Help

This document is a PDF version of the online help. This PDF file is provided so you can easily print multiple topics from the help information or read the online help in PDF format.

**Note:** Some topics do not convert properly to PDF, causing format problems. Some elements of online help are completely removed from the PDF version. Those problem topics can be successfully printed from within the online help.

# Contents

Contents .....	6
Applications .....	17
Using a Model-Driven Approach .....	18
Using an Application Development Lifecycle .....	19
Understanding Model Relationships .....	19
Understanding Application Models .....	19
Opening Application Windows .....	20
Opening Application List Windows .....	21
Opening Application Version Details Windows .....	22
Opening Application Model Details Windows .....	22
Creating, Defining, Publishing and Deleting Applications .....	22
Creating an Application and Application Version .....	23
Editing Application Properties .....	24
Cloning an Application Model .....	24
Changing the Domain for an Application Model .....	25
Defining an Application Layer .....	26
Configuring Application Layer Workflows .....	28
Accessing an Application Workflow .....	28
Configuring a Layer Workflow Action .....	29
Delegating to Platform Software Workflows .....	29
Configuring a Layer Workflow Backout .....	30
Defining Application Options .....	31
Using the Application Consumability Lifecycle .....	33
Publishing Application Service Designs .....	35
Using Application Service Designs .....	37
Exporting and Importing Applications .....	38
Exporting an Application .....	38
Importing An Application .....	39
Creating, Defining and Deleting Application Bundles .....	41
Adding and Cloning Bundle Definitions .....	41

Adding Software Artifacts .....	42
Using Software Artifact Provider Plugins to Add Artifacts .....	43
Adding Software Artifacts with the CVS Artifact Provider Plugin .....	44
Adding Software Artifacts with Jenkins Artifact Provider Plugin .....	45
Adding Software Artifacts with the Server File System Artifact Provider Plugin ..	47
Adding Software Artifacts with the SFTP Artifact Provider Plugin .....	48
Adding Software Artifacts with the SVN Artifact Provider Plugin .....	50
Selecting an Existing File from DSL .....	51
Uploading Software Artifacts to DSL .....	52
Downloading Software Artifacts from an External URL .....	53
Deleting Bundle Definitions .....	54
Adding, Cloning, Editing and Deleting Deployment Topologies .....	54
Adding a Deployment Topology .....	55
Cloning a Deployment Topology .....	55
Editing a Deployment Topology .....	56
Validating a Deployment Topology .....	56
Deleting a Deployment Topology .....	58
Adding Input Component Parameters to Levels .....	58
Adding a Parameter Definitions Group to an Application Model .....	58
Adding an Input Component Parameter to a Layer .....	59
Adding an Input Component Parameter to a Workflow .....	59
Adding and Deleting Connections and Endpoints .....	60
Adding and Deleting Connections .....	61
Creating, Modifying and Deleting Endpoints .....	61
Deploying and Redeploying Applications .....	62
Deployment of an Application .....	62
Redeployment of an Application .....	63
How to Deploy an Application .....	64
How to Redeploy an Application .....	67
How to Execute Redeploy Workflows .....	71
Opening a Deployed Application Window .....	71

Setting Start and Stop Workflows for Deployment .....	72
Undeploying an Application .....	72
Deploying Local Developer Builds .....	73
Running CLI Commands Remotely .....	73
Configuring the Deploy Script .....	74
<b>Platforms .....</b>	<b>75</b>
Using Platform Models .....	76
Opening a List of Platform Models .....	77
Opening a Platform Model Details Window .....	77
Creating a Platform Model .....	79
Modifying a Platform Model .....	80
Modifying Infrastructure Design .....	81
Opening a Platform Designer Tab .....	82
Editing, Adding and Deleting Infrastructure Templates .....	84
Modifying Platform Software Capabilities .....	85
Modifying Platform Hardware and Operating System Capabilities .....	88
Modifying Platform Deployer Capabilities .....	90
Modifying Platform Model Connections .....	93
Unpublishing, Enabling and Disabling Platform Service Designs .....	94
Using the Platform Model Consumability Lifecycle .....	96
Modifying Platform Model Stored Parameters .....	98
Modifying Platform Model Specifications .....	100
Modifying Platform Model Access Rights .....	100
Publishing Platform Model Service Designs .....	101
Deleting a Platform Model .....	103
Using Add-on Platform Software .....	104
Browsing Lists of Add-on Platform Software .....	105
Using HP CDA Out-of-Box Monitoring Agents .....	106
Opening an Add-on Platform Software Details Window .....	108
Creating Add-on Platform Software .....	109
Modifying Add-on Platform Software .....	110

Modifying Platform Software Requirements and Capabilities .....	110
Modifying Platform Software Access Rights .....	111
Using Platform Software Workflows .....	112
Using Platform Software Parameters .....	112
Modifying Platform Software Specifications .....	112
Using the Platform Software Consumability Lifecycle .....	113
Deleting Add-on Platform Software .....	115
Exporting and Importing Platform Software .....	116
Exporting Platform Software .....	116
Importing Platform Software .....	117
Provisioning and Deprovisioning Platforms .....	118
Opening a List of Provisioned Platforms .....	119
Opening a Provisioned Platform Details Window .....	120
Provisioning a Platform .....	121
Using the Platform Provisioning Report .....	123
Deprovisioning a Platform .....	123
Using Shared Platforms .....	124
Sharing a Platform .....	125
Unsharing a Platform .....	127
<b>Infrastructure .....</b>	<b>128</b>
Launching a Provider Designer .....	129
Launching a Provider Manager .....	130
Creating Infrastructure Templates from the CLI .....	130
Creating Infrastructure Templates from the CLI, an Example .....	132
Importing Infrastructure Templates .....	133
Opening Infrastructure Template Windows .....	135
Manually Synchronizing Infrastructure Templates .....	136
Using the Synchronize Template Report .....	137
Editing Infrastructure Templates .....	138
Deleting Infrastructure Templates .....	138
<b>Monitoring Administration .....</b>	<b>139</b>

Using Agent-Based vs. Agent-Less Monitoring Tools .....	139
Implementing Monitoring .....	140
Step 1: Identify the Monitoring Tools in Your Environment .....	141
Step 2: Configure Your Monitoring Tool to Send Events to HP CDA .....	141
Step 3: Configure HP CDA to Communicate with the Monitoring Tool .....	141
Step 4: Define Monitors .....	142
Creating Monitors by Importing HP SiteScope Monitoring Templates .....	142
Creating Monitors Manually .....	143
Step 5: Define Monitoring Policies .....	145
Out-of-Box Monitoring Policies .....	146
Step 6: Assign Monitoring Policies to Application Topologies .....	150
Remonitoring .....	152
Deploying and Undeploying Applications with Monitors .....	153
Aggregating Monitoring Metrics within Tiers .....	153
Step 7: Review the State of Monitored Applications .....	155
Monitoring Applications in Amazon Elastic Compute Cloud .....	156
HP Diagnostics and Amazon EC2 .....	157
HP SiteScope and Amazon EC2 .....	158
Establishing SSH Tunnels .....	159
Using SSH Tunnels with HP Diagnostics .....	159
Using SSH Tunnels with HP SiteScope .....	160
Using SSH Tunnels with Nagios .....	161
<b>Administration .....</b>	<b>162</b>
Managing Content .....	163
Organizing by Tenant and Domain .....	164
Creating and Deleting Tenants .....	165
Creating and Deleting Domains .....	166
Using Domains and Repositories .....	168
Using Domains When HP CDA and HP ALM are Integrated .....	171
Disabling Default Domain Sign-In .....	171
Setting Default Tenant or Domain Access Rights .....	172

Adding Parameters to Tenants and Domains .....	175
Enforcing a Base Path .....	175
Adding Provisioning Keys to a Centralized Key Store .....	175
Exporting Tenant or Domain Content .....	176
Importing and Exporting Archives .....	177
Exporting Single Models to a ZIP Archive File .....	177
Exporting Multiple Models to a ZIP Archive File .....	178
Importing Model Archives .....	179
Managing Lifecycle Processes .....	180
Using Lifecycle Best Practices .....	181
Creating Lifecycle Processes .....	182
Defining Lifecycle Stages .....	183
Defining Lifecycle Transitions .....	185
Defining Lifecycle Tasks .....	185
Defining Lifecycle Policies .....	186
Adding, Deleting and Editing Lifecycle Approvers .....	187
Defining Lifecycle Automatic Actions .....	188
Defining Lifecycle Permissions .....	188
Publishing a Lifecycle Process .....	189
Exporting a Lifecycle Process .....	190
Managing Policies and Assertions .....	190
Creating and Deleting Technical Policies .....	192
Default Technical Policies .....	194
Adding, Deleting and Modifying Tags .....	194
Opening a Tag Management Window .....	195
Importing and Exporting Tag Archives .....	198
Exporting and Importing Tags as XML .....	199
Exporting Tags as XML .....	200
Importing Tags as XML .....	201
Adding Tags .....	202
Editing Tags .....	202

Further Defining Tags .....	203
Adding an Interface Operation to a Tag .....	204
Removing Tags .....	207
Adding and Running Administration Tasks .....	207
Adding Custom Tasks .....	208
Adding Artifact Sync Tasks .....	209
Running Tasks .....	210
Managing Users, Groups and Roles .....	211
Managing Users .....	212
Creating and Importing Users and Administrators .....	213
Editing Users and Administrators .....	214
Assigning Users to Roles .....	214
Assigning Users to Groups .....	215
Setting Default Domains for Users .....	216
Setting a User's Default Domain Role .....	217
Setting New Artifact Ownership .....	217
Enabling and Disabling and Users .....	218
Retiring Users in Multi-Tenant Environments .....	219
Retiring Users in Environments without Tenants .....	220
Managing Groups .....	220
Creating Groups .....	221
Adding and Removing Group Members .....	221
Assigning Groups to Additional Domain Roles .....	222
Setting Default Domains for Groups .....	222
Retiring and Deleting Groups .....	223
Managing Roles .....	224
Using User Interface Roles .....	226
Using Lifecycle Roles .....	226
Creating and Deleting Roles .....	228
Changing the Sharing Principal .....	229
Adding and Removing Roles Members .....	230

Managing Security and Access Control .....	231
Integrating External Services .....	232
Integrating Artifact Plugins .....	233
Configuring a CVS Artifact Provider Plugin .....	234
Configuring a Jenkins Artifact Provider Plugin .....	235
Configuring a Server File System Artifact Provider Plugin .....	236
Configuring an SFTP Artifact Provider Plugin .....	236
Configuring an SVN Artifact Provider Plugin .....	238
Integrating Application Deployer Plugins .....	238
Defining a Deployer Plugin .....	239
Testing a Plugin Prior to Updating .....	240
Uploading New Plugins .....	240
Updating Plugins .....	240
Integrating Platform Provisioning Plugins .....	241
Uploading a Platform Provisioning Plugin .....	241
Editing a Platform Provisioning Plugin .....	242
Updating a Platform Provisioning Plugin .....	242
Deleting a Platform Provisioning Plugin .....	243
Configuring a Platform Plugin .....	243
Adding a Platform Plugin Configuration .....	244
Adding Authentication Keys to a Platform Plugin Configuration .....	244
Deleting Authentication Keys from a Platform Plugin Configuration .....	245
Editing a Platform Plugin Configuration .....	246
Deleting a Platform Plugin Configuration .....	246
Integrating Existing Infrastructure .....	246
Exporting Plugins .....	248
Exporting and Importing Plugin Configurations .....	249
Exporting Plugin Configurations .....	249
Importing Plugin Configurations .....	250
Managing Configurations and Systems .....	251
Create Empty Data Artifacts .....	252

Customizing SSL .....	252
Changing System Settings .....	253
Changing the License System .....	254
Running a Self-Test .....	255
Changing Policies for Artifact Form Validation .....	256
Running Administration Scripts .....	256
Running the cdaexec Administration Script .....	257
Running the export Administration Script .....	259
Running the import Administration Script .....	261
Running the reset Administration Script .....	264
Running the serverstart Administration Script .....	264
Running the serverstop Administration Script .....	265
Running the servicestart Administration Script .....	265
Running the servicestop Administration Script .....	266
Running the setup Administration Script .....	266
How to Start the Setup Tool Wizard .....	267
Running the Setup Tool Wizard to Change the HP License Key .....	268
Running the Setup Tool Wizard to Install Updates .....	268
Running the Setup Tool Wizard to Configure Server Endpoints .....	269
Running the Setup Tool Wizard to Configure an SMTP Server .....	270
Running the Setup Tool Wizard to Change Lightweight SSO Settings .....	271
Running the Setup Tool Wizard to Un-install HP CDA .....	272
Running the setup Script as a Command Line Tool .....	272
Running the ssltool Administration Script .....	273
Shared Content .....	276
HP CDA Command Line Interface (CLI) .....	276
Floating or Elastic IP Addresses .....	339
HP Diagnostics Probe Deployment .....	341
Configuring Monitoring Tools to Send Events to HP CDA .....	343
Configuring HP Diagnostics to Send Events to HP CDA .....	343
Configuring HP SiteScope to Send Events to HP CDA .....	344

Configuring HP Operations Manager to Send Events to HP CDA .....	346
Revising the HP Operations Manager Configuration .....	347
Uninstalling the HP CDA Integration .....	348
Configuring Nagios to Integrate with HP CDA .....	348
Managing Parameters .....	350
Adding Parameters .....	351
Using Stored Parameter Sets .....	354
Creating or Editing a Stored Parameter Set .....	355
Saving Parameter Values to a Stored Parameter Set .....	355
Using a Stored Parameter Set During Provisioning or Deployment .....	355
Using Substitution Variables .....	356
Using Substitution Variables in Workflow Actions .....	356
Using Substitution Variables in Fields .....	357
Using Substitution Variables in Inline Strings .....	357
Adding Reference Variable Parameters .....	359
Using Previous Command Output Variables .....	363
Canceling Provisioning or Deployment Operations .....	368
Canceling During Provisioning or Deployment .....	368
Backing Out of Provisioning or Deployment Operations .....	369
System-Generated Backout Steps .....	369
User-Created Backout Steps .....	370
Forced Backout .....	370
Manual Backout .....	371
Forced Cleanup of Failed Operations .....	371
HP CDA Workflows .....	372
HP CDA Programming Operations .....	372
Placed File Programming Operation .....	374
Placed Directory Programming Operation .....	379
Executed Script Programming Operation .....	380
Service Command Programming Operation .....	383
External Component Programming Operation .....	387

Invoke Workflow Programming Operation .....	388
Enabling and Disabling Programming Operations .....	389
Base Path Parameter .....	391
For More Information .....	392

# Applications

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

The following topics provide information about creating, managing and deploying applications.

Topic	Information Included
<a href="#">"Using a Model-Driven Approach " on the facing page</a>	Includes information on HP CDA models, lifecycles and relationships.
<a href="#">"Opening Application Windows " on page 20</a>	How to open windows that show information about applications, versions and bundles.
<a href="#">"Creating, Defining, Publishing and Deleting Applications " on page 22</a>	How to perform the functions used with applications, from creation to end-of-life.
<a href="#">"Creating, Defining and Deleting Application Bundles " on page 41</a>	How to perform the functions used with bundles, from creation to end-of-life.
<a href="#">"Adding, Cloning, Editing and Deleting Deployment Topologies " on page 54</a>	How to work with the dynamic relationship between application layers and platform compute resources.
<a href="#">"Adding Input Component Parameters to Levels " on page 58</a>	How to add input components at different application model levels. As a best practice, add and reuse parameters for multiple deployments.

<a href="#">"Adding and Deleting Connections and Endpoints " on page 60</a>	How to use the connections and endpoints that define how application layers connect to each other.
<a href="#">"Deploying and Redeploying Applications " on page 62</a>	How to deploy and undeploy applications.

## Using a 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

An application model contains layers, and layers contain application workflows. When a workflow runs, components in the workflow translate into a deployer-specific model. A 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 bundle 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.

Topic	Information Included
<a href="#">"Using an Application Development Lifecycle " on next page</a>	Information about the continuous process of managing an application through governance, development, and maintenance.
<a href="#">"Understanding Model Relationships " on next page</a>	Information about the relationships between models, topologies and platforms.
<a href="#">"Understanding Application Models " on next page</a>	Information about the terms you find in HP CDA help.

## Using an Application Development Lifecycle

An application moves through lifecycle stages, such as Development, QA, Staging, or Production. This is the Application Lifecycle Management (ALM) workflow—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)
- other products, such as Opscode Chef

See the *HP Continuous Delivery Automation Installation and Configuration Guide* for more information about deployment providers.

## Understanding Model 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.

## Understanding Application Models

In HP CDA, you must define how you want the application and bundle server artifacts deployed on target platforms. An 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 bundles systems. You can deploy applications using deployment topologies, manage application deployments, and track the lifecycle of application bundles.

Conceptually, an 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 container
  - An application version defines its package definition (the external bundle artifacts to access), including the location of the bundle 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.
- **Options** specify the organization of layers into different options of the application, which can be selected to be deployed at deployment time.
- **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. Workflows can contain programming operations. To learn more, see ["HP CDA Workflows " on page 372](#) and ["HP CDA Programming Operations " on page 372](#)
- **Deployment Topology**: Map the application model's layers to server groups in a certain platform.

## Opening Application Windows

The following topics provide information about how to open application windows.

Topic	Information Included
<a href="#">"Opening Application List Windows " on next page</a>	How to open a window that displays a list of applications.
<a href="#">"Opening Application Version Details Windows " on page 22</a>	How to open a window that displays application version details.
<a href="#">"Opening Application Model Details Windows " on page 22</a>	How to open a window that displays application model details.

## Opening Application List Windows

Access to applications depends on which type of user you log in as. Users who need easy access to application models are application developers, domain administrators and tenants.

You can install HP CDA as a multi-tenancy application and create tenants. Domain administrators manage access to tenant data. Access is specific to a certain tenant, so tenants cannot access each other's data.

For example, tenants can be separate companies, departments within a company, and so on. The 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.

To view applications, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA

The My Applications and All Applications windows provide the following tabs:

- **Simple View:** Lists all applications that you, as an application developer, create. 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 the application lifecycle. Status values show 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 their target domain, and shows when an application version was last modified. This view allows you to perform actions on application artifacts:
  - **Edit:** Edits all selected application artifacts. You can edit application properties, change the application owner, change read and 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).

## Opening Application Version Details Windows

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.

## Opening Application Model Details Windows

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. On the Overview tab, click a model name to open an application model details window.

## Creating, Defining, Publishing and Deleting Applications

Layers are the primary components of an application model. (A layer corresponds to a tier in a multi-tier application.) An application model defines the layers of the application, including the recipe for managing the deployment of the application.

The following topics provide information about using applications and application models.

Topic	Information Included
<a href="#">"Creating an Application and Application Version" below</a>	How to create an application and define its version qualities.

<a href="#">"Editing Application Properties " on the facing page</a>	How to edit artifact properties, such as the name and description.
<a href="#">"Cloning an Application Model " on the facing page</a>	How to clone an entire application model, including information such as layers, topologies and parameters.
<a href="#">"Changing the Domain for an Application Model " on page 25</a>	How to change the domain for artifacts, sub-artifacts, and dependencies.
<a href="#">"Defining an Application Layer " on page 26</a>	How to create a layer for each application tier.
<a href="#">"Configuring Application Layer Workflows " on page 28</a>	How to access and configure workflow actions and backouts.
<a href="#">"Using the Application Consumability Lifecycle " on page 33</a>	How to move an application to different lifecycle stages that govern whether the application can be shared (consumed).
<a href="#">"Publishing Application Service Designs " on page 35</a>	How to pass an application service design to a provider.
<a href="#">"Using Application Service Designs " on page 37</a>	How to view, publish, enable and disable a service design.

## Creating an Application and Application Version

When you create an application—you must first name it and then define its version properties.

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, verify that version numbers are unique.

To create an application, complete the following steps:

1. Click the Applications tab to open the Applications Home window.
2. On the Applications menu, choose **Create Application** to open the Create Application wizard.
3. Make entry in the Applications Properties dialog box:
  - a. Type an application name and (optional) description.
  - b. Click **Next**.
4. Make entry in the Application Version Properties dialog box:

- a. Type a version name and number for the application.
  - b. *Optional*: Enter a brief application version description.
  - c. On the Build Lifecycle Process menu, choose **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.
5. Click **Finish**. The application and the application version are created.

## Editing Application Properties

To edit application properties, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. On the Overview tab, click a model name to open an application model details window.
5. Click **Edit** on the right to open an edit window and make entry
  - a. Type a change to the name or description.
  - b. Click **Save**.

## Cloning an Application Model

You can clone an entire application model, including all the information you have added, such as layers, topologies, and parameters.

As a common use case, you create a copy (clone) of the application model, and then make minor changes to the clone. In this way, you can produce the next version of the application model or 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. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. On the Overview tab, click a model name to open an application model details window.
5. Click **Clone** on the right side of the window.
6. On the Clone dialog box, click **OK** to confirm that you want to clone the application model.
7. The New Events dialog shows 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. Click **Edit** on the Application Model Actions window to rename a cloned application model.

## Changing the Domain for an Application Model

You can publish artifacts, sub-artifacts, 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 for an application model, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA

3. On the Simple view, click a version name to open an application version details window.
4. On the Overview tab, click a model name to open an application model details window.
5. Click **Change Domain** to open the Change Domain dialog box and make entry:
  - a. Choose a Target Domain, such as System or Default. If you create custom domains, they also appear on this menu.
  - b. *Optional:* Expand **Advanced Options**.
    - i. Select whether to include sub artifacts and dependencies, as required.
    - ii. Make sure that **Retain** is set as the Artifact Ownership value.
    - iii. *Optional:* If you need to reassign ownership, click **Select** to make changes.
  - c. 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 the 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).

**Note:** Application layers, once created, can be added to *application options*, which allow you to design the application with separately deployable options. For more information, see "[Defining Application Options](#)" on page 31.

To define an application layer:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. On the Overview tab, click a model name to open an application model details window.
5. Click the **Layers** tab.
6. Click **Add Layer**.
7. In the Layer Name dialog box, type a name, and click **Add**.
8. Expand the layer that you just created.
9. On the Workflows tab, define the actions that will be executed, such as Deploy, Undeploy, Start, and Stop. To learn more, see "[HP CDA Workflows](#) " on page 372 and "[HP CDA Programming Operations](#) " on page 372
10. *Optional:* Expand the Add Workflow link to define custom workflows, such as Pause, Restart, and Backup.
11. On 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. For more information, see "[Managing Parameters](#) " on page 350
12. On the Requirements tab, click **Change** 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. If you change requirements after the layer is included in a deployment topology, you might invalidate the topology. For more information, see "[Validating a Deployment Topology](#) " on page 56.
13. On 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. Click **Endpoints**, and then click **Create New Endpoint**. On the Add window, type a name and port. *Optional:* Select **Load Balanceable** 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** option, you must enter load balancer parameter values in the deployment topology map.

- b. Click **Add Connection**, and then click **to Layer**.
- c. On the Select Artifact window, specify the endpoint.

## Configuring Application Layer Workflows

An application workflow is a sequence of actions that are performed on an application. To learn what workflows HP CDA provides, see ["HP CDA Workflows " on page 372](#)

**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. On the Action tab, click **Browse** to select an external component.

The following topics provide information about application workflows.

Topic	Information Included
<a href="#">"Accessing an Application Workflow " below</a>	How to open a window where you can see layer workflows.
<a href="#">"Configuring a Layer Workflow Action " on next page</a>	How to add programming operations (such as Placed File, Executed Script and so on) to a workflow.
<a href="#">"Delegating to Platform Software Workflows" on next page</a>	How to invoke a workflow in platform software from an application workflow.
<a href="#">"Configuring a Layer Workflow Backout " on page 30</a>	How to define the steps that are required to clean up if a workflow fails.

## Accessing an Application Workflow

To access an application workflow:

1. In the Simple View, click the application version.
2. In the Application Version view, click an application model.
3. In the Application Model view, click 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, click the **Workflows** tab to view workflows for the layer.

## Configuring a Layer Workflow Action

You can configure programming operations, (such as Placed File, Executed Script and so on) at the application layer workflow level.

For a list of programming operations and how to create them, see ["HP CDA Programming Operations " on page 372](#)

When you create an application layer workflow Placed File programming operation, your use case determines what you enter in the Details section. You can configure permissions on the file that will be placed on the target server:

**Opscode Chef** deployer and **Linux** target platform: Use the Details section to set up Linux-style permissions.

**HP Server Automation** deployer and **Microsoft Windows** or **Linux** target platform: Use the Details section to set up platform-based permissions.

**Opscode Chef** deployer and **Microsoft Windows** target platform: Do not use the Details section. It does not apply.

## Delegating to Platform Software Workflows

HP CDA supports delegation of control from Invoke Workflow programming operations in application layers to platform software workflows. This functionality allows such platform software workflows to be invoked from multiple applications, or from different layers within the same application.

Delegation from Invoke Workflow programming operations to platform software workflows is accomplished through *Interface operations* (referred to as *Interfaces* in the HP CDA user interface), which you define on *Tags*. When the tag containing the interface operations is specified as a Capability of a given platform software, HP CDA automatically inserts these defined Interfaces as workflows into the software, and you can add the necessary programming operations to those workflows that will run when control is delegated to them.

When you define Interfaces on tags, you can also add input component parameters to them. Such input component parameters can be used to pass parameter values from application layers to software workflows when the workflows execute.

The general steps that are required to implement delegation to a platform software workflow are as follows:

- If the platform software that will contain the workflow to which control will be delegated has not been created, create it. See ["Creating Add-on Platform Software " on page 109](#) and ["Modifying](#)

[Add-on Platform Software](#) " on page 110 for information.

- Define the necessary Interfaces on the Tags that you will add to the platform software as Capabilities, and add any input component parameters to the Interfaces. See ["Adding an Interface Operation to a Tag"](#) on page 204 for information.
- Add the tags to the platform software as Capabilities. This action automatically places empty workflows into the platform software. See ["Modifying Platform Software Requirements and Capabilities"](#) on page 110 for information.
- Add the necessary programming operations and relevant parameters to the workflows. These are the operations that will be executed when control passes from the pertinent application layer to the software workflows. See ["HP CDA Programming Operations"](#) on page 372 for information.
- Add the tags to the pertinent application layers from which you want to delegate control as Requirements. The topic ["Defining an Application Layer"](#) on page 26 contains information on how to add Requirements to an application layer.
- Add Invoke Workflow programming operations to the layers, referencing the Interface operations to the platform software workflows to which you want that Invoke workflow operation to delegate control. See ["Invoke Workflow Programming Operation"](#) on page 388 for information.
- Assign the platform software to the logical platform on which the application will be deployed. When the Invoke Workflow programming operation is invoked from the application layer, all of the platform software that is attached to the logical platform is scanned and the relevant workflows of the platform software which has been tagged as a Capability is invoked. See ["Modifying Platform Software Capabilities"](#) on page 85 for information.

## ***Configuring a Layer Workflow Backout***

In this use case, you define the steps that are required to clean up if a workflow execution fails. This process backs out previously completed actions.

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

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

The following task steps are generic—you can use them to configure a backout for any type of layer workflow action in an application model.

To configure a workflow layer backout operation:

1. Navigate to the Application Model view and select the **Layers** tab.
2. Expand a layer, such as the DB Layer, the App Server Layer, and so on.
3. Select the **Backout** tab.
4. Complete the setup instructions and **Advanced Options**.
5. Click **Save**.

## Defining Application Options

Application Options allow you to group layers in an application model into separately-deployable options of the application. This allows the application designer to define and model all of the options of an application early in the design process, but only deploy certain options that are ready for deployment and testing, for example.

In a modeled application, application options can be created and layers added to them. At deployment time, you select which options of the application you wish to deploy, and then the following occurs during deployment.

- All mapped layers that are not associated with application options are deployed, and any such unmapped layers are skipped.
- The layers that are associated with the options that have been selected for deployment are deployed.
- Layers that are associated with options that have not been selected for deployment are not deployed.

**Note:** Layers are deployed onto the servers in a provisioned platform according to a *deployment topology* that is chosen during deployment time. Separate deployment topologies can be created for the different application options that are specified in this procedure. Refer to ["Adding a Deployment Topology" on page 55](#) for information on creating deployment topologies.

### **To add an application option:**

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:

- **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
  4. On the Overview tab, click a model name to open an application model details window.
  5. Click the **Options** tab.
  6. Click **Add Option** to open the Create Option dialog box.
  7. Type a name for the option and an optional description into the pertinent fields, and then click the **Save** button.

***To add layers to an application option:***

1. On the **Options** tab, click the Edit (pencil) icon that is to the far right of the screen, in line with the option name.  
  
The Edit Option dialog box opens.
2. Click the **Select Layers** button to open the Select Layers dialog box.  
  
The Select Layers dialog box lists all of the layers that are in the application model.
3. Select the checkboxes that precede the names of the layers that you want to add to the option.

**Note:** A layer can only be included in one application option at any given time. If a layer is currently included in an application option, the name of the option will be shown in the Option Name column of the table in the Select Layers dialog box. If you select such a layer, a warning message (**Warning:** <layer\_name> is already a part of <option\_name> and will be moved to this option.) will appear in the dialog box, and the layer will be removed from its current option and added to this option should you proceed with the operation.

4. Click the **Save** button to close the Select Layers dialog box.  
  
Notice that the selected layers now appear under the Layers section of the Edit Option dialog box.
5. Click **OK** in the Edit Option dialog box to close it.

***To remove layers from an application option:***

1. On the **Options** tab, expand the application option from which you want to remove layers.
2. Click the Delete (**X**) icon that is to the far right of the screen, in line with a layer that you want to remove from the option.
3. Click **OK** in the Remove Layer message box that appears.

**To delete an application option:**

1. On the **Options** tab, click the Delete (**X**) icon that is to the far right of the screen, in line with the option name.
2. Click **OK** in the Delete Option message box that appears.

## Using the Application Consumability Lifecycle

The application Lifecycle tab shows the stages that govern whether an application can be shared with other HP CDA users and published. When an application developer works on an application model, the application does not appear in the All Applications view and it cannot be published.

As you work with a lifecycle, the graphic changes, so it becomes a visual timeline of your application's lifecycle.

You can click on any stage to see information about the stage, but this action does not advance the application. The application's current stage displays larger than the other stages.

Lifecycle stages:

- **New:** The application is in progress. It cannot be viewed by other HP CDA users or published. (The publish icon, on the Overview and Service Designs tabs, is not active.) When an application developer decides that the application is ready to share, the developer moves it to Release Voting.
- **Release Voting:** When the application moves to this stage, HP CDA checks any policies associated with it. For example, a policy may require approvals at this stage, so approvers receive email notification of their approval tasks. If a policy fails, HP CDA returns the application to the New stage. Otherwise, the application advances to the Released stage.

**Note:** HP CDA requires all related software artifacts to be in the Released state before the platform can advance to Released. However, after the platform moves into Released state, a user can move related software artifacts back to the New Stage.

- **Released:** The application appears on All Applications and it can be published. The publish icon

is active.

- **Retirement Voting:** When an application is no longer needed, an application developer can move it to the Retirement Voting stage. Any policies associated with it will be checked and executed, just as during the Release Voting stage. If policies are met, the application moves to Retired. Otherwise, it returns to Released.
- **Retired:** The application cannot be published or deployed.

Administrators can add the following to stages:

- **Tasks:** Any assigned tasks must be completed for the application to advance to the next stage.
- **Policies:** Any policies must be validated for the application to advance to the next stage.
- **Approvers:** Approvers receive email notification when their assigned approval tasks come due.

### Lifecycle Actions

Application designers can perform the following actions on the Lifecycle tab.

You can advance the application to the next stage.

To advance to the next stage, complete the following steps:

1. Click **Move to Stage**.

The next stage name displays.

2. Click the name to confirm.

The application advances to the next stage.

If you have multiple lifecycles defined, you can move the application to any stage in any lifecycle.

To move the application to a selected stage and lifecycle, complete the following steps:

1. Click **Set Stage/Process**.
2. In the **Set Process** dialog box:

Select a lifecycle from the **Process** menu.

Select a stage from the **Stage** menu.

Select **Approved**.

Click **Set**.

The Lifecycle graphic displays the application at the selected stage and process.

You can end governance and remove the application from its lifecycle.

To end the governance of an application, complete the following steps:

1. Click **End Governance**.
2. Click **End Governance** in the confirmation box.

The Lifecycle graphic no longer displays. You see options to start governance.

To start governance, complete the following steps:

- Click **Start Governance**.

The application displays at the New stage of the default lifecycle process.

## Publishing Application Service Designs

An application model can be published as a service design, which means the application's service design is passed to a provider, and the provider can further modify the design to create a service offering. To publish, the following requirements must be met:

- The application and its artifacts must be in Released state.
- You must have a plugin configured for the provider.

When an application is eligible for publication, the Publish icon becomes active.

Application service designs can be published from either the application's **Overview** or **Service Designs** tabs.

To publish an application service design, complete the following steps:

1. Display the application model.
2. Click the **Overview** or **Service Designs** tab.
3. Click **Publish**.

**Note:** This operation publishes the current application model. If you click to select a current service design, it has no effect on the operation.

The **Publish an Application Service Design** wizard opens.

4. In the **Introduction** window:
  - a. Read information about wizard steps.
  - b. Click **Next**.
5. In the **Provider** window:
  - a. Choose a **Provider** on the menu.
  - b. The service design will be published to the selected provider.
  - c. Click **Next**.
6. In the **Lifecycle** window:
  - a. Select a **Default** lifecycle stage. This selection determines which lifecycle will be published with the service design.
  - b. Click **Next**.
7. In the **Service Design Information** window, type:
  - a. **Service Design Name** for the application service design.
  - b. **Service Design Description**, optional.
  - c. Click **Next**.
8. In the **Deployment Topology** window:
  - a. Choose a **Deployment Topology** on the menu. The deployment topology will be published with the service design.
  - b. Click **Next**.
9. In the **Application Parameters** window, select application parameter settings:

Accept, enter or edit settings as needed.

For information about the icons, click [here](#).

The eye icon  controls whether your entry is visible in the service design. Click to make your entry visible, or turn visibility off.

The lock icon  controls whether your entry can be changed by service subscribers. Click to mark your entry as changeable or not changeable. (A changeable setting can be over-ridden, set to read-only, by the service provider.)

**Edit values** allows you to change default values that display. Click **Edit values** to open an Edit dialog window.

10. On the **Monitoring Settings** window:
  - a. Accept, enter or edit settings for monitoring parameters, if monitoring policies are attached to the selected deployment topology.
  - b. Click **Next**.
11. A **Summary** window of your entry appears.
12. *Optional:* Click **Back** to return to any windows where you want to make changes.
13. Click **Finish**.

Publishing begins.

## Using Application Service Designs

After you publish an application model's service design, you can view it on the application's Service Design tab.

To view a list of published application models service designs, complete the following steps:

1. Display the application model.
2. Click the **Service Designs** tab.

A list of service designs displays.
3. Click the **Name** link to display the details of a service design,

You can change the status of a service design, so it is no longer available as a published design.

To unpublish a service design, complete the following steps:

1. Click to select the service design.
2. Click **Unpublish**.
3. Click **OK** to confirm.

The service design no longer displays.

You can select settings that give your service design a status of Enabled or Disabled when published. (See your service provider documentation for information on what the statuses mean for your provider.)

To enable or disable a service design, complete the following steps.

1. Click to select the service design.
2. Click **Enable** or **Disable**.
3. Click **OK** to confirm.

To publish a service design, see "[Publishing Application Service Designs](#)" on page 35.

## Exporting and Importing Applications

An application can be exported from the HP CDA repository as an XML file, which can be modified if desired and imported to the repository. The XML that is exported is in an easy to read and understand format to facilitate editing in a standard XML browser or text editor.

The following topics explain how to export and import applications.

Topic	Information Contained
<a href="#">"Exporting an Application "</a> below	How to export an application from the HP CDA repository as an XML file.
<a href="#">"Importing An Application "</a> on next page	How to import an application XML file into the HP CDA repository.

## Exporting an Application

You can export applications from the HP CDA repository as XML. When you export an application, the following components of the application are included in the exported XML:

- Application layers
- Requirements defined within application layers
- Application workflows
- Application workflow programming operations
- Parameters

The following components of the application are not included in the exported XML:

- Software artifacts
- Deployment topologies

The application export feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

To export an application as XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Click the All Applications link in the Applications section of the HP CDA user interface, and copy the exact name of the application that you want to export.
3. Open the REST client, and issue a GET request with the following characteristics:

- **URL:**

```
https://<cda_
address>:8443/cda/remote/rest/v1/application?name="<app_name>"
(Secure HTTP)
```

```
http://<cda_
address>:8080/cda/remote/rest/v1/application?name="<app_name>"
(Standard HTTP)
```

Where <cda\_address> is the IP address or FQDN of the HP CDA instance, and <app\_name> is the name of the application that you want to export. If the GET request succeeds (a status code of 200 OK is returned), the XML for the application will be returned.

**Note:** To export the XML for all applications, issue a GET request to the above URL without the ?name="<app\_name>" parameter.

- **Content Type:** application/xml
4. Copy the application XML to an editor for modification, or save it to a text file for later editing and import.

## Importing An Application

You can import application XML to the HP CDA repository. The XML that you import has been exported using the procedure ["Exporting an Application" on previous page](#) and typically has been stored to a text file. Using the import functionality, you can perform following operations:

- Create a new application by specifying an application name that does not exist already in the HP CDA repository. This will create a new application, the underlying layers and requirements, workflows, workflow programming operations, and parameters.

**Note:** You cannot define software artifacts and deployment topologies as part of an application.

**Note:** While importing an application, it is essential that the XML tag hierarchy be maintained. Hence it is advisable to export an existing application and edit its name, the underlying layers and requirements, workflows, workflow programming operations, and parameters rather than creating the XML import document from scratch.

- Update an existing application by specifying the same Application Name in the XML and specifying `force=true` in the URL as mentioned in the procedure below. In such a case, you can add additional layers and requirements, workflows, workflow programming operations, and parameters to the existing application XML and those changes will be reflected when the XML document is imported.

The application import feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

To import application XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Open the REST client and issue a POST request with the following characteristics:

- **URL:**

`https://<cda_address>:8443/cda/remote/rest/v1/application` (Secure HTTP)

`http://<cda_address>:8080/cda/remote/rest/v1/application` (Standard HTTP)

Where `<cda_address>` is the IP address or FQDN of the HP CDA instance.

- **Header:** `application/xml`

- **Body:** The application XML to be imported to the repository.

If the POST request succeeds (a status code of 200 OK is returned), the XML for the application has been imported into the HP CDA repository and you can now view it in the HP CDA interface.

**Note:** If a status code of 409 Conflict is returned, an application with the same name already exists in the HP CDA repository. To import and overwrite the application in the repository with the new version, issue the POST request with the "force" parameter set to "true," for example:

```
https://<cda_  
address>:8443/cda/remote/rest/v1/application?force=true
```

```
http://<cda_address>:8080/cda/remote/rest/v1/application?force=true
```

## Creating, Defining and Deleting Application Bundles

Topic	Information Included
<a href="#">"Adding and Cloning Bundle Definitions " below</a>	How to add and clone bundles, and the difference between those functions.
<a href="#">"Adding Software Artifacts " on the facing page</a>	How to use DSL, an external URL, or a provider plugin to add software artifacts.
<a href="#">"Deleting Bundle Definitions " on page 54</a>	How to delete a bundle.

## Adding and Cloning Bundle Definitions

You can add or clone an application bundle definition.

When you *add* a bundle definition, you create a copy of the default package definition. To do so, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.

4. Click the **Software Artifacts** tab to open a window where you see bundle definitions.
5. Click **Add Bundle Definition** to open the Bundle Definition Name dialog box. Make entry:
  - a. Type a name for the new bundle definition.
  - b. Click **Add**.

When you *clone* a bundle definition, you create a copy of the bundle that is currently selected. To do so, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Software Artifacts** tab to open a window where you see bundle definitions.
5. Select the definition that you want to clone, and click **Clone**.
6. The definition is copied and named *<Clone of Bundle Name>*.
7. In the **Name** text box, type a meaningful name.
8. Click **Save**.

## Adding Software Artifacts

The initial step 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 are components used to build a system or application, and can be the source files or binary files that your organization develops, or any additional files that the application requires.

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

**Note:** If you want to use a bundle server, you must know the connection information for that bundle server, such as the URL and username and password, if required. When you select a bundle 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.

Software artifacts can reside locally (for example, on your bundle server), or they can reside remotely and be linked using a URL. Use a continuous integration (CI) tool to manage the bundle process.

The following topics provide information about adding and reconfiguring artifacts.

Topic	Information Included
<a href="#">"Using Software Artifact Provider Plugins to Add Artifacts " below</a>	How to use artifact provider plugins to integrate with external repositories.
<a href="#">"Selecting an Existing File from DSL " on page 51</a>	How to add software artifacts from DSL.
<a href="#">"Uploading Software Artifacts to DSL " on page 52</a>	How upload software artifacts to DSL.
<a href="#">"Downloading Software Artifacts from an External URL " on page 53</a>	How to download software artifacts from an external URL.

## ***Using Software Artifact Provider Plugins to Add Artifacts***

HP CDA provides the following software artifact provider plugins:

- CVS source repository
- Jenkins build system
- HP CDA server local directory, drive, or mapped drive
- SFTP server directory
- SVN repository

Topic	Information Included
-------	----------------------

<a href="#">"Adding Software Artifacts with the CVS Artifact Provider Plugin " below</a>	How to use an artifact provider plugin to integrate with CVS source repositories.
<a href="#">"Adding Software Artifacts with Jenkins Artifact Provider Plugin " on next page</a>	How to use an artifact provider plugin to integrate with Jenkins build systems.
<a href="#">"Adding Software Artifacts with the Server File System Artifact Provider Plugin " on page 47</a>	How to use an artifact provider plugin to integrate with a local directory, drive, or mapped drive on the HP CDA server.
<a href="#">"Adding Software Artifacts with the SFTP Artifact Provider Plugin " on page 48</a>	How to use an artifact provider plugin to integrate with a Secure FTP (SFTP) server.
<a href="#">"Adding Software Artifacts with the SVN Artifact Provider Plugin " on page 50</a>	How to use an artifact provider plugin to integrate with Subversion repositories.

## ***Adding Software Artifacts with the CVS Artifact Provider Plugin***

To use the CVS plugin to add software artifacts, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Software Artifacts** tab, and then the **Add Software Artifacts** button to open the Add Software Artifacts wizard.
  - a. On the Source Type menu, choose **CVS Artifact Provider Plugin**.
  - b. On the Provider menu that appears, choose the appropriate CVS artifact provider plug-in configuration for the CVS repository you want to use.
  - c. Click **Next**.
3. The Define Software Artifacts dialog box prompts you for the following filter options (When you apply these options, it refines which software artifacts are displayed in the artifact browser.):

Filter Option	Description
---------------	-------------

CVS Revision	Select to specify which CVS revision (branch, tag, revision number) to use for accessing the CVS Repository. The default value is HEAD.
Filter Option	Select to determine which artifacts will be shown in the browser and which will be included in the resultant software bundle at deployment time. Available options are: <ul style="list-style-type: none"><li>■ <b>Everything:</b> Do not filter any artifacts.</li><li>■ <b>Include:</b> Only artifacts that match the File Set Filter are included in the list of available artifacts and resultant software bundle at deployment time.</li><li>■ <b>Exclude:</b> Artifacts that match the File Set Filter are excluded from the list of available artifacts and from the resultant software bundle at deployment time.</li></ul>
File Set Filter	Select to set a regular expression filter to use in conjunction with either the Exclude or Include Filter option.  For example, to include only Java class files, the pattern <code>*.class</code> should be entered in this text box, with the Include option selected for the Filter Option. You can apply multiple filters by entering the set of filters as a space separated list (for example, <code>*.class *.xml</code> to include all <code>.class</code> and <code>.xml</code> files).

4. Click **Finish**.
5. Click **Apply Filter**. The filter is applied to the current browser session.

**Note:** If you do not click **Apply Filter**, the options will not be applied to the current browsing session, and they will not be saved in the resultant artifact description for the application bundle packaging when you deploy.

After the filters are set and applied, you can view the contents of the CVS repository to locate the specific directory or individual software artifacts to include in the bundle definition.

## ***Adding Software Artifacts with Jenkins Artifact Provider Plugin***

**Note:** The Jenkins artifact provider plug-in requires that the HP Application Lifecycle Intelligence (ALI) plug-in is installed and configured on the Jenkins build system. HP ALI is a

set of capabilities, reports and metrics that provide complete ALM traceability, which enables ALM stakeholders to make informed decisions.

ALI is agnostic to your organization's technologies. ALI can aggregate information from heterogeneous environments including open source and commercial tools.

To use the Jenkins plugin to add software artifacts:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Software Artifacts** tab.
5. Click the **Add Software Artifacts** button to open the Add Software Artifacts wizard. Make entry on the Select Software Source window:
  - a. **Source Type:** choose **Jenkins Artifact Provider Plugin**.
  - b. On the Provider menu that appears, choose a Jenkins Artifact Provider Plug-in configuration for the Jenkins build server you want to use.
  - c. Click **Next**.
6. The Define Software Artifacts window opens. Enter filter options to select which artifacts display in the artifact browser:
  - a. **Build Status:** Select options to determine which builds are included (successful, failed and so on).
  - b. **Build Number:** Select an option:
    - **Latest Build:** To select only the most recent build.
    - **All Builds:** To select all builds, sorted in descending order, that match the Build Status option.
  - c. **Filter Option:**

Click [here](#) for a description.

Determines which artifacts are shown in the browser and included in the resultant software bundle at deployment time. Available options are:

- **Everything:** Do not filter any artifacts.
- **Include:** Only artifacts that match the File Set Filter are included in the list of available artifacts and resultant software bundle at deployment time.
- **Exclude:** Artifacts that match the File Set Filter are excluded from the list of available artifacts and from the resultant software bundle at deployment time.

d. **File Set Filter:**

[Click here for a description.](#)

Select to set a regular expression filter to use in conjunction with either the Exclude or Include Filter option. For example, to include only Java class files, enter the pattern `*.class`, with the Include option selected for the Filter Option. You can apply multiple filters by entering the set of filters as a space separated list (for example, `*.jar *.pom` to include all `.jar` and `.pom` files).

7. Click **Apply Filter**. The filter is applied to the current browser session.

**Note:** If you do not click **Apply Filter**, the options will not be applied to the current browsing session, and they will not be saved in the resultant artifact description for the application bundle packaging when you deploy.

After the filters are set and applied, you can view the contents of the Jenkins build repository to locate the specific build job directory or individual software artifacts to include in the bundle definition.

## ***Adding Software Artifacts with the Server File System Artifact Provider Plugin***

To use the Server File System plugin to add software artifacts, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Software Artifacts** tab, and then the **Add Software Artifacts** button to open the Add Software Artifacts wizard.

- a. In the Source Type menu, choose **Server File System Artifact Provider Plugin**.
  - b. The Provider menu appears. Choose the appropriate Server File System artifact provider plug-in configuration.
5. Click **Next** to open the Define Software Artifacts wizard window, and enter the following:

**Filter Option:** Use this option to select what artifacts show in the browser and are included in software bundle deployment.

[Click here for a description.](#)

Determines which artifacts are shown in the browser and included in the resultant software bundle at deployment time. Available options are:

- **Everything:** Do not filter any artifacts.
- **Include:** Only artifacts that match the File Set Filter are included in the list of available artifacts and resultant software bundle at deployment time.
- **Exclude:** Artifacts that match the File Set Filter are excluded from the list of available artifacts and from the resultant software bundle at deployment time.

**File Set Filter:** Set a regular expression filter to use with Exclude or Include (Filter Options).

[Click here for a description.](#)

Select to set a regular expression filter to use in conjunction with either the Exclude or Include Filter option. For example, to include only Java class files, enter the pattern `*.class`, with the Include option selected for the Filter Option. You can apply multiple filters by entering the set of filters as a space separated list (for example, `*.jar *.pom` to include all `.jar` and `.pom` files).

6. Click **Apply Filter**. The filter is applied to the current browser session.

**Note:** If you do not click **Apply Filter**, the options will not be applied to the current browsing session, and they will not be saved in the resultant artifact description for the application bundle packaging when you deploy.

7. View the contents of the directory specified by the Root/Parent Directory Server File System plugin configuration parameter in the browser to locate the directory or individual software artifact file to include in the bundle definition.
8. Select the desired directory or file to be added to the software bundle, and then click **Finish**.

## ***Adding Software Artifacts with the SFTP Artifact Provider Plugin***

To use the SFTP plugin to add software artifacts, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Software Artifacts** tab, and then the **Add Software Artifacts** button to open the Add Software Artifacts wizard.

- a. In the Source Type menu, choose **SFTP Artifact Provider Plugin**.
- b. The Provider menu appears. Choose the appropriate SFTP artifact provider plug-in configuration.

5. Click **Next** to open the Define Software Artifacts wizard window, and enter the following:

**Filter Option:** Use this option to select what artifacts show in the browser and are included in software bundle deployment.

[Click here for a description.](#)

Determines which artifacts are shown in the browser and included in the resultant software bundle at deployment time. Available options are:

- **Everything:** Do not filter any artifacts.
- **Include:** Only artifacts that match the File Set Filter are included in the list of available artifacts and resultant software bundle at deployment time.
- **Exclude:** Artifacts that match the File Set Filter are excluded from the list of available artifacts and from the resultant software bundle at deployment time.

**File Set Filter:** Set a regular expression filter to use with Exclude or Include (Filter Options).

[Click here for a description.](#)

Select to set a regular expression filter to use in conjunction with either the Exclude or Include Filter option. For example, to include only Java class files, enter the pattern `*.class`, with the Include option selected for the Filter Option. You can apply multiple filters by entering the set of filters as a space separated list (for example, `*.jar *.pom` to include all `.jar` and `.pom` files).

6. Click **Apply Filter**. The filter is applied to the current browser session.

**Note:** If you do not click **Apply Filter**, the options will not be applied to the current browsing session, and they will not be saved in the resultant artifact description for the application bundle packaging when you deploy.

7. View the contents of the directory specified by the Root/Parent Directory SFTP plugin configuration parameter in the browser to locate the directory or individual software artifact file to include in the bundle definition.
8. Select the desired directory or file to be added to the software bundle, and then click **Finish**.

## ***Adding Software Artifacts with the SVN Artifact Provider Plugin***

To use the SVN plugin to add software artifacts, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Software Artifacts** tab, and then the **Add Software Artifacts** button to open the Add Software Artifacts wizard.
  - a. On the Source Type menu, choose **SVN Artifact Provider Plugin**.
  - b. The Provider menu appears. Choose the appropriate SVN artifact provider plug-in configuration.
5. Click **Next** to open the Define Software Artifacts wizard window, and enter the following:

**Revision:** Type a specific revision number to select artifacts based on that revision, or type **HEAD** to retrieve the latest revision.

**Filter Option:** Use this option to select what artifacts show in the browser and are included in software bundle deployment.

[Click here for a description.](#)

Determines which artifacts are shown in the browser and included in the resultant software bundle at deployment time. Available options are:

- **Everything:** Do not filter any artifacts.
- **Include:** Only artifacts that match the File Set Filter are included in the list of available artifacts and resultant software bundle at deployment time.
- **Exclude:** Artifacts that match the File Set Filter are excluded from the list of available artifacts and from the resultant software bundle at deployment time.

**File Set Filter:** Set a regular expression filter to use with Exclude or Include (Filter Options).

[Click here for a description.](#)

Select to set a regular expression filter to use in conjunction with either the Exclude or Include Filter option. For example, to include only Java class files, enter the pattern `*.class`, with the Include option selected for the Filter Option. You can apply multiple filters by entering the set of filters as a space separated list (for example, `*.jar *.pom` to include all `.jar` and `.pom` files).

6. Click **Apply Filter**. The filter is applied to the current browser session.

**Note:** If you do not click **Apply Filter**, the options will not be applied to the current browsing session, and they will not be saved in the resultant artifact description for the application bundle packaging when you deploy.

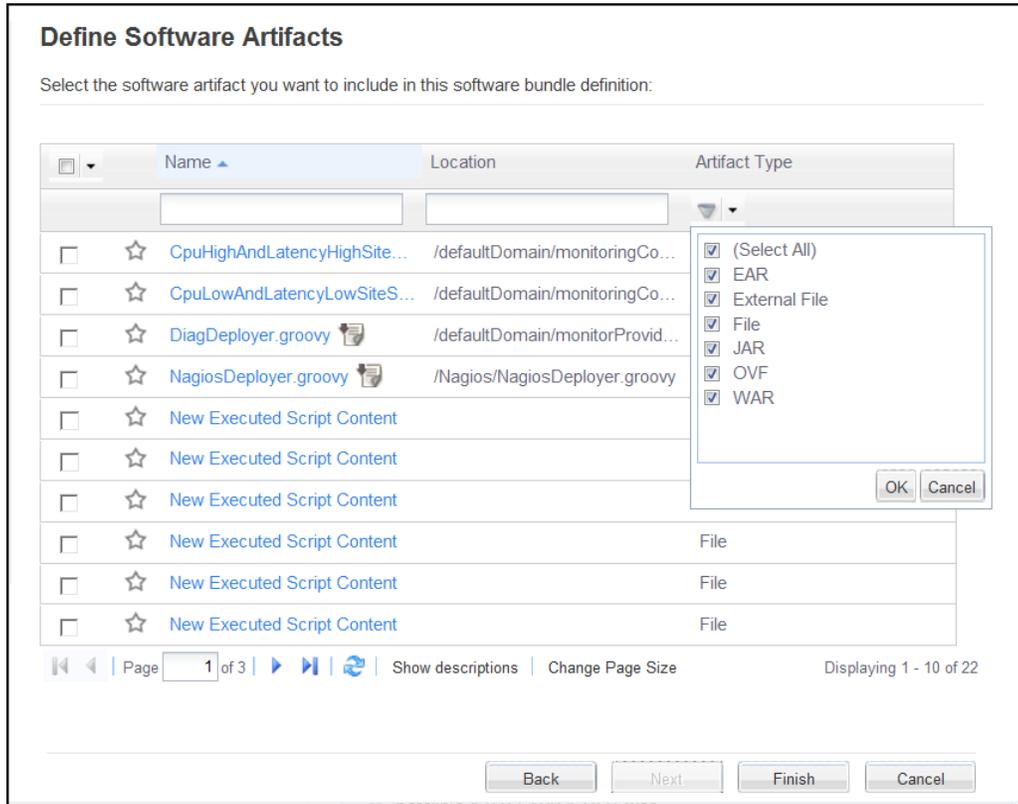
After the filters are set and applied, you can view the contents of the SVN build repository to locate the specific build job directory or individual software artifacts to include in the bundle definition.

## Selecting an Existing File from DSL

To reconfigure a software artifact by using an existing image from the DSL, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. On the Overview tab, click a model name to open an application model details window.
5. Click the version name to open a details window.
6. Click the **Software Artifacts** tab.

7. Click **Add Software Artifacts** and choose **Select Existing Files(s) from DSL**.
8. Click **Next**.
9. Select each artifact to include in the software bundle definition. Use the empty text boxes at the top of each column to filter by artifact name or location.



10. Click **Finish**.

## Uploading Software Artifacts to DSL

To reconfigure a software artifact using an existing file:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA

3. On the Simple view, click a version name to open an application version details window.
4. On the Overview tab, click a model name to open an application model details window.
5. Click the version name to open a details window.
6. Click the **Software Artifacts** tab.
7. Click **Add Software Artifacts** and choose Upload File to DSL from the menu.
8. Click **Next**.
9. Select **File** or **URL**. Select either the file to upload, or specify the URL of the file location.
10. Click **Finish**.

## ***Downloading Software Artifacts from an External URL***

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

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Software Artifacts** tab.

**Note:** When you add Software artifacts to a bundle definition, the software artifacts are automatically added to all the definitions.

5. Click **Add Software Artifacts** and choose **Download File from External URL** on the menu.
6. *Optional:* Select **Cache copy in CDA library**. The first time HP CDA runs a deployment, a copy of the bundle artifact will be downloaded from the URL and saved into the CDA system. This option allows you to reuse that file in subsequent executions of the application workflow that use the bundle artifact: it makes the file more readily available. Note that the artifact file used here is the one specified in the URL, not the saved copy.
7. Click **Next**.
8. Type the external URL address for the software artifact and click **Finish**.

## Deleting Bundle Definitions

To delete a bundle definition, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Software Artifacts** tab to open a window where you see bundle definitions.
5. Select the definition that you want to delete and click **Delete**.
6. Click **OK** to confirm.

## Adding, Cloning, Editing and Deleting Deployment Topologies

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.

The following topics provide information about working with deployment topologies.

Topic	Information Included
<a href="#">"Adding a Deployment Topology " on next page</a>	How to add a deployment topology.
<a href="#">"Cloning a Deployment Topology " on next page</a>	How to clone a deployment topology.
<a href="#">"Editing a Deployment Topology " on page 56</a>	How to edit a deployment topology.
<a href="#">"Validating a Deployment Topology " on page 56</a>	Information about validation framework checks performed by HP CDA. Includes information about how a deployment topology becomes invalid and what to do.
<a href="#">"Deleting a Deployment Topology " on page 58</a>	How to delete a deployment topology.

## Adding a Deployment Topology

In HP CDA, the topology is required to map the application model 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. Navigate to the Application Model view and click the **Deployment Topologies** tab.
2. Click **Add New Topology** to open the Create topology wizard.
  - a. Read the Introduction and click **Next** to open the Basic Properties window.
  - b. On the Basic Properties window, enter the name of the topology and specify a stage, such as Development, QA, Staging, or Production. Click **Next**.
  - c. On the Platform window, you see platforms that fulfill your layer requirements and stage. Click **Next**.
  - d. On the Monitoring Policies window, add Monitoring policies as required. For information on how to define monitoring policies and monitoring providers, and how to assign monitoring policies to application topologies, see "[Monitoring Administration](#)" on page 139. Click **Next**.
  - e. On the Summary window, review your entry. Click **Finish** to save the 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 and select the **Deployment Topologies** tab.
2. Select an application model and then click **Clone**.
3. 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.

## Editing a Deployment Topology

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

To edit a deployment topology:

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

## Validating a Deployment Topology

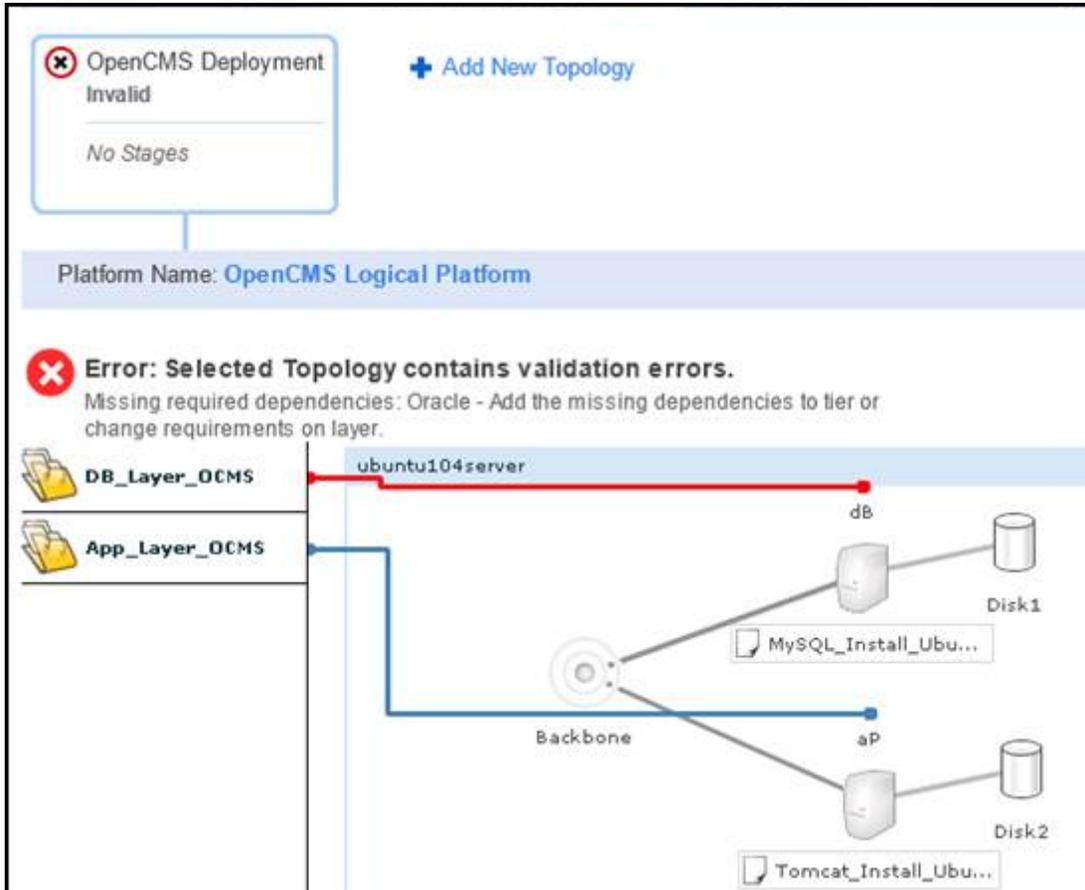
HP CDA performs validation framework checks of deployment topologies and shows the outcome on the Deployment Topologies tab.

Valid topologies display blue layer lines and a "Ready for Use" status.

Invalid topologies display a red layer line in the invalid layer and an "Invalid" status. You also see information about the problem, such as, "Missing required dependencies: Oracle - Add the missing dependencies to tier or change requirements on layer."

When you create a deployment topology, its state is always valid, because the Create Topology wizard offers only valid platforms. At the time of creation, layer requirements and platform capabilities match. However, over time, users can change either requirements or capabilities—and create a mismatch.

Because of the potential to invalidate topologies, each time a user changes requirements or capabilities, HP CDA checks to determine whether topologies remain valid. The following graphic shows an invalid topology.



For information about platform modifications that can invalidate a topology, see:

- ["Editing, Adding and Deleting Infrastructure Templates " on page 84](#)
- ["Modifying Platform Hardware and Operating System Capabilities " on page 88](#)
- ["Modifying Platform Software Capabilities " on page 85](#)

For information about layer modifications that can invalidate a topology, see ["Defining an Application Layer " on page 26.](#)

You can attempt to deploy an invalid topology, if you believe the problem will not impact deploying or running your application. HP CDA does not prevent the attempted deployment. However, as a best practice, you should maintain accurate requirement and capability information.

For example, you have an application that requires Apache Tomcat 6 and you plan to deploy the application to a platform that runs TomCat 7. The topology appears invalid because of the mismatch, but you could successfully deploy. In this case, you should update your requirement to be **TomCat v6 or later**. Your topology then appears with an accurate Valid status.

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

## Adding Input Component Parameters to Levels

In HP CDA, you can add input components at different levels of the application model. As a best practice, add these parameters so that you can reuse them for multiple deployments.

The following topics provide information about where and how to add input component parameters.

Topic	Information Included
<a href="#">"Adding a Parameter Definitions Group to an Application Model "</a> below	How to add input component parameters at the top of the application model.
<a href="#">"Adding an Input Component Parameter to a Layer "</a> on next page	How to add input component parameters at the layer level, such as the DB Layer, App Server Layer, and so on.
<a href="#">"Adding an Input Component Parameter to a Workflow "</a> on next page	How to add input component parameters at the workflow action level, such as in the configure mysql script.

## Adding a Parameter Definitions Group to an Application Model

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

1. Go to the Application Model view and select the **Parameters** tab.
2. In the Parameter Definitions Group view, click **+ Add**.

3. Complete the Add dialog.
4. Click **Add** to save the parameter.

## Adding an Input Component Parameter to a Layer

To add an input component parameter to a layer:

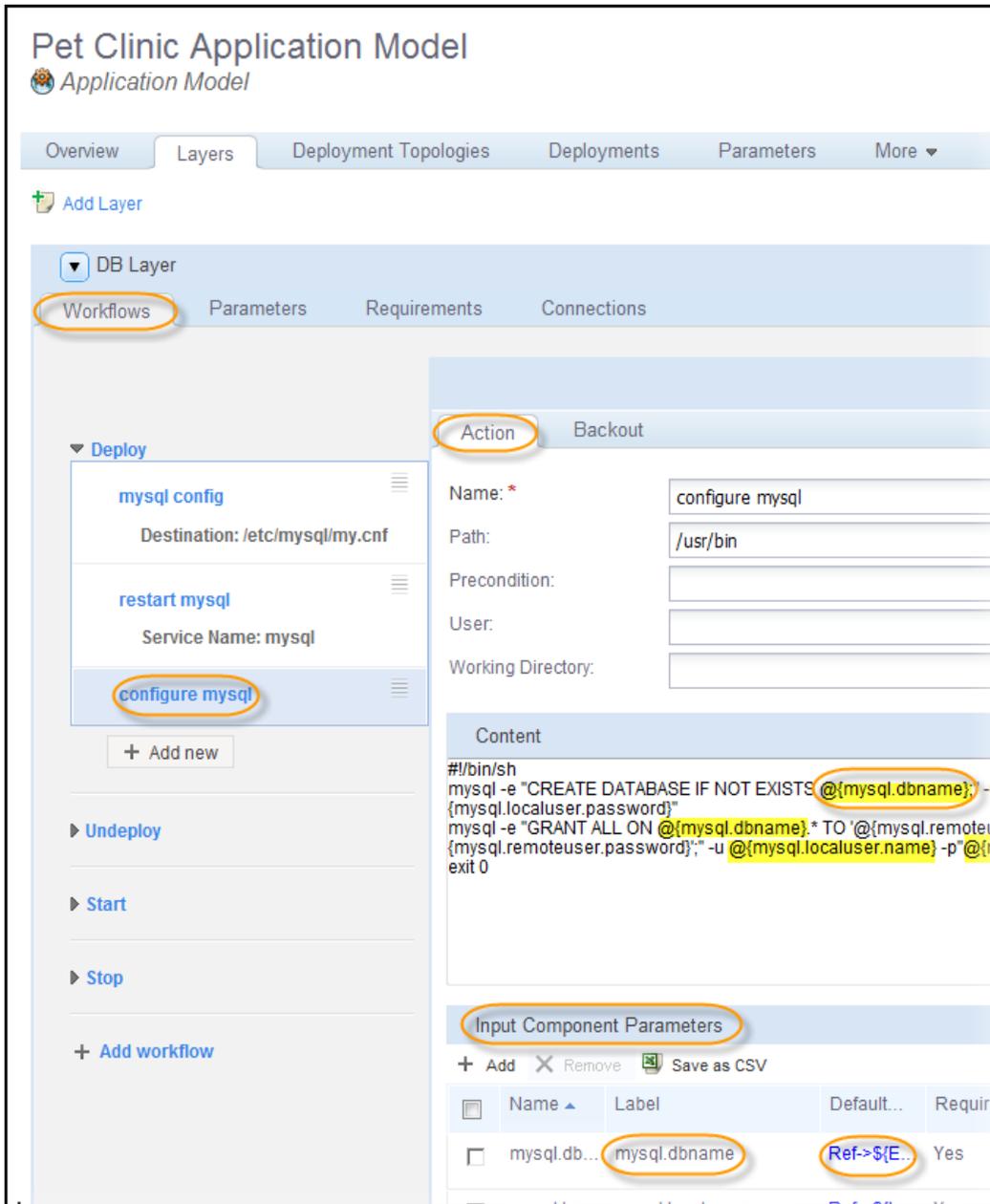
1. Navigate to the Application Model view and select the **Layers** tab.
2. Select the **Layers** tab.
3. Expand a layer and then select its **Parameters** tab.
4. In the Parameter Definitions Group view, click **+ Add**.
5. Complete the Add dialog.
6. Click **Add** to save the parameter.

## Adding an Input Component Parameter to a Workflow

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

To add an input component parameter to a workflow:

1. Navigate to the Application Model view and select the **Layers** tab.
2. Expand a layer and then select its **Workflow** tab.
3. Expand a workflow action, such as Deploy, Undeploy, Start, or Stop.
4. In the Action tab Input Component Parameters view, click **+ Add**.
5. Complete the Add dialog.
6. Click **Add** to save the parameter.



## Adding and Deleting Connections and 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.

Topic	Information Included
<a href="#">"Adding and Deleting Connections " below</a>	How to add and delete connections.
<a href="#">"Creating, Modifying and Deleting Endpoints " below</a>	How to create, modify and delete endpoints.

## Adding 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. Navigate to the Application Model view and select the Layers tab.
2. Expand a layer.
3. Click the **Connections** tab.
4. Expand Connection(s).
5. *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.
6. *Optional:* To add an endpoint, click **Add Connection**.

## Creating, Modifying and 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. Navigate to the Application Model view and click the **Layers** tab.
2. Expand a layer.
3. Click the **Connections** tab.
4. Expand **Endpoint(s)**.
5. *Optional:* To edit or remove an endpoint that has already been defined, select the endpoint and then click **Edit** or **Remove**.
6. *Optional:* To create a new endpoint, click **Create New Endpoint**.
7. On the Add dialog box, type a name and the port.
8. *Optional:* Select **Load Balanceable** to specify that the layer will be load balanced. By default, this option is not selected.

If you choose load balancing by selecting the **Load Balanceable** checkbox, you must enter load balancer parameter values in the deployment topology map.

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

## Deploying and Redeploying Applications

If an application has not been deployed, it can be deployed via a *deployment* operation. Once a deployed instance of the application exists, it can be *redeployed*. Redeploy workflows can also be run just on application layers or realized servers that host a deployed application as well.

### Deployment of an Application

Applications are deployed when a *deployed instance* of the application does not exist. This is sometimes referred to as the *initial deployment* of the application. You can deploy an application after the following conditions are met:

- A platform has been modeled and provisioned
- The application has been modeled
- The application's layers have been mapped to the platform tiers

The following sequence occurs during the application deployment process:

- The application is packaged. During packaging, a bundle that contains all the artifacts that are configured on the application Software Artifacts tab are accessed and made available to the deployment process.

- The Deploy application workflows for the following layers in the application model are executed:
  - Layers that are included in enabled application options.
  - Layers that are not included in any application options.
- Monitors are deployed if monitoring policies had been applied to respective application layers in the deployment topology.

**Note:** If the application does not contain separately deployable application options, then the Deploy workflows for all layers included in the application model are executed.

If the deployment operation is successful, a deployed instance of the application is created, which is to say that the application artifacts have been deployed onto the realized platform, the application has been properly configured, and it can now be run.

## Redeployment of an Application

Redeployment allows you to perform certain operations on an application that has already been deployed, such as redeploying software artifacts that have been modified, or performing *incremental deployment*. Incremental deployment allows you to deploy application options in stages, for example, a subset of application options can be deployed initially, and additional options can be deployed at a later time once they have been fully developed, tested and are available for deployment. During redeployment, application options that are already deployed can be redeployed and undeployed as well. Redeployment of an application can be performed multiple times.

**Note:** Redeployment in the sense described in this section uses the Redeploy Application wizard and performs redeployment at the application level. For certain operations such as deploying minor patches to a deployed application, full application redeployment might not be necessary, and running the Redeploy workflows at an application layer or realized server group level might suffice. Refer to the topic ["How to Execute Redeploy Workflows " on page 71](#) for more information.

The sequence that occurs during the application redeployment process is similar to the application deployment process, with the following differences:

- The application is packaged as during deployment, but during redeployment, the user can choose a different bundle definition and bundle to deploy. This is useful, for example, for deploying major patches or fixes to an already deployed application.
- Application workflows are executed as follows:
  - The Deploy workflow is executed for application layers that had not previously been deployed but are selected for deployment in the Redeploy Application wizard .
  - The Redeploy workflow is executed for application layers that had previously been deployed and are selected for deployment in the Redeploy Application wizard.

- The Undeploy workflow is executed for application layers that had previously been deployed but are not selected for deployment in the Redeploy Application wizard.

**Note:** The Redeploy workflows for all application layers that are not included in any application options are also executed during redeployment.

The following topics provide information about application deployment and redeployment.

Topic	Information Included
<a href="#">"How to Deploy an Application " below</a>	How to make the required Deploy Application wizard entries to deploy an application.
<a href="#">"How to Redeploy an Application " on page 67</a>	How to make the required Redeploy Application wizard entries to redeploy an application.
<a href="#">"How to Execute Redeploy Workflows " on page 71</a>	How to execute the Redeploy workflows for application layers or realized server groups of a deployed application.
<a href="#">"Opening a Deployed Application Window " on page 71</a>	How to open a window where you can see details of a deployment.
<a href="#">"Setting Start and Stop Workflows for Deployment " on page 72</a>	How to include Start and Stop workflows on all deployed servers or on an individual server
<a href="#">"Undeploying an Application " on page 72</a>	How to undeploy an application.
<a href="#">"Deploying Local Developer Builds " on page 73</a>	How to bundle artifacts that are stored on a system (such as a software developer's local system) and deploy them.

## How to Deploy an Application

To deploy an application, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.

4. *Optional*: Click a model name to open an application model details window.
5. Click the **Overview** tab, if it is not already selected, and click **Deploy** on the right to open the Deploy Application wizard.
6. Make the following entries in the Application Artifacts window of the Deploy Application wizard:
  - a. **Deployment Name**: Type a screen name for the application deployment.
  - b. **Bundle Definition**: Select the bundle definition that you want to deploy in this drop-down field.

If a bundle definition has been selected as the default bundle on the Software Artifacts tab, that bundle definition will be initially displayed in this field.

- c. **Bundle to be deployed**: By default, LATEST will be displayed in this field, indicating that a new bundle will be created from the selected package definition when the application is deployed, and the bundle will subsequently be available on the Bundles tab. You can also click the Select an Existing Bundle link, select a pre-existing bundle in the Select Artifact dialog box that appears, and then click **Select**. Only the bundles related to the selected bundle definition will be shown in the Select Artifact dialog box.

**Note**: The artifacts that are contained in the bundle that has been selected are listed in the table that is below the Bundle to be deployed field, like what is shown in the following figure:

Artifacts in Bundle Definition : Default Bundle Definition				
Artifact ▲	Relative URI	Version	Provider	Provider Type
build.sh	IBTPortal/buil...	LATEST	SVN	SVN
<a href="http://jenkins-blr.blr.adam.qa.hp.com/svn/IBTPortal/build.sh">http://jenkins-blr.blr.adam.qa.hp.com/svn/IBTPortal/build.sh</a>				
CDA 1.21_QA...		LATEST		
clicommands.txt		LATEST		
confidential.txt	./confidential.txt	LATEST	SFTP	SFTP Artifact ...

Artifacts shown in the table with no provider information listed might be from different sources, such as artifacts that are downloaded directly from an external URL or artifacts that are stored in the Definitive Software Library (DSL).

7. Click the **Next** button.
8. If application options have been defined for the application, select the application options that you want to deploy in the Application Options window, and then click the **Next** button.
9. Make the following entries in the Deploy Settings window, and then click the **Next** button:
  - a. **Application Model**: Select the application model to be deployed in this drop-down field.

**Note:** If application options have been selected in this procedure, only application models in which those application options have been created will be available for selection in the Application Model drop-down field.

- b. **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.

**Note:** If application options have been selected in this procedure, only deployment topologies in which all of the application option's layers have been mapped will be available for selection in the Deployment Topology drop-down field, as topologies containing unmapped application option layers are not valid topologies. Layers that are not contained in any application options, however, can remain unmapped in a topology and such unmapped layers will be shown as skipped in the deployment report.

- c. **Provisioned Platform**: Select the provisioned platform onto which the application is to be deployed in this drop-down field.
    - d. **Force Automatic Backout and cleanup on any failure**: Select this option if you want a backout operation to begin automatically if the deployment operation fails.
    - e. **Advanced Options**:

**Note:** HP strongly recommends that you enter a job timeout limit of approximately how long you expect deployment to take. Otherwise, errors in deployment can cause the operation to hang indefinitely.

Select **Job Timeout**, and type a time and unit of measurement. If deployment takes longer than the timeout limit, the operation cancels out.

Select **Backout Timeout**, and type a time and unit of measurement. If a backout operation takes longer than the timeout limit, the operation cancels out.

You can use the deployment Report to confirm that a timeout is the reason an operation failed.

10. If editable parameters appear in the Application Parameters window, enter or modify the

parameter values either manually or load a stored parameter set to fill in the parameter values, and then click the **Next** button.

**Note:** Parameters vary, depending on what parameters were designed into the application and its related artifacts. If an application developer defined a parameter as **user-exposed**, you can edit it. If an application developer defined the parameter as **private-scope**, you can see the parameter, but you cannot edit it.

11. Make any necessary changes in the Monitor Parameters window, and then click the **Next** button.
12. A Summary window opens. The summary window lists, among other things, which application options are and are not selected for deployment. Review the summary and then click **Finish** to begin the application deployment process.

## How to Redeploy an Application

To redeploy an application, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Applications menu, choose either:
  - **My Applications** to see a list of all applications that are owned by the user who is logged in
  - **All Applications** to see a list of all applications in HP CDA
3. On the Simple view, click a version name to open an application version details window.
4. Click the **Deployed Instances** tab, if it is not already selected, and then click the name of the deployment that you want to redeploy.
5. *Optional:* Click the name of the deployment under the Deployment heading on the Application Version screen for the deployment that you want to redeploy.
6. Click **Redeploy** on the right to open the Redeploy Application wizard.
7. Make the following entries in the Application Artifacts window of the Redeploy Application wizard:

**Note:** You cannot change the name of the deployment in the Redeploy Application wizard.

- a. **Bundle Definition:** Select the bundle definition that you want to deploy in this drop-down field, or leave it as-is if you want to redeploy the same bundle.

If a bundle definition has been selected as the default bundle on the Software Artifacts tab, that bundle definition will be initially displayed in this field.

- b. **Bundle to be deployed:** By default, LATEST will be displayed in this field, indicating that a new bundle will be created from the selected package definition when the application is deployed, and the bundle will subsequently be available on the Bundles tab. You can also click the Select an Existing Bundle link, select a pre-existing bundle in the Select Artifact dialog box that appears, and then click **Select**. Only the bundles related to the selected bundle definition will be shown in the Select Artifact dialog box.

**Note:** The artifacts that are contained in the bundle that has been selected are listed in the table that is below the Bundle to be deployed field, like what is shown in the following figure:

Artifacts in Bundle Definition : Default Bundle Definition

Artifact ▲	Relative URI	Version	Provider	Provider Type
build.sh	IBTPortal/buil...	LATEST	SVN	SVN
<i>http://jenkins-blr.blr.adam.qa.hp.com/svn/IBTPortal/build.sh</i>				
CDA 1.21_QA...		LATEST		
clicommands.txt		LATEST		
confidential.txt	./confidential.txt	LATEST	SFTP	SFTP Artifact ...

Artifacts shown in the table with no provider information listed might be from different sources, such as artifacts that are downloaded directly from an external URL or artifacts that are stored in the Definitive Software Library (DSL).

8. Click the **Next** button.
9. If application options have been defined for the application, select the application options that you want to deploy, redeploy, or undeploy in the Application Options window, as follows:

**Note:** In this window, checkboxes for the options that are currently deployed are selected and the checkboxes for the options that are not currently deployed are not selected when the window opens.

- To redeploy a deployed application option, leave the checkbox selected
- To undeploy a deployed application option, deselect the checkbox

- To deploy a undeployed application option, select the checkbox
- To leave an application option undeployed, leave the checkbox deselected

10. Click the **Next** button.
11. Make the following entries in the Redeploy Settings window if necessary or leave them as-is, and then click the **Next** button:

**Note:** The following fields will contain the selections that are applicable to the current deployment when the Redeploy Settings window opens.

- a. **Application Model:** Select the application model to be deployed in this drop-down field.

**Note:** If application options have been selected in this procedure, application models in which those application options have been created will be available for selection in the Application Model drop-down field.

- b. **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.

**Note:** If application options have been selected in this procedure, only deployment topologies in which all of the application option's layers have been mapped will be available for selection in the Deployment Topology drop-down field, as topologies containing unmapped application option layers are not valid topologies. Layers that are not contained in any application options, however, can remain unmapped in a topology and such unmapped layers will be shown as skipped in the deployment report.

- c. **Provisioned Platform:** Select the provisioned platform onto which the application is to be deployed in this drop-down field.
- d. **Force Automatic Backout and cleanup on any failure:** Select this option if you want a backout operation to begin automatically if the redeployment operation fails.
- e. **Advanced Options:**

**Note:** HP strongly recommends that you enter a job timeout limit of approximately how long you expect redeployment to take. Otherwise, errors in redeployment can cause the operation to hang indefinitely.

Select **Job Timeout**, and type a time and unit of measurement. If redeployment takes longer than the timeout limit, the operation cancels out.

Select **Backout Timeout**, and type a time and unit of measurement. If a backout operation takes longer than the timeout limit, the operation cancels out.

You can use the Redeployment Report to confirm that a timeout is the reason an operation failed.

12. If editable parameters appear in the Application Parameters window, enter or modify the parameter values either manually or load a stored parameter set to fill in the parameter values, and then click the **Next** button.

**Note:** Parameters vary, depending on what parameters were designed into the application and its related artifacts. If an application developer defined a parameter as **user-exposed**, you can edit it. If an application developer defined the parameter as **private-scope**, you can see the parameter, but you cannot edit it.

13. Make any necessary changes in the Monitor Parameters window, and then click the **Next** button.
14. A Redeployment Summary window opens. The summary window lists, among other things, which application options are and are not selected for deployment, redeployment, and undeployment. Review the summary and then click **Finish** to begin the application redeployment process.

**Note:** An application model can contain *optional layers*, which need not be mapped to a tier in a given deployment topology. Such unmapped layers will not be executed during deployment or redeployment.

Two states are displayed on the Overview tab on the Application Deployment screen for a deployed application:

- **Deployment State:** This is the current state of the deployed application. If this state is not DEPLOYED, then the application cannot be redeployed.
- **Redeployment State:** Once an application redeployment has been attempted, this state will be one of the following:
  - **SUCCESSFUL:** The redeployment operation was successful and the application can be run successfully.
  - **FAILED :** The redeployment operation has failed. When the redeployment operation fails, you will not be able to execute redeployment again. This is to prevent any inconsistencies that can happen when a redeployment operation is attempted on a failed redeployment. You will need to perform a backout operation after the failure before attempting to perform redeployment again. During a backout operation, the appropriate backout steps get executed depending on whether the corresponding layer belongs to an application option that is deployed, redeployed or undeployed.

- BACKOUT : The failed redeployment operation has been successfully backed out. You can perform a redeployment operation when the redeployment is in this state.

- CANCELLED : The redeployment operation has been canceled. You can perform a redeployment operation when the redeployment is in this state since a cancel operation will execute a backout operation, thus leaving the system in a clean state.

## How to Execute Redeploy Workflows

For certain operations such as deploying minor patches or changing configuration parameters for a deployed application, full application redeployment might not be necessary, and running the Redeploy workflows at a realized application layer or realized server group level might suffice.

Redeploy links are available on the Application Deployment screen for every realized layer and server group of a given deployed application, if Redeploy workflows for those entities have been created. An example of such Redeploy links is shown in the following figure:



When you click the Redeploy link for the respective layer or server group, the programming operations for that Redeploy workflow are executed. There is no wizard displayed at that time, hence you cannot change the artifact bundle, select application options, etc. This operation is similar to how start/stop or any custom workflows are executed.

**Note:** Imported HP CDA application models that were developed prior to HP CDA software version 1.30 will not contain Redeploy workflows unless you add them manually as custom (user-defined) workflows. Whenever you add layers to an application model, however, an empty Redeploy workflow is included automatically.

## Opening a Deployed Application Window

To open a window that lists deployed applications, complete the following steps:

1. Click the **Applications** tab to open the Applications Home window.
2. On the Deployments menu, choose either:
  - My Deployments** to open a window that lists deployments owned by the user who is signed in.
  - All Deployments** to open a window that lists of all deployments in HP CDA.

The My Deployments and All Deployments windows show:

- **Name:** The application deployment name that is assigned during deployment.
- **Application:** The application that is associated with the deployed version.
- **Application Version:** The version of the application that was deployed.
- **Bundle:** The bundle that was created by the application deployment or that was specified as an existing package when the application was deployed.
- **Stage:** The deployed application's current development lifecycle stage.
- **Status:** Whether the application deployment was successful or failed.

## Setting Start and Stop Workflows for Deployment

When you model an application, you have the option of adding Start and Stop workflows in each of its layers. If you add those workflow types, you see them on the Application Deployment window for the application that was deployed from the model.

Click **Start** or **Stop** that is next to an application layer on the Application Deployment window to execute the Start or Stop workflow on only that 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 **Start** or **Stop** on the right side of the window to run the Start or Stop workflows on all deployed servers that are part of the application model.

## Undeploying an Application

To undeploy an application, complete the following steps:

1. Click the **Applications** tab to open the Applications home window.
2. On the Deployments menu, choose one of the following:

- **My Deployments:** to see a list of deployed applications that are owned by the user who is logged in
  - **All Deployments:** to see a list of all deployed applications
3. Click the deployed application name to open the Application Deployment window.
  4. *Optional:* Click the **Overview** tab , if it is not already selected.
  5. Click **Undeploy** on the right side of the window.
  6. Click **OK** to start the undeploy operation.

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

**Note:** After an undeploy operation starts, you cannot cancel it. On the Report window, you can initiate a manual backout operation if the undeploy operation fails and forced automatic backout is not enabled for the operation.

## Deploying Local Developer Builds

You can use HP CDA to bundle artifacts that are stored on a system such as a software developer's local system and deploy them. You must copy some files from the HP CDA server machine to the software development system and modify them to accomplish this task. The following sections explain how to do this.

## Running CLI Commands Remotely

This procedure configures a computer system so that the cdaexec script can be run locally on it, thus allowing HP CDA CLI commands to be run from a system other than the system on which HP CDA is installed.

**Note:** The system on which you want to run CLI commands must be running a version of the JDK that is supported for HP CDA. Refer to the HP CDA Support Matrix for information about supported versions of the JDK.

1. Copy the file named "client.trustore" that is in the <cda\_home\_directory>\client\conf directory to a directory on the system from which you want to run HP CDA CLI commands (referred to throughout the remainder of this procedure as the local client machine).

This “client.trustore” file contains the server certificate within the trustore and is required for SSL communication with HP CDA.

2. Copy the files named “hp-adam-cli-xxxx.jar” and “sample-client-deployment-1.0.0.zip” that are in the <cda\_home\_directory>\client\deploy directory to a directory on the local client machine.
3. Unzip the “sample-client-deployment-1.0.0.zip” file to a directory on the local client machine.

**Note:** The “sample-client-deployment-1.0.0.zip” file contains the cdaexec script as well as other files and scripts. One script in the zip file, named “deploy,” is explained further in ["Configuring the Deploy Script "](#) below.

4. After unzipping the contents of the zip file to the local client machine, open the “cdaexec.bat” file that is in the unzipped directory, make the following modifications to the file contents, and then save the “cdaexec.bat” file:
  - Edit the TRUST\_STORE variable to point to the directory where you copied the “client.trustore” file in Step 1.
  - Edit the java classpath (-cp) parameter to point to the directory where you copied the “hp-adam-cli-xxxx.jar” file in Step 2.
5. Edit “cliforcda.properties” file that is in the unzipped directory so that the server, user, and password parameters contained in it reflect the HP CDA instance for which you want to remotely run CLI commands.

The cdaexec script can now be run on the local client machine.

## ***Configuring the Deploy Script***

The “sample-client-deployment-1.0.0.zip” contains a script named “deploy” which can be used to bundle artifacts that are stored on the local client machine and then deploy them using HP CDA. To use the deploy script, you must modify its contents to replace certain UUIDs, the file path or name of the artifacts to be packaged, and the deployment name with values that are relevant to the HP CDA system and the deployment.

Use the model list CLI commands to obtain the relevant UUIDs. For information on the available model list commands, see ["HP CDA Command Line Interface \(CLI\) "](#) on page 276.

**Note:** To use the model list commands or run the deploy script on the local client machine, the cdaexec script must be configured to run on the local client machine. For details, see ["Running CLI Commands Remotely "](#) on previous page.

# Platforms

A platform is an environment upon which you can deploy one-to-many applications, each with a different topology.

**Note:** The same topology can be deployed only once per realized platform.

Platforms include several elements:

- Platform Provisioning Plugins: to provide the architectural element that allows HP CDA to extend and integrate with infrastructure providers.
- Plugin Configurations: to define an instance of an infrastructure provider.
- One or more infrastructure templates that are suited to the needs of your intended applications. You can add multiple templates to a single platform to establish hybrid provisioning across different providers, or you can use multiple templates from the same provider.
- If your infrastructure templates do not specify all the software capabilities your infrastructure needs, you can model additional (add-on) software and add it to your platform model.

When your platform model is complete, you can provision it to create a realized platform. Then you can deploy applications to the platform.

At the end of a platform's life, you deprovision it. This step is important, because it frees up resources. During the deprovisioning process, add-on software that was added during provisioning will be undeployed before deprovisioning of the platform. This option of undeploying platform software is specified by the platform designer under the deployer tab of each platform tier.

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.

The following topics provide information about tasks platform developers perform when working with platforms.

Topic	Information Included
<a href="#">"Using Platform Models " on the facing page</a>	How to create, modify, publish and delete a platform.
<a href="#">"Using Add-on Platform Software " on page 104</a>	How to create add-on platform software for platforms. Includes information about workflows and programming actions.
<a href="#">"Provisioning and Deprovisioning Platforms " on page 118</a>	How to provision, cancel, back out and deprovision.

## Using Platform Models

A platform is an environment upon which you can deploy one-to-many applications, each with a different topology.

**Note:** The same topology can be deployed only once per realized platform.

Platforms include several elements:

- Platform Provisioning Plugins: to provide the architectural element that allows HP CDA to extend and integrate with infrastructure providers.
- Plugin Configurations: to define an instance of an infrastructure provider.
- One or more infrastructure templates that are suited to the needs of your intended applications. You can add multiple templates to a single platform to establish hybrid provisioning across different providers, or you can use multiple templates from the same provider.
- If your infrastructure templates do not specify all the software capabilities your infrastructure needs, you can model additional (add-on) software and add it to your platform model.

When your platform model is complete, you can provision it to create a realized platform. Then you can deploy applications to the platform.

At the end of a platform's life, you deprovision it. This step is important, because it frees up resources. During the deprovisioning process, add-on software that was added during provisioning will be undeployed before deprovisioning of the platform. This option of undeploying platform software is specified by the platform designer under the deployer tab of each platform tier.

A platform model is composed of, and defines, one or more tiers. Each tier is a group of one or more servers that contain the same OS and software, and reside in a common network configuration.

After you create a platform model, you can add to it and modify it. When you are satisfied that a platform meets the needs of the applications that you plan to deploy on it, you can provision the platform.

At the end of the platform model's life, you deprovision and delete it.

The following topics provide information about tasks platform developers perform when working with platform models.

Topic	Information Included
<a href="#">"Opening a List of Platform Models " on next page</a>	How to open a list of platform models that you own, or a list of all platforms in HP CDA.
<a href="#">"Opening a Platform Model Details Window " on next page</a>	How to find a platform model and see its details.

Topic	Information Included
<a href="#">"Creating a Platform Model " on page 79</a>	How to create a platform model.
<a href="#">"Modifying a Platform Model " on page 80</a>	How to modify a platform model's properties and design. Includes information about adding or modifying infrastructure templates, connections, lifecycles and service blueprints.
<a href="#">"Publishing Platform Model Service Designs " on page 101</a>	How to publish a released platform model to a selected service provider.
<a href="#">"Deleting a Platform Model " on page 103</a>	How to delete a platform model.

**Note:** To provision a platform, see ["Provisioning a Platform " on page 121.](#)

## Opening a List of Platform Models

To open a list of platform models, complete the following steps:

1. Click the **Platforms** tab to open the Platforms Home window.
2. On the Platforms menu, choose one of the following:
  - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
  - **All Platforms** to see a list of all platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms that you have selected as Favorites.

3. *Optional:* Click the arrows beside the column names to sort the list in ascending or descending order.
4. *Optional:* Click a platform name to open its details window. See ["Opening a Platform Model Details Window " below.](#)

## Opening a Platform Model Details Window

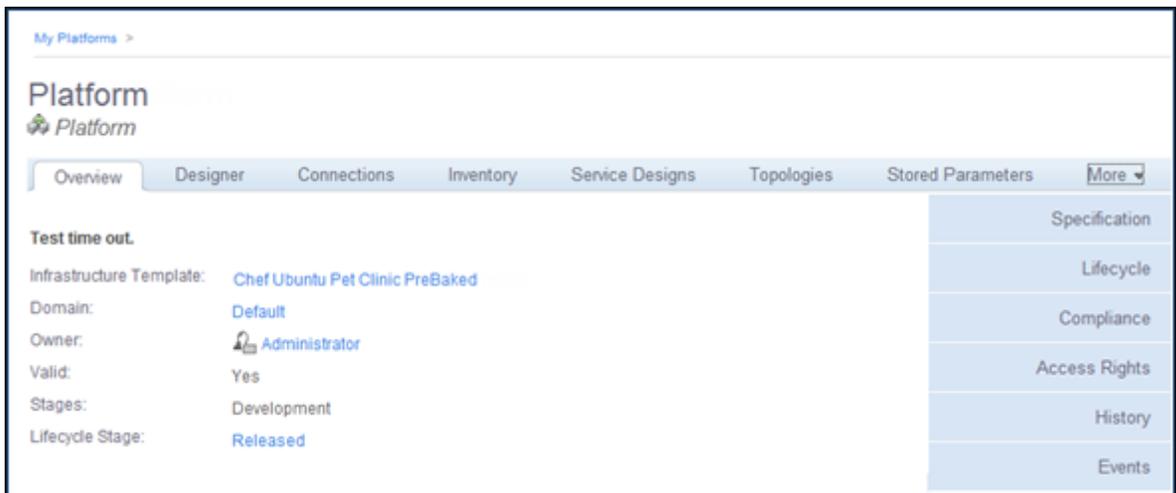
To open a platform model details window, complete the following steps:

1. Click the **Platforms** tab to open the Platforms Home window.
2. On the Platforms menu, choose one of the following:
  - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
  - **All Platforms** to see a list of all platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms that you have selected as Favorites.

3. *Optional:* Click the arrows beside the column names to sort the list in ascending or descending order.
4. Click a platform name to open its details window.

Platform details are organized on tabs.



The following tabs are display-only:

- **Overview:** Contains general information, such as the infrastructure templates on which the model is based, domain and user ownership information, the currently specified platform capabilities, and important events related to the platform model.
- **Inventory:** Lists the platforms that have been provisioned using this platform model and general information about each:

**Name:** Provisioned platform name.

**Stage:** The software development cycle stage to which a provisioned platform belongs.

**Deployed Applications:** The number of applications that are currently deployed on a provisioned platform.

**Status:** The platform provisioning operations status.

**Note:** Click **Name** to see a Platform Deployment screen that includes further details of a provisioned (or failed) platform.

- **Topologies:** Topologies that are associated with a platform.
- **History:** Lists the revision history of the platform model in a table format. You can use the information to compare revisions.
- **Compliance:** Artifact compliance details.
- **Events:** Lists events related to the platform model, and provides search and filtering capabilities on those events.

For information about tabs that you can use to modify a platform model, see "[Modifying a Platform Model](#)" on the facing page.

## Creating a Platform Model

To create a platform model, complete the following steps:

1. Click the **Platforms** tab to open the Platforms Home window.
2. On the Platforms menu, choose **Create Platform** to open the Create Platform wizard.
3. Make entry in the Basic properties wizard step:
  - a. **Name:** Type a screen name for the platform model.
  - b. **Description:** Type (optional) description.
  - c. **Stages:** Select software development cycle stages associated with the model. You can only provision the platform model to one of the stages you select here.
  - d. Click **Next**.
4. The Select template wizard step opens.
5. Click **Add Template** to open the Select template dialog box.

You see all the synchronized templates for the default deployer. Use this dialog box to add or delete infrastructure templates until your platform is complete.

6. To create platform infrastructure:

a. *Optional:* On the menu, choose a different deployer to see synchronized infrastructure templates for that deployer.

b. Select an infrastructure template.

HP CDA displays details of the template and a graphic preview, so you can determine whether the infrastructure template fulfills your needs.

The template name populates Template Instance Name.

c. If you want to add the template to the platform:

*Optional:* Edit **Template Instance Name**.

Click **Add template**.

d. If you don't want to add the template to your platform:

Click to select, review and add another template, or click **Cancel**.

**Note:** You can add infrastructure templates that use different deployers. HP CDA does not limit the number of infrastructure templates you can add.

7. Click **Next** when you are done adding templates.

8. A Summary window opens.

9. Click **Finish**.

## Modifying a Platform Model

You can make additions and modifications to platform models.

The following topics provide information about tasks platform developers perform when modifying platforms.

Topic	Information Included
<a href="#">"Modifying Infrastructure Design " below</a>	How to use a platform Designer tab to modify a platform's design. Includes information on how to add, remove or edit infrastructure templates, add or remove platform software, change hardware and operating system capabilities, and add information about a platform provisioning deployer.
<a href="#">"Modifying Platform Model Connections " on page 93</a>	How to use connections to describe an association to software and external service endpoints required by platform software.
<a href="#">"Unpublishing, Enabling and Disabling Platform Service Designs " on page 94</a>	How to create and publish a service design of a platform.
<a href="#">"Using the Platform Model Consumability Lifecycle " on page 96</a>	How to use the lifecycle that determines when a platform can be published. Includes information about how to change a platform's lifecycle stage, and end governance.
<a href="#">"Modifying Platform Model Stored Parameters " on page 98</a>	How to create, edit and delete stored parameters that are used during platform provisioning.
<a href="#">"Modifying Platform Model Specifications " on page 100</a>	How to add and edit specifications for a platform.
<a href="#">"Modifying Platform Model Access Rights " on page 100</a>	How to modify a platform's read and write access rights.

## ***Modifying Infrastructure Design***

After you create a platform model, you can use the platform Designer tab to add or modify certain design properties. You can modify:

- Infrastructure templates
- platform software
- Hardware and operating system capabilities
- Provisioning deployer parameters

Other properties are set in the platform's infrastructure templates and cannot be changed. Properties that you cannot change include:

- The basic platform components shown in the platform designer
- Component layout
- Connections

The following topics provide information about tasks platform developers perform when working with platform design.

Topic	Information Contained
<a href="#">"Opening a Platform Designer Tab " below</a>	How to open and use a graphic representation of a platform.
<a href="#">"Editing, Adding and Deleting Infrastructure Templates " on page 84</a>	How to access a native interface where you can edit templates. How to add and delete infrastructure templates from a platform.
<a href="#">"Modifying Platform Software Capabilities " on page 85</a>	How to add or remove platform software that will be part of the provisioned platform. That software will be available to application developers.
<a href="#">"Modifying Platform Hardware and Operating System Capabilities " on page 88</a>	How to assign capabilities to server groups that will be provisioned and used by application developers.
<a href="#">"Modifying Platform Deployer Capabilities " on page 90</a>	How to add information about the deployer that is used during platform provisioning.

## ***Opening a Platform Designer Tab***

To open a platform Designer tab, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

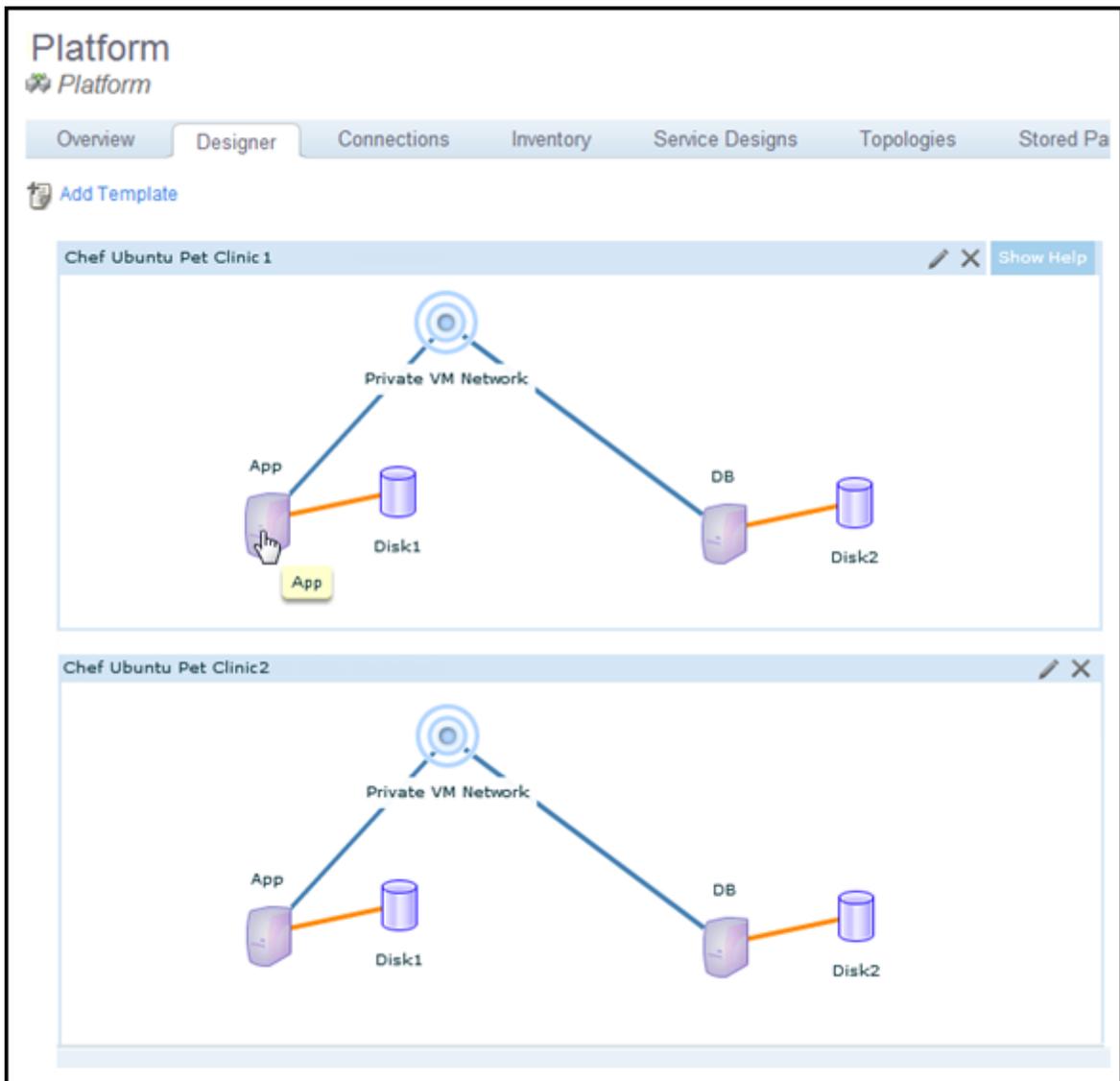
**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as

Favorites.

c. Click a platform name to open its details window.

2. Click the **Designer** tab.

The image shows a platform that includes two infrastructure templates. Click on a network, server group or disk to view (and in some cases, edit) its properties. See "[Modifying Infrastructure Design](#)" on page 81.



## ***Editing, Adding and Deleting Infrastructure Templates***

HP CDA provides access to templates' native interface, where you can edit the templates. You can access the interface from a platform Designer tab or the Infrastructure Home tab.

If you delete a template or edit it to remove capabilities for a platform that is included in a deployment topology, you might invalidate the deployment topology. After you make infrastructure changes, HP CDA performs a validation framework check and updates topology status (valid or invalid). To learn more, see "[Validating a Deployment Topology](#) " on page 56.

To edit an infrastructure template from a Designer tab, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.
  - c. Click a platform name to open its details window.
2. Click the **Designer** tab to open a platform Designer window. You see information and a graphic representation of the platform's infrastructure templates.
3. Click **Edit Template Source**  for a template to open the template in its native designer. Make edits in the designer.
4. When done, click **Sign Out** or close the designer window.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

To add a template to a platform, complete the following steps:

1. Open a platform Designer tab.
2. Click **Add Template** to open the Select template dialog box where you see all the synchronized templates for the default deployer. Make entry:
  - a. *Optional:* On the Default Deployer menu, choose a different deployer to see synchronized infrastructure templates for that deployer.

- b. Select a template. A preview opens, so you can decide whether the template meets your needs.
- c. *Optional:* Edit the template name in Template Instance Name.
- d. Click **Add template**. The template appears on the Designer tab.

To remove a template from a platform, complete the following steps:

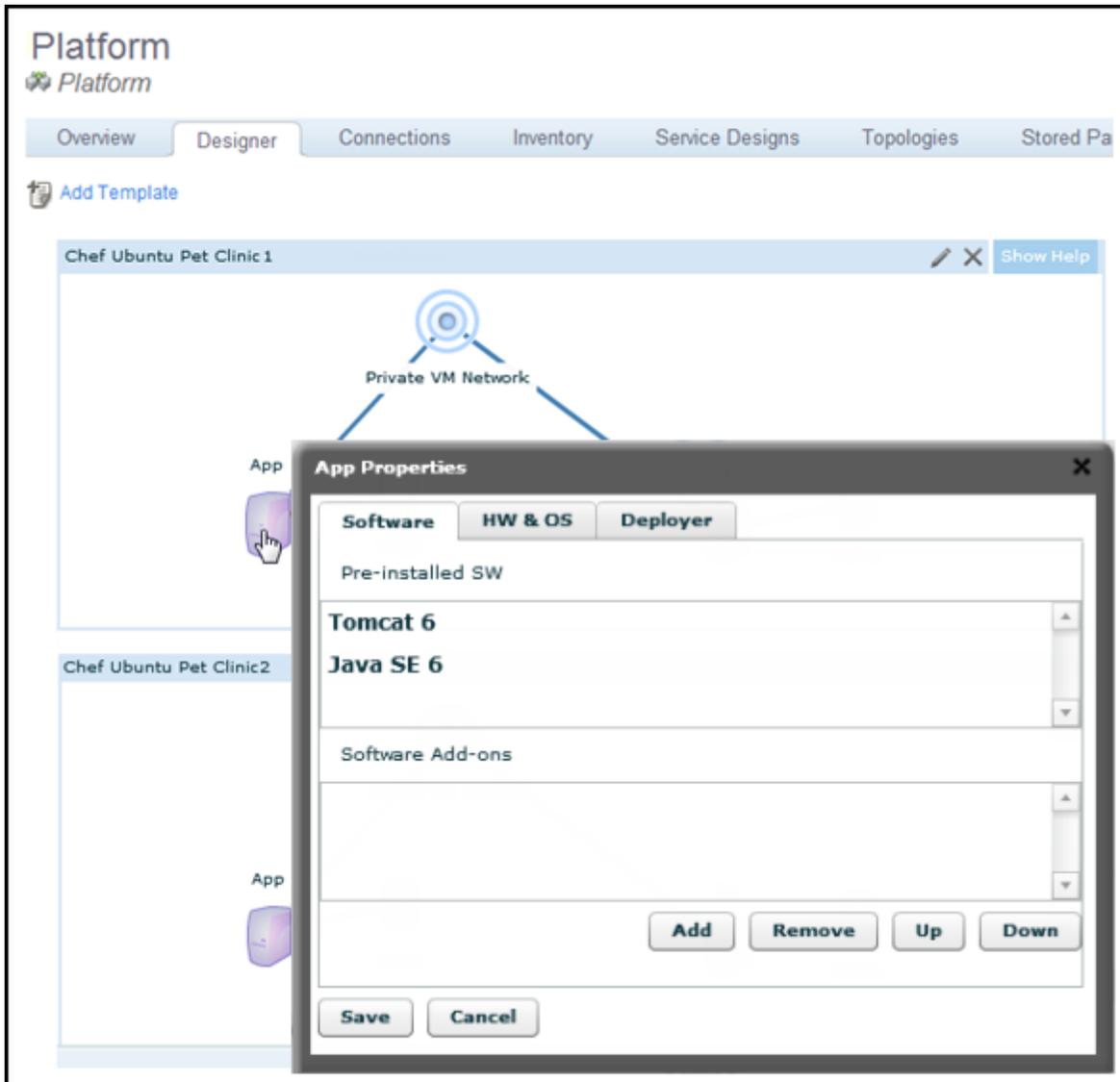
1. Open the platform Designer tab.
2. Click **Remove**  for the template.
3. Click **Yes** to confirm. The template no longer appears on the Designer tab.

## ***Modifying Platform Software Capabilities***

Software capabilities identify the software that will be installed on a provisioned platform. Software capabilities are assigned on a per server group basis.

Application developers can view assigned software capabilities to determine whether a platform's server groups provide the software capabilities their application layers require.

If you change software capabilities on a platform that is included in a deployment topology, you might invalidate the topology. After you make platform software changes, HP CDA performs a validation framework check and updates topology status (valid or invalid). To learn more, see ["Validating a Deployment Topology " on page 56](#).



To assign software capabilities, complete the following steps:

1. On the platform Designer window, click the symbol for the server group to which you want to assign software capabilities. A server group Properties dialog box opens.
2. Click the **Software** tab, if it is not selected.
3. The Software tab contains two areas that show the pre-installed and add-on software capabilities that are currently specified for the server group.

To add pre-installed software capabilities:

1. Click **Add** to open the Add New Software dialog box. Make entry:
2. Click **Pre-installed SW**.
3. Click **Next >**.
4. In the tree view that appears:
  - a. Select a software capability to add.
  - b. *Optional:* Select **Managed by Software Add-On**, if required.
  - c. Click **Next**.
5. Your Managed by Software Add-On selection determines which screen displays next.

If you selected **Managed by Software Add-On**, complete the following steps:

1. Select the add-on software that will manage the pre-installed software.
2. Click **Next >**, and then perform the steps for the Not Selected option.

If you did not select the option (or are continuing from above), complete the following steps:

1. If the software being added has parameters associated with it, modify the parameter values, if necessary, and click **Finish**.
2. If the software being added does not have parameters associated with it, click **Finish**.

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

3. Click **Save**.

To add add-on software capabilities, complete the following steps:

1. Click **Add** to open the Add New Software dialog box. Make entry:
  - a. Click **Add-on SW**, and then click **Next**.
  - b. Select add-on software from the list.
  - c. Click **Next >**.

- If the software being added has software capabilities associated with it, and the software represented by the capabilities has parameters associated with it, modify the parameter values, if necessary, and click **Finish**.
  - If the software being added does not have parameters associated with it, click **Finish**.
2. Click **Save** on the Properties dialog box.

To remove software capabilities, complete the following steps:

1. Click the software capability, and click **Remove**.
2. Click **Save** on the Properties dialog box.

To change the software capability listing order:

1. Click a software capability in either the **Pre-installed SW** or **Software Add-ons** list.
2. Click **Up** and **Down** to change the capability position in the list.
3. Repeat Step 2 as required to reorder the list.

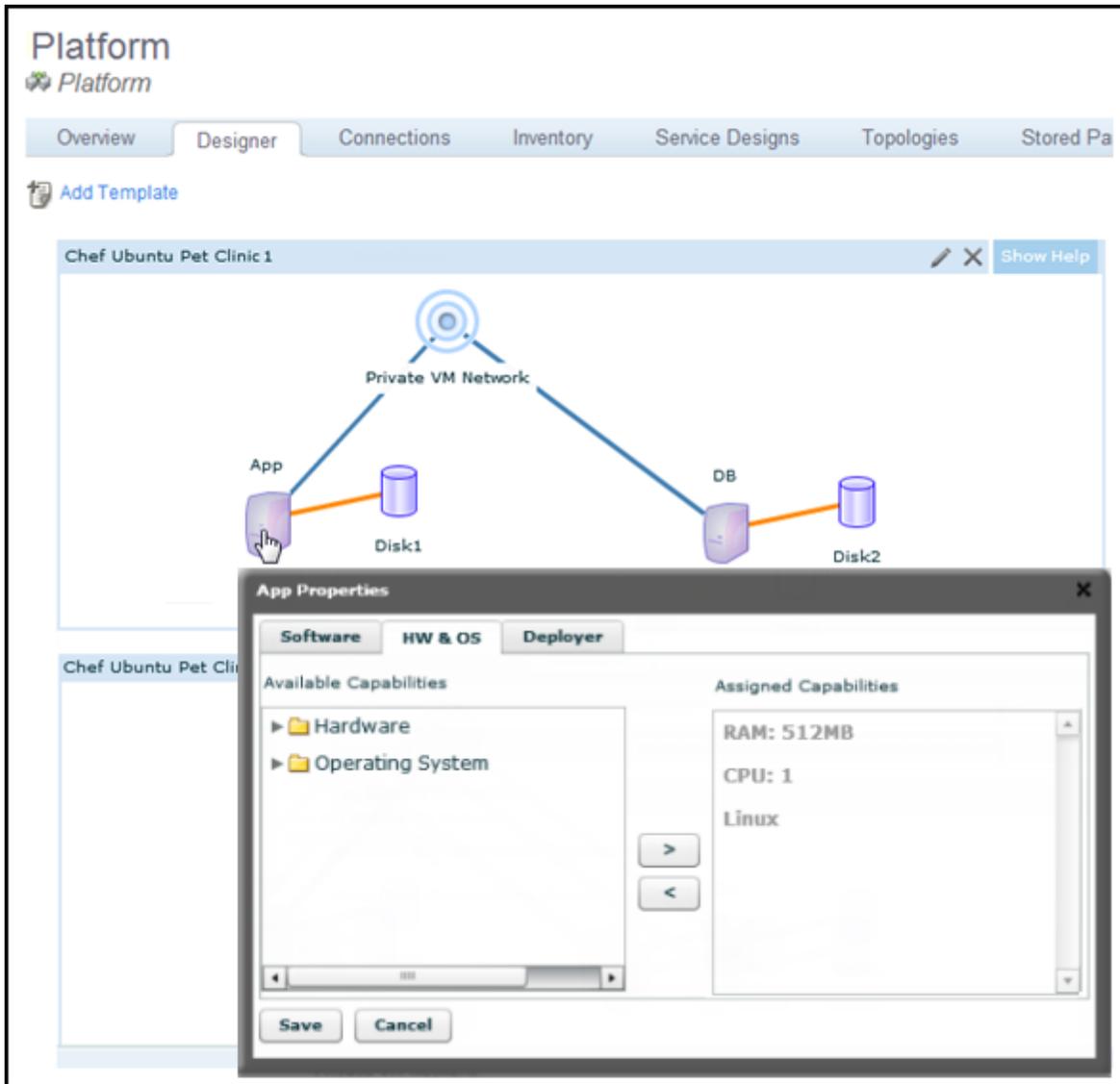
**Note:** When you change the Pre-installed SW list order, you only affect how the list appears in the interface. When you change Software Add-ons list order, you change both how the list appears, and also the order in which add-on software is installed when you provision the platform.

4. When done, click **Save** on the Properties dialog box to save the changes.

## ***Modifying Platform Hardware and Operating System Capabilities***

You can use a platform Designer tab to assign and modify hardware and operating system capabilities for the platform that will be provisioned from the platform model. These capabilities are assigned on a per server group basis.

If you change capabilities on a platform that is included in a deployment topology, you might invalidate the topology. After you make changes, HP CDA performs a validation framework check and updates topology status (valid or invalid). For more information, see "[Validating a Deployment Topology](#)" on page 56.



To assign capabilities, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

- c. Click a platform name to open its details window.
2. Click the **Designer** tab.
  3. Click the symbol for a server group to open its Properties dialog box. Make entry:
    - a. *Optional:* Click the **HW & OS** tab if it is not selected.
    - b. The HW & OS tab contains two areas:
      - o **Available Capabilities** (on the left) show the available hardware and OS capability tags.
      - o **Assigned Capabilities** (on the right) 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.
    - c. Expand **Available Capabilities** and select a capability that you want to assign.

If you do not see a capability that you want to set, you can add the capability on the Administration tab Tag Management menu option. If you do not have the required privileges, ask an Administrator.

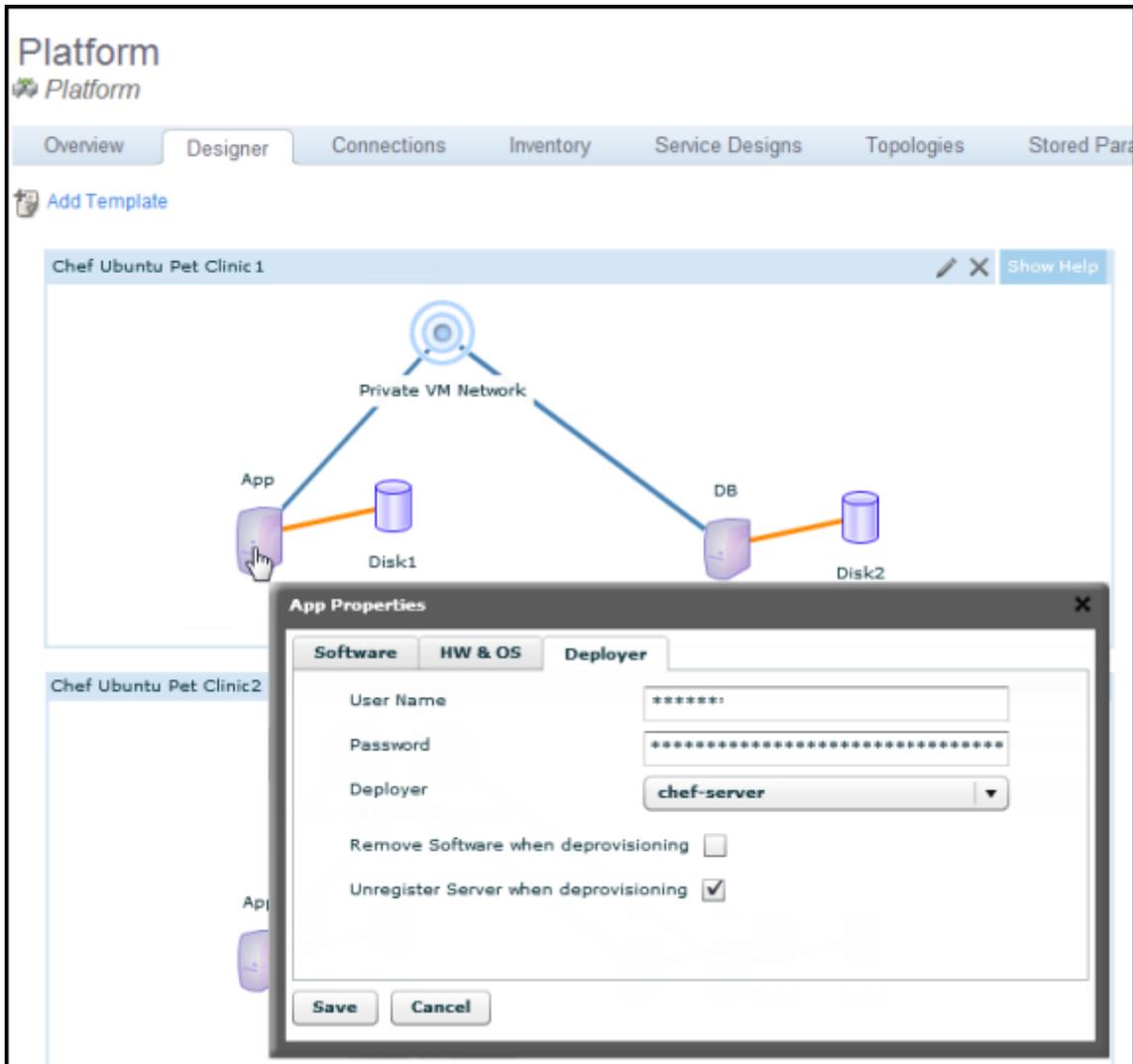
- d. Click the greater than (>) key to move the capability to the **Assigned Capabilities** list.

**Note:** Click the less than (<) key to remove capabilities from the Assigned Capabilities list.

- e. *Optional:* Continue moving hardware and OS capabilities as required.
- f. Click **Save** to save the changes.

## ***Modifying Platform Deployer Capabilities***

HP CDA requires information about the deployer used during platform provisioning.



To enter information about a platform provisioning deployer, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as

Favorites.

- c. Click a platform name to open its details window.
2. Click the **Designer** tab.
  3. Click the symbol for a server group to open a Properties dialog box. Make entry:
    - a. *Optional*: Click the **Deployer** tab, if it is not selected.
    - b. Modify parameters as required:

**Note:** If you choose HP SA on the Deployer menu, User Name and Password are inactive.

If you use key-based authentication, Password is inactive.

- **User Name:** Type the username of the person who will log into the deployer.

If you use cloud-based resources, your cloud service provider supplies a user name when your service account is set up.

**Note:** If you use Microsoft Windows domain-based resources, the user name must be in the format **<Domain>\<user>**.

- **Password:** Type the user password

**Note:** If you use cloud-based resources, the Password is inactive. A key file (rather than a password) provides authentication.

- **Deployer:** You see deployer plugins that are integrated with HP CDA on the menu. Choose a plugin configuration to use as a platform provisioning deployer.

**Note:** If you use a Chef deployer, you must set up a separate Chef server 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, Chef servers must be set up in the cloud.

- **Remove Software when deprovisioning:** Select this option to remove software from a server when the server is deprovisioned.

This option has no effect on virtual servers, because they are deleted as part of deprovisioning. However, if you use physical servers, this option can be useful.

- **Unregister Server when deprovisioning:** Select this option to unregister a server when it is deprovisioned.

If you do not choose this option for virtual servers, a server registration record is stored in the deployer database. The record is useless because virtual servers are deleted as part of deprovisioning.

However, leaving a registration record of a physical server in the deployer database can be useful.

- c. Click **Save**.

## ***Modifying Platform Model Connections***

A platform model can contain descriptions of communication endpoints that are exposed by platform software. 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 an endpoint database port for mySQL through TCP on port 3306.

Connections describe an association to software and external service endpoints required by platform software. For example, Web server platform software might need to communicate with a database server 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. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

- c. Click a platform name to open its details window.
2. Click the **Connections** tab to see platform connections.
  3. Click **New Endpoint** to open the New Endpoint dialog box. Make entry:

- a. **Software Instance:** Choose the platform software to which the endpoint is added.
- b. **Name:** Type an endpoint name to identify the endpoint (on the user interface) and to which a connection can be specified.
- c. **Description:** Optionally, type an endpoint description.
- d. **Path:** Optionally, type information to specify what is exposed as part of this end point. For example, on an application tier, this could be a Web-context exposed at the end point.
- e. **Port:** Type the port to be exposed.
- f. **Protocol:** Select the protocol the endpoint uses for communications.
- g. **Load Balanceable:** Select this option if 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 you do not select this option, requests will go to the server that hosts this tier.

**Note:** If you select the Load Balanceable option, you must enter load balancer parameter values in the deployment topology map.

- h. Click **OK**.

## ***Unpublishing, Enabling and Disabling Platform Service Designs***

After you publish a platform, you can view it on the platform Service Designs tab.

To view published service designs for a platform, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as

Favorites.

- c. Click a platform name to open its details window.
2. Click the **Service Designs** tab to open a list of service designs.
3. Click a service design name to open its details window.

You can change the status of a service design, so it is no longer available as a published design.

To unpublish a service design, complete the following steps:

1. On the Service Designs tab, select the service design.
2. Click **Unpublish**.
3. Click **OK** to confirm.

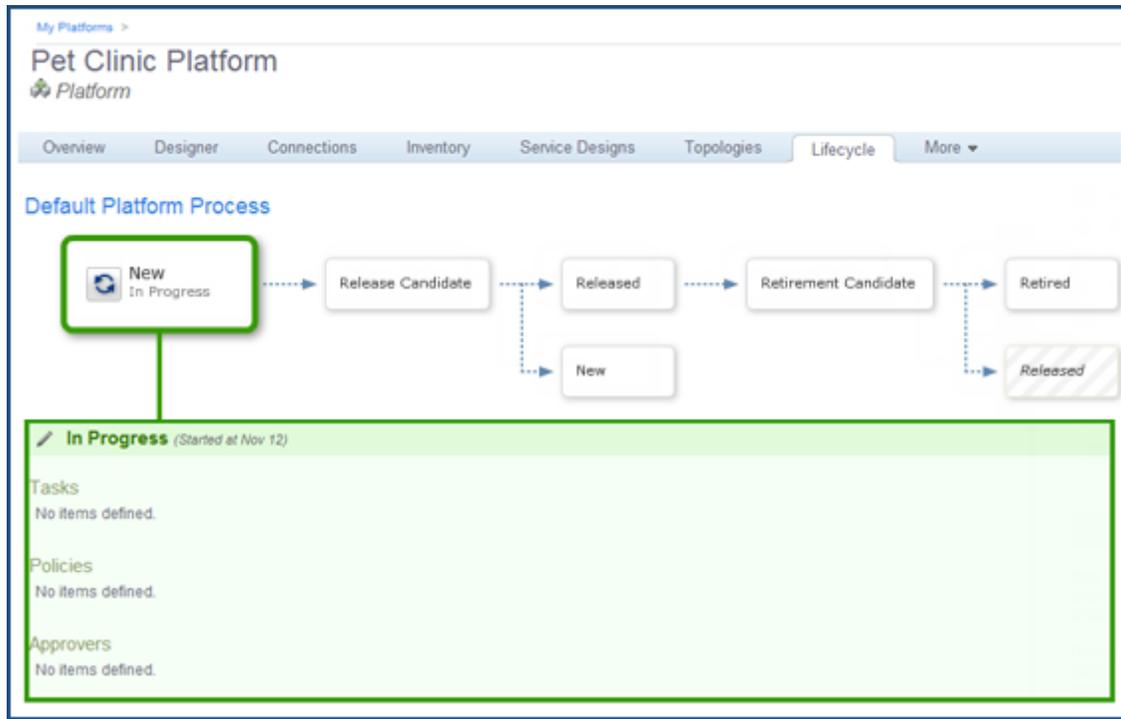
You can select settings that give your service design a status of Enabled or Disabled when published. (See your service provider documentation for information on what the statuses mean for your provider.)

To enable or disable a service design, complete the following steps.

1. On the Service Designs tab, select the service design.
2. Click **Enable** or **Disable**.
3. Click **OK** to confirm.

To publish a service design, see "[Publishing Platform Model Service Designs](#) " on page 101.

## Using the Platform Model Consumability Lifecycle



The platform Lifecycle tab shows the stages that govern whether a platform can be shared and published. When a platform developer creates a platform model, the platform is in the New stage. It cannot be published.

Above, you see the default lifecycle. As you work with a lifecycle, the graphic changes, so it becomes a visual timeline of your platform's lifecycle.

You can click on any stage to see information about the stage, but this action does not advance the platform. The platform's current stage displays larger than the other stages.

Lifecycle stages:

- **New:** The platform is in progress. It does not appear in All Platforms views and it cannot be published. (The publish icon, on the Overview and Service Designs tabs, is not active.) When a platform developer decides that the platform is ready to share, the developer moves it to Release Voting.
- **Release Candidate:** When the platform moves to this stage, HP CDA checks any policies associated with it. For example, a policy may require approvals at this stage, so approvers receive email notification of their approval tasks. If a policy fails, HP CDA returns the platform to the New stage. Otherwise, the platform advances to the Released stage.

**Note:** HP CDA requires all related software artifacts to be in the Released state before the platform can advance to Released. However, after the platform moves into Released state, a user can move related software artifacts back to the New Stage.

- **Released:** The platform appears on All Platforms and can be published. The publish icon is active.
- **Retirement Candidate:** When a platform is no longer needed, a platform developer can move it to the Retirement Voting stage. Any policies associated with it will be checked and executed, just as during the Release Voting stage. If policies are met, the platform moves to Retired. Otherwise, it returns to Released.
- **Retired:** The platform cannot be published or provisioned.

Administrators can add the following to stages:

- **Tasks:** Any assigned tasks must be completed for the platform to advance to the next stage.
- **Policies:** Any policies must be validated for the platform to advance to the next stage.
- **Approvers:** Approvers receive email notification when their assigned approval tasks come due.

### Lifecycle Actions

Platform designers can perform the following actions on the Lifecycle tab.

You can advance the platform to the next stage.

To advance to the next stage, complete the following actions:

1. Click **Move to Stage**.  
The next stage name displays.
2. Click the name to confirm.
3. The platform advances to the next stage.

If you have multiple lifecycles defined, you can move the platform to any stage in any lifecycle.

To move the platform to a selected stage and lifecycle, complete the following actions:

1. Click **Set Stage/Process** to open the Set Process dialog box. Make entry:
  - a. Select a lifecycle from the **Process** menu.
  - b. Select a stage from the **Stage** menu.

- c. Select a status for **Approved**.
  - d. Click **Set**.
2. The Lifecycle graphic shows the platform at the selected stage and process.

You can end governance and remove a platform from its lifecycle.

To end governance, complete the following steps:

1. Click **End Governance**.
2. Click **End Governance** in the confirmation box.

The Lifecycle graphic no longer appears. You see options to start governance.

To start governance, complete the following steps:

- Click **Start Governance**.

The window opens at the New stage of the default lifecycle process.

## ***Modifying Platform Model Stored Parameters***

You can store one or more parameter sets (infrastructure design settings) for each platform infrastructure template. During platform provisioning, you can select and associate one set of stored parameters with each infrastructure template being provisioned.

You can create stored parameters during provisioning or add parameters to an existing platform infrastructure template.

To add stored parameters to a platform infrastructure template, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as

Favorites.

- c. Click a platform name to open its details window.
2. Click the **Stored Parameters** tab. (It may be on the More tab menu.)
3. Click **New** to open the New Configuration wizard. Make entry:
  - a. Enter the stored configuration **Name**.
  - b. Click **Next**.
  - c. Enter values for the infrastructure design settings.
  - d. Click **Finish**.

The stored parameter appears in the Stored Parameters list.

To edit a parameter set that is stored in a platform model infrastructure template, perform the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

- c. Click a platform name to open its details window.
2. Click the **Stored Parameters** tab to see a list of stored parameters.
3. Select a parameter set.
4. Click **Edit** to open the Edit Configuration wizard.
  - a. Accept or edit **Name**.
  - b. Click **Next**.

- c. Edit the infrastructure design settings.
- d. Click **Finish**.

The wizard closes and the parameter set saves.

To delete a parameter set that is stored in a platform model's infrastructure template, perform the following steps:

1. On the Stored Parameters tab, select a stored parameters set.
2. Click **Delete**.

## ***Modifying Platform Model Specifications***

Use the Specification tab to open a text editor where you can enter or edit specifications for the platform model. Click **Source** in the text editor to display the HTML formatting tags in the specification.

## ***Modifying Platform Model Access Rights***

You can assign platform read and write access rights to users, groups and roles.

To add access rights, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

- c. Click a platform name to open its details window.

2. Click the **Access Rights** tab.
3. Click either **Add Read Only Access** or **Add Read/Write Access**.
4. An Add dialog box opens.
5. Click **Select Entities**.

The **Select Entities** dialog box opens.

6. Select entities by doing one of the following:
  - Make entry to search for a user, group or role.
  - Browse and click to select entities.
7. Click **Select**.

The entity displays in the **Add** dialog box.

8. Click **Add**.

The entity displays in the **Access Rights** list.

To remove access rights, complete the following steps:

1. On the Access Rights tab, select a user, group or role.
2. Click **Remove**.
3. Click **Remove** in the **Remove Access** confirmation box.
4. The user, group or role no long appears with those rights.

## Publishing Platform Model Service Designs

You can **publish** a platform model as a service design, which means the platform's service design is passed to a provider, and the provider can further modify the design to create a service offering. To publish, the following requirements must be met:

- The platform and its artifacts must be in Released state.
- You must have a plugin configured for the provider.

When a platform is eligible for publication, the Publish icon becomes active.

Platform service designs can be published from either the platform's **Overview** or **Service Designs** tabs.

To publish a platform service design, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.
  - c. Click a platform name to open its details window.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

2. Click the **Overview** or **Service Designs** tab.
3. Click **Publish**.

**Note:** This operation publishes the current platform model. If you click to select a current service design, it will have no effect on the operation.

The Publish a Platform Service Design wizard opens.

4. Read the Introduction window information about the required steps.
5. Click **Next**.
6. On the Provider window:
  - a. On the menu, choose a **Provider**. The service design will be published to that provider.
  - b. Click **Next**.
7. On the Service Design Properties window:
  - a. Type **Service Design Name** for the platform service design.
  - b. *Optional:* Type a **Service Design Description**.
  - c. Click **Next**.

8. On the Lifecycle window:
  - a. Select a **Default** lifecycle stage. This selection determines what lifecycle stage is selected by default when users create a service request.
  - b. Click the eye  to turn **Visibility** of the lifecycle stages on or off. A stage must be marked **Visible** to be visible in the service offering.
  - c. Click **Next**.
9. An Infrastructure Design window opens for each infrastructure template in the platform. Different templates have different entry requirements. Existing infrastructure templates do not require any entry.
  - a. Change editability and visibility as for other parameters.
  - b. Click **Next**.
10. A Summary window of your entry opens.
11. *Optional:* Click **Back** to return to windows and make changes.
12. Click **Finish**.

Publishing begins.

## Deleting a Platform Model

To delete a platform model, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

- c. Click a platform name to open its details window.

2. Select a platform, and click **Delete**.
3. Click **OK** to confirm.

The platform is deleted and no longer appears in the list.

## Using Add-on Platform Software

At times, infrastructure templates might not specify all of the software capabilities that your platform requires. In that case, you can model and include additional software, known as add-on software.

A platform software model consists of the following:

- **Requirements and Capabilities:**

**Requirements:** Identify what the platform software requires of the platform onto which it is deployed for successful software execution. Requirements might include an operating system, minimum amount of memory, other software, etc.

**Capabilities:** Identify the additional capabilities that the platform software brings to the platform on which it is installed.

**Note:** Use HP CDA's tagging feature to add requirements and capabilities. Designers later reference the tags during platform and application modeling.

- **Workflows** that contain:

**Programming Operations:** Can define input component parameters that reference parameter definition groups. Parameter definitions groups provide platform software-level reference points for values that are used within more than one programming operation. Definitions eliminate the need to type the same value in various programming operations.

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

- 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

The following topics provide information about tasks that platform developers perform when working with platform software.

Topic	Information Included
<a href="#">"Browsing Lists of Add-on Platform Software " below</a>	How to open lists of add-on software. Includes information about how to quickly edit or delete software.
<a href="#">"Using HP CDA Out-of-Box Monitoring Agents " on the facing page</a>	How to use HP CDA out-of-box monitoring agents (probes) for platform software.
<a href="#">"Opening an Add-on Platform Software Details Window " on page 108</a>	How to open an add-on platform software details window to see details such as workflows, parameters, lifecycle and access rights.
<a href="#">"Creating Add-on Platform Software " on page 109</a>	How to create add-on platform software.
<a href="#">"Modifying Add-on Platform Software " on page 110</a>	How to modify platform software properties such as requirements and capabilities, software workflows, parameters and specifications.
<a href="#">"Deleting Add-on Platform Software " on page 115</a>	How to delete add-on platform software.

## Browsing Lists of Add-on Platform Software

To see a list of add-on platform software, complete the following steps:

1. Click the **Platforms** tab to open the Platforms Home window.
2. On the Software menu, choose **Browse Software** to open the Software window.
3. *Optional:* Change the view between All software, My software (created by the user who is logged in), software that is designated as Favorites (by the user who is logged in) or Released software.
4. Use the Software window to perform the following tasks.

To edit add-on platform software, complete the following steps:

1. Select add-on platform software, and click **Edit** to see a menu of edit options.
2. Choose edit options, and work through the prompts.

To delete add-on platform software, complete the following steps:

1. Select add-on platform software, and click **Delete**.
2. Click **OK** to confirm the delete.

The add-on platform software is deleted and no longer appears in the list.

## Using HP CDA Out-of-Box Monitoring Agents

HP CDA includes monitoring agents (probes) for software that is provisioned with the platform. The monitoring agents are listed as platform software. See ["Browsing Lists of Add-on Platform Software" on previous page](#).

The following table lists HP CDA out-of-box agents (probes) for platform software.

Platform Software	Description
HP Diagnostics	
Diagnostics .NET Probe for Windows	<p>The software completes the following steps:</p> <ul style="list-style-type: none"> <li>• Installs the Diagnostics .NET probe on a Windows target server.</li> <li>• The probe software is downloaded from an HTTP location specified in <b>Platforms &gt; Browse Software &gt; Software Name &gt; Workflows &gt; Download .NET Probe</b>.</li> </ul> <p>You must:</p> <ul style="list-style-type: none"> <li>• modify the location to the HTTP server that contains the probe software reachable from the CDA server. Use <b>Platforms &gt; Browse Software &gt; Software Name &gt; Workflows &gt; Download.NET Probe</b>.</li> </ul>

Diagnostics Java Probe for Linux	<p>The software completes the following steps:</p> <ul style="list-style-type: none"> <li>• Installs the Diagnostics Java probe on a Linux target server.</li> <li>• The probe software is downloaded from an HTTP location specified in <b>Platforms &gt; Browse Software &gt; Software Name &gt; Workflows &gt; Download Java Probe</b>.</li> </ul> <p>You must:</p> <ul style="list-style-type: none"> <li>• modify the location to the HTTP server that contains the probe software reachable from the HP CDA server. Use <b>Platforms &gt; Browse Software &gt; Software Name &gt; Workflows &gt; Download Java Probe</b>.</li> </ul>
Diagnostics Java Probe for Windows	<p>The software completes the following steps:</p> <ul style="list-style-type: none"> <li>• Installs the Diagnostics Java probe on a Windows target server.</li> <li>• The probe software is downloaded from an HTTP location specified in <b>Platforms &gt; Browse Software &gt; Software Name &gt; Workflows &gt; Download Java Probe</b>.</li> </ul> <p>You must:</p> <ul style="list-style-type: none"> <li>• modify the location to the HTTP server that contains the probe software reachable from the HP CDA server. Use <b>Platforms &gt; Browse Software &gt; Software Name &gt; Workflows &gt; Download Java Probe</b>.</li> </ul>
Nagios	
Nagios_Linux_Agent	<p>The software completes the following steps:</p> <ul style="list-style-type: none"> <li>• Installs the Nagios agent on a Linux (Ubuntu10.4 and Ubuntu12.4) target server.</li> </ul>
HP Operations Manager	
HP Operations Agent for Unix	<p>The software completes the following steps:</p> <ul style="list-style-type: none"> <li>• Downloads the agent TAR ball from an external HTTP server or DSL onto the target server.</li> <li>• Installs the HP Operations Agent and configures it to HP Operations Manager.</li> </ul>

HP Operations Agent for Windows	The software completes the following steps: <ul style="list-style-type: none"><li>• Downloads the agent TAR ball from an external HTTP server or DSL onto the target server.</li><li>• Installs the HP Operations Agent and configures it to HP Operations Manager.</li></ul>
---------------------------------	---

## Opening an Add-on Platform Software Details Window

To open an add-on platform software window, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.
  - c. Click a platform name to open its details window.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

2. On the Software menu, choose **Browse Software** to open the Software window.
3. The Software window provides the following tabs:
  - **All:** Lists all platform software in HP CDA
  - **My:** Lists all platform software owned by the user who is currently logged in
  - **Favorites:** Lists all platform software that is marked as a favorite by the user who is currently logged in
  - **Released:** Lists all software that has been approved and released
4. Tabs provide the following details:
  - **Name:** Platform software name.
  - **Lifecycle Stage:** See "[Using the Platform Software Consumability Lifecycle](#)" on page 113.

- **Owner:** The platform software owner.
- **Domain:** The domain to which the platform software belongs.

## Creating Add-on Platform Software

**Note:** Do not create add-on software for the purpose of adding HP SA agent software. HP CDA cannot add HP SA agent software as add-on platform software. Two alternate methods of adding HP SA are to:

Include the HP SA agent software in an infrastructure template for virtual machines that might use HP SA as a deployer.

Add the SA agent software to the provisioned platform manually, and register the provisioned platform with the SA server.

To create add-on platform software, complete the following steps:

1. Click the **Platforms** tab to open the Platforms Home window.
2. On the Software menu, choose **Create Software** to open the Software window and make entry:
  - a. Type a **Name** for the platform software model.
  - b. *Optional:* Type a **Description**.
  - c. *Optional:* Add requirements. Click here to see steps.
    - i. Under the Requirements heading, click **Change** to open the Change Layer Requirements dialog box.
    - ii. Click a tab to change hardware, operating system or software requirements.
    - iii. Add or delete requirements until they are complete.
    - iv. Click **OK** to save the requirements.
  - d. Click **Save** to save the platform software.

For example, to create platform software that requires Linux 2.6 or newer, complete the following steps:

1. On the Software window, under the Requirements heading, click **Change** to open the Change Layer Requirements dialog box.
2. Click the **Operating System** tab.
3. Click **Add Requirement** to open a New Requirement dialog box.
4. On the Operator menu, choose **Exact and Newer Versions**.
5. Under the Requirement heading, expand Unix and select **Linux 2.6** (Unix > Linux > Linux2.6).
6. On the Architecture menu, choose **64 bit**.
7. Click **OK** to save the requirement.

## Modifying Add-on Platform Software

The following topics provide information about tasks platform developers perform when modifying platform software.

Topic	Information Contained
<a href="#">"Modifying Platform Software Requirements and Capabilities " below</a>	Information about changing software requirements and capabilities.
<a href="#">"Modifying Platform Software Access Rights " on next page</a>	How to change read and write access rights for add-on platform software.
<a href="#">"Using Platform Software Workflows " on page 112</a>	Information about using HP CDA workflows and programming operations.
<a href="#">"Using Platform Software Parameters " on page 112</a>	Information about software parameter groups and referencing.
<a href="#">"Modifying Platform Software Specifications " on page 112</a>	How to use a text editor to enter platform software specifications.
<a href="#">"Using the Platform Software Consumability Lifecycle " on page 113</a>	How to use the platform software lifecycle.

## Modifying Platform Software Requirements and Capabilities

At times, infrastructure templates might not specify all of the software capabilities that your platform requires. In that case, you can model and include additional software, known as add-on software.

A platform software model consists of the following:

- **Requirements and Capabilities:**

**Requirements:** Identify what the platform software requires of the platform onto which it is deployed for successful software execution. Requirements might include an operating system, minimum amount of memory, other software, etc.

**Capabilities:** Identify the additional capabilities that the platform software brings to the platform on which it is installed.

**Note:** Use HP CDA's tagging feature to add requirements and capabilities. Designers later reference the tags during platform and application modeling.

## ***Modifying Platform Software Access Rights***

To modify read and write access rights for add-on platform software, complete the following steps:

1. Click the **Platforms** tab to open the Platforms Home window.
2. On the Software menu, choose **Browse Software** to open the Software window.
3. Click a platform software name to open its details window.
4. Click the **Access Rights** tab to see read and write access rights.
5. Use the Access Rights tab to perform the following tasks.

To add rights, complete the following steps:

1. Click either **Add Read Only Access** or **Add Read/Write Access** to open a related dialog box.
  - Click **Select Entities** to open the Select Entities dialog box.
    - i. Select entities by using either the standard search functionality or by browsing.
    - ii. Click **Select** to add the entity to the Add dialog box.
2. Click **Add**.

The entity is added to the Access Rights list.

To remove access rights, complete the following steps:

1. Select the entity and click **Remove**.
2. Click **Remove** in the Remove Access confirmation box.

The entity no longer appears in the Access Rights list.

## ***Using Platform Software Workflows***

You can create workflows, such as Deploy and Undeploy, that contain a sequence of programming operations. You can also create interfaces that ensure the delegation of platform software workflows. To learn more, see "[HP CDA Workflows](#) " on page 372

## ***Using Platform Software Parameters***

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.

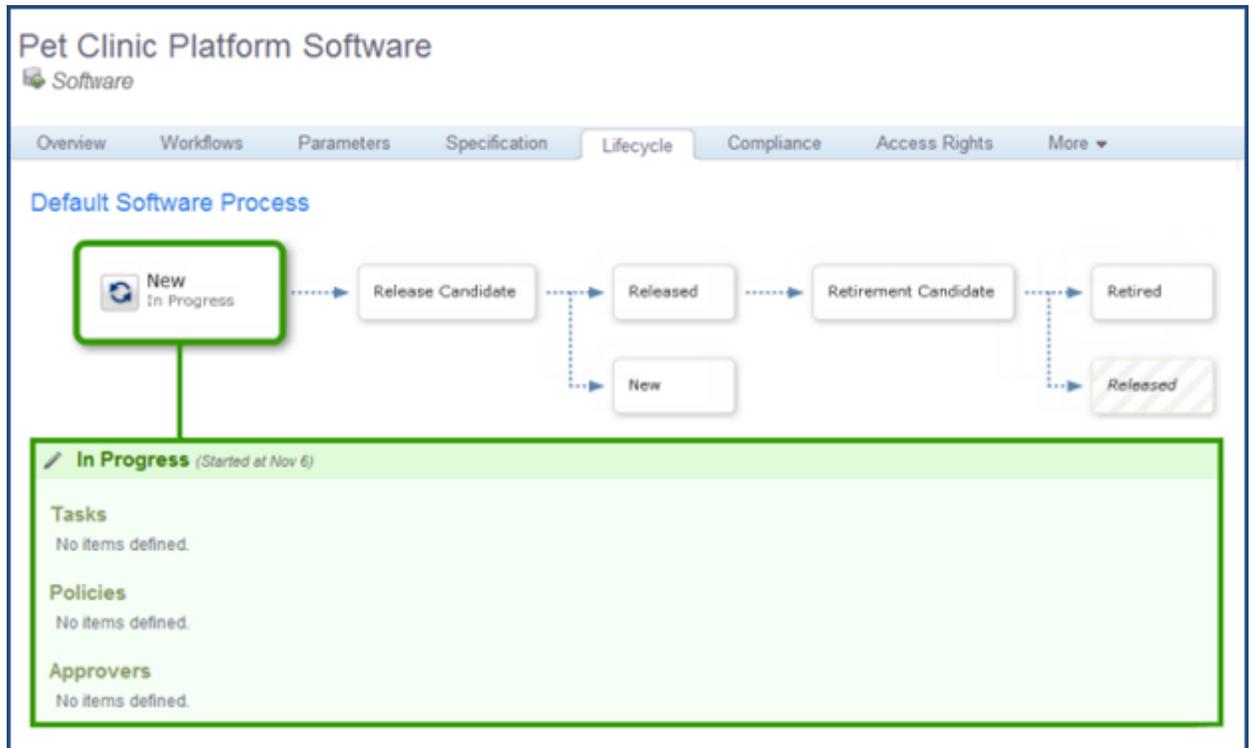
Parameters that are defined in the parameter definitions group in turn can reference the following other types of parameters:

- 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.

## ***Modifying Platform Software Specifications***

Use the Specification to open a text editor where you can enter or edit specifications for the platform model. Click **Source** in the text editor to display the HTML formatting tags in the specification.

## Using the Platform Software Consumability Lifecycle



Above, you see the default lifecycle. As you work with a lifecycle, the graphic changes, so it becomes a visual timeline of your software's lifecycle.

You can click any stage to see information about the stage, but this action does not advance the software. The software's current stage displays larger than the other stages.

Lifecycle stages:

- **New:** The software is in progress. It cannot be published. (The publish icon, on the Overview and Service Designs tabs, is not active.) When a platform developer decides that the software is ready to share, the developer moves it to Release Voting.
- **Release Candidate:** When the software moves to this stage, HP CDA checks any policies associated with it. For example, a policy may require approvals at this stage, so approvers receive email notification of their approval tasks. If a policy fails, HP CDA returns the software to the New stage. Otherwise, the software advances to the Released stage.

**Note:** HP CDA requires all related software artifacts to be in the Released state before the platform can advance to Released. However, after the platform moves into Released state, a user can move related software artifacts back to the New Stage.

- **Released:** The software is available for use.

- **Retirement Candidate:** When add-on platform software is no longer needed, a platform developer can move it to the Retirement Voting stage. Any policies associated with it will be checked and executed, just as during the Release Voting stage. If policies are met, the software moves to Retired. Otherwise, it returns to Released.
- **Retired:** The software is no longer available for use.

Administrators can add the following to stages:

- **Tasks:** Any assigned tasks must be completed for the software to advance to the next stage.
- **Policies:** Any policies must be validated for the software to advance to the next stage.
- **Approvers:** Approvers receive email notification when their assigned approval tasks come due.

### Lifecycle Actions

Platform designers can perform the following actions on the Lifecycle tab.

You can advance the software to the next stage.

To advance to the next stage, complete the following actions:

1. Click **Move to Stage**.  
The next stage name displays.
2. Click the name to confirm.
3. The software advances to the next stage.

If you have multiple lifecycles defined, you can move the software to any stage in any lifecycle.

To move the software to a selected stage and lifecycle, complete the following actions:

1. Click **Set Stage/Process**.
2. In the **Set Process** dialog box:
  - a. Select a lifecycle from the **Process** menu.
  - a. Select a stage from the **Stage** menu.
  - b. Select an **Approved** status.
  - c. Click **Set**.

The Lifecycle graphic displays the software at the selected stage and process.

You can end governance and remove the software from its lifecycle.

To end the governance of software, complete the following steps:

1. Click **End Governance**.
2. Click **End Governance** in the confirmation box.

The Lifecycle graphic no longer displays. You see options to start governance.

To start governance on the software, complete the following steps:

1. Click **Start Governance**.

The software displays at the New stage of the default lifecycle process.

## Deleting Add-on Platform Software

You can delete add-on platform software in two locations.

To delete when you have the add-on platform software displayed, complete the following steps:

1. Click the **Overview** tab.
2. Click the **Delete** link on the right.
3. Click **OK** to confirm the delete.

A message displays to confirm successful deletion.

To delete from the **Software** listing window, complete the following steps.

1. Click the **Platforms** tab, from the highest level tabs.
2. Click the **Browse Software** link on the left.

The Software window displays.

3. Click to select the software you want to delete.
4. Click **Delete**.
5. Click **OK** to confirm the delete.

The software no longer displays.

## Exporting and Importing Platform Software

Platform software can be exported from the HP CDA repository as an XML file, which can be modified if desired and imported to the repository. The XML that is exported is in an easy to read and understand format to facilitate editing in a standard XML browser or text editor.

The following topics explain how to export and import platform software.

Topic	Information Contained
<a href="#">"Exporting Platform Software " below</a>	How to export platform software from the HP CDA repository as an XML file.
<a href="#">"Importing Platform Software " on next page</a>	How to import a platform software XML file into the HP CDA repository.

### Exporting Platform Software

You can export platform software from the HP CDA repository as XML. This feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

To export platform software as XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Browse the HP CDA platform software by clicking the Browse Software link in the Platforms section of the HP CDA user interface, and copy the exact name of the platform software that you want to export.
3. Open the REST client, and issue a GET request with the following characteristics:

- **URL:**

```
https://<cda_
address>:8443-
/cda/remote/rest/v1/platform/software?name="<platform_sw_name>"
(Secure HTTP)
```

```
http://<cda_
address>:8080-
/cda/remote/rest/v1/platform/software?name="<platform_sw_name>"
(Standard HTTP)
```

Where `<cda_address>` is the IP address or FQDN of the HP CDA instance, and `<platform_sw_name>` is the name of the platform software that you want to export. If the GET request succeeds (a status code of 200 OK is returned), the XML for the platform software will be returned.

**Note:** To export all platform software XML, issue a GET request to the above URL without the `?name="<platform_sw_name>"` parameter.

- *Content Type:* application/xml
4. Copy the platform software XML to an editor for modification, or save it to a text file for later editing and import.

## Importing Platform Software

You can import platform software XML to the HP CDA repository. The XML that you import has been exported using the procedure ["Exporting Platform Software" on previous page](#) and typically has been stored to a text file. Using the import functionality, you can perform following operations:

- Create a new platform software by specifying a platform software name that does not exist already in the HP CDA repository. This will create a new platform software, the underlying workflow operations, workflow components and parameters defined as part of the platform software.

**Note:** While importing a platform software, it is essential that the XML tag hierarchy be maintained. Hence it is advisable to export an existing platform software and edit its name, the underlying workflows, workflow components, and parameters rather than creating the XML import document from scratch.

- Update an existing platform software by specifying the same Platform Software Name in the XML and specifying `force=true` in the URL as mentioned in the procedure below. In such a case, you can add additional workflows, workflow operations, and parameters to the existing platform software XML and those changes will be reflected when the XML document is imported.

The platform software import feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

To import platform software XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Open the REST client and issue a POST request with the following characteristics:

- **URL:**

`https://<cda_address>:8443/cda/remote/rest/v1/platform/software`  
(Secure HTTP)

`http://<cda_address>:8080/cda/remote/rest/v1/platform/software`  
(Standard HTTP)

Where <cda\_address> is the IP address or FQDN of the HP CDA instance.

- **Header:** application/xml

- **Body:** The platform software XML to be imported to the repository.

If the POST request succeeds (a status code of 200 OK is returned), the XML for the platform software has been imported into the HP CDA repository and you can now view it in the HP CDA platform software browser.

**Note:** If a status code of 409 Conflict is returned, platform software with the same name already exists in the HP CDA repository. To import and overwrite the platform software in the repository with the new version, issue the POST request with the "force" parameter set to "true," for example:

```
https://<cda_
address>:8443/cda/remote/rest/v1/platform/software?force=true
```

```
http://<cda_
address>:8080/cda/remote/rest/v1/platform/software?force=true
```

## Provisioning and Deprovisioning Platforms

A platform is an environment upon which you can deploy one-to-many applications, each with a different topology.

**Note:** The same topology can be deployed only once per realized platform.

Platforms include several elements:

- Platform Provisioning Plugins: to provide the architectural element that allows HP CDA to extend and integrate with infrastructure providers.
- Plugin Configurations: to define an instance of an infrastructure provider.
- One or more infrastructure templates that are suited to the needs of your intended applications. You can add multiple templates to a single platform to establish hybrid provisioning across

different providers, or you can use multiple templates from the same provider.

- If your infrastructure templates do not specify all the software capabilities your infrastructure needs, you can model additional (add-on) software and add it to your platform model.

When your platform model is complete, you can provision it to create a realized platform. Then you can deploy applications to the platform.

At the end of a platform's life, you deprovision it. This step is important, because it frees up resources. During the deprovisioning process, add-on software that was added during provisioning will be undeployed before deprovisioning of the platform. This option of undeploying platform software is specified by the platform designer under the deployer tab of each platform tier.

After you create a platform model, you must provision the platform before you can deploy an application to it. Provisioning turns the platform model into a realized platform.

The following topics provide information about tasks that platform developers perform when provisioning platforms.

Topic	Information Contained
<a href="#">"Opening a List of Provisioned Platforms " below</a>	How to open a list of provisioned platforms that you own, or a list of all provisioned platforms in HP CDA.
<a href="#">"Opening a Provisioned Platform Details Window " on the facing page</a>	How to find and open details about a provisioned platform.
<a href="#">"Provisioning a Platform " on page 121</a>	How to provision a platform. Includes information about how to cancel during a provision.
<a href="#">"Using the Platform Provisioning Report " on page 123</a>	How to use the Platform Provisioning Report during provisioning to monitor ongoing operations and cancel provisioning.
<a href="#">"Deprovisioning a Platform " on page 123</a>	How to deprovision a platform.

## Opening a List of Provisioned Platforms

To see a list of provisioned platforms, complete the following steps:

1. Click the **Platforms** tab to open the Platforms Home window.
2. On the Provisioned Platforms menu, choose one of the following:
  - **My Provisioned Platforms** to open a list of provisioned platforms that are owned by the user who is currently logged in.
  - **All Provisioned Platforms** to open a list of all provisioned platforms in HP CDA.

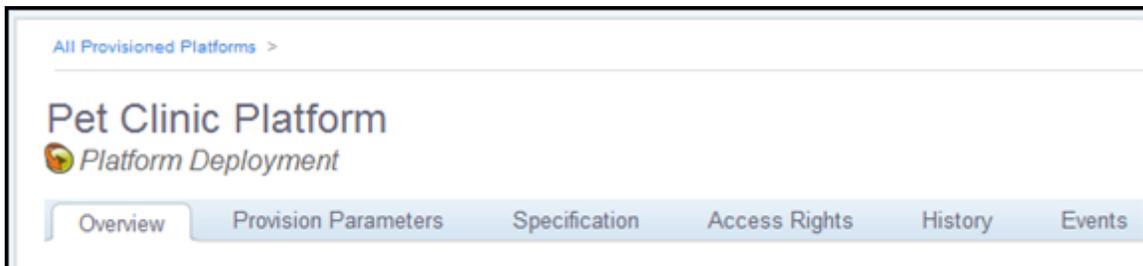
3. *Optional:* Click the arrows beside the column names to sort the list in ascending or descending order.
4. Click a provisioned platform name to open its details window. See "[Opening a Provisioned Platform Details Window](#) " below.

## Opening a Provisioned Platform Details Window

1. Display the provisioned platform you want. (Click here for steps.)
  - a. Click **Platforms** to open the Platforms Home window.
  - b. On the Provisioned Platforms menu, choose one of the following:
    - **My Provisioned Platforms** to open a list of platform models that are owned by the user who is currently logged in.
    - **All Provisioned Platforms** link to open a list of all released provisioned platforms in HP CDA.

**Note:** After you see a list, you can use the tabs that appear above it to switch between My Provisioned Platforms, All Provisioned Platforms, and platforms you have selected as Favorites.

- c. Click a provisioned platform name to open its details window.



Provisioned platforms provide the following tabs:

- **Overview:** Lists information about the platform and applications that are deployed on it.

Click under the State heading to open the related platform provisioning Report. See "[Using the Platform Provisioning Report](#) " on page 123.

You can deprovision on this tab. See "[Deprovisioning a Platform](#) " on page 123.

- **Provision Parameters:** See the parameters associated with each of the provisioned platform's templates.

- **Specification:** Lists specifications that are associated with the provisioned platform.
- **Access Rights:** Lists read and write access rights for the provisioned platform. You can modify rights on this tab.
- **History:** Lists revisions of the provisioned platform.
- **Events:** Lists major events (and the actors) for the provisioned platform.

## Provisioning a Platform

To find a platform to provision, complete the following steps:

1. Open the platform model details window. (Click here for steps.)
  - a. Click the **Platforms** tab to open the Platforms Home window.
  - b. On the Platforms menu, choose one of the following:
    - **My Platforms** to see a list of platform models that are owned by the user who is currently logged in.
    - **All Platforms** to see a list of all released platform models in HP CDA.
  - c. Click a platform name to open its details window.

**Note:** After you see the list, you can use the tabs that appear above it to switch between My Platforms, All Platforms and platforms you have selected as Favorites.

To make entry in the **Provision platform** wizard, perform the following steps:

1. Click the **Overview** tab, if it is not already selected.
2. Click **Provision**, on the right side of the screen.

The **Provision platform** wizard opens.

3. Enter Platform settings:
  - a. **Provisioned Platform Name:** Type the screen name for the provisioned platform.
  - b. **Stage:** Choose the lifecycle stage assigned to the provisioned platform. The menu only offers choices that were selected for the platform model (during creation or edit).
  - c. **Force Automatic Backout and cleanup on any failure:** Select whether you want a

backout operation to begin automatically if the provisioning operation fails.

d. Enter **Advanced Options**:

**Note:** HP strongly recommends that you enter a job timeout limit of approximately how long you expect provisioning to take. Otherwise, errors in provisioning can cause the operation to hang indefinitely.

Select **Job Timeout**, and type a time and unit of measurement. If provisioning takes longer than the timeout limit, the operation cancels out.

Select **Backout Timeout**, and type a time and unit of measurement. If a backout operation takes longer than the timeout limit, the operation cancels out.

You can use the platform provisioning Report to confirm that a timeout is the reason an operation failed. See "[Using the Platform Provisioning Report](#) " on next page.

4. Click **Next**.

5. The wizard provides a dialog box for each template associated with the platform model. Different templates have different entry requirements.

**Caution:** At this time, HP Cloudsystem Matrix does not accept Unicode characters. If you are provisioning an HP Cloudsystem Matrix template, use standard English characters when you enter **Hostname Replacement Text** and **Service Name**. If you make Unicode entry, provisioning will fail.

If:

- A template does not have any associated parameters, the dialog box does not provide entry fields.
- Entry fields are empty, you must enter required information, as indicated by the red asterisk (\*).
- Default parameters display, you can accept or modify them.
- You want to use a stored parameter set, click **Load Parameters** and select the set. You can edit stored values as needed.

Click **Next**.

When you have entered all required parameters for each template, a Summary window opens.

6. Click **Finish**.

Provisioning begins.

7. *Optional:* Use the platform provisioning Report to see provisioning operations or cancel provisioning. See "[Using the Platform Provisioning Report](#)" below.

**Note:** When you use the CLI to edit or delete an infrastructure template, HP CDA does not immediately detect the change. The template's status does not change until the Template Status Updater task runs. If you attempt to provision a platform that uses a deleted template before the update task runs, provisioning fails. You see a message that the template cannot be found.

## Using the Platform Provisioning Report

HP CDA creates a step-by-step report of your provisioning and backout activity.

To see the report while provisioning or backout is in process, complete the following steps:

1. Click **See report for more details** to open the Report window, where you see information and a status for each task.
2. *Optional:* Click **Notify Me** to receive email notification when provisioning completes.
3. *Optional:* Click **Cancel** to cancel provisioning.

To view the report after provisioning finishes, complete the following steps:

1. Click **Events** to open the Latest Events window.
2. Click an event name to open its details window.

**Note:** You can use the Provision platform wizard to set timeout options for provisioning and backout operations. If you select timeout options and the operation time exceeds the limit you set, the operation ends with a status of Failed.

If provisioning times out while a template is being provisioned, provisioning of all templates associated with the platform ends with a status of Aborted.

## Deprovisioning a Platform

To deprovision a platform, complete the following steps:

1. Display the provisioned platform you want. (Click [here](#) for steps.)

- a. Click **Platforms** to open the Platforms Home window.
- b. On the Provisioned Platforms menu, choose one of the following:
  - o **My Provisioned Platforms** to open a list of platform models that are owned by the user who is currently logged in.
  - o **All Provisioned Platforms** link to open a list of all released provisioned platforms in HP CDA.

**Note:** After you see a list, you can use the tabs that appear above it to switch between My Provisioned Platforms, All Provisioned Platforms, and platforms you have selected as Favorites.

- c. Click a provisioned platform name to open its details window.
2. Look at the number in the Deployed Applications column.

**Note:** 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 platform name.
4. The platform's Platform Deployment window opens.
5. On the Overview tab, click **Deprovision** on the right.
6. Click **OK** in the Deprovision dialog box to confirm.
7. When the deprovisioning operation starts, a Report window opens. The report lists the deprovisioning steps and their status.

**Note:** After a deprovisioning operation starts, you cannot cancel it. From the Report window, you can start a manual backout if deprovisioning fails and forced automatic backout is not enabled.

## Using Shared Platforms

Say your enterprise offers a set of common server or platform resources, but you only have a single license for each. In such cases, you are limited by an inability to share that single license amongst other platforms that you wish to provision and deploy. With HP CDA you can share and reuse common resources among multiple applications, thereby saving your company the expense of

having to purchase additional licenses every time you wish to deploy an application amongst different platforms.

Specifically, HP CDA supports the sharing of simple platforms. A *simple platform* (also known as a *logical platform*) is a platform that consists of a single infrastructure template or design. By default, all HP CDA platform designs are simple. You can share a simple platform after it has been provisioned. After a simple platform has been shared, it can be imported into a composite platform model design.

You can reuse a shared platform in another platform model to create a *composite platform model* (also known as a *composite logical platform*). To reuse a shared platform in another model, you must add the shared platform template to a composite model design. So, a composite platform model is comprised of simple templates along with one or more shared platform templates.

Shared platforms inherit the template infrastructure of the platform upon which they are based. Thus, you cannot use Designer to modify any Deployer, Software, or Hardware and Operating System settings for a shared platform, because these settings are inherited from the shared platform.

Note: You cannot share a provisioned (realized) composite platform. In other words, once a composite platform model has been provisioned, it can not be shared again.

When sharing a platform in HP CDA, be sure to adhere to the following guidelines:

- You can share or reuse any realized platform in HP CDA. After the platform is shared, it can be used in other platform designs as well.
- After sharing a platform in HP CDA that template is made available within the Existing Infrastructure Provider field in the Select template dialog.

Note: This shared template is internal to HP CDA and is *not* listed in Infrastructure page.

- After sharing a platform, you can add the shared platform template to a logical platform design to create a shared composite platform design.
- Use shared platforms with caution. Any change on a shared platform can impact other users. You cannot unshare a platform until you have deprovisioned *all* platforms that are using it.

## Sharing a Platform

Typically, users need to share a platform that has common software that is required by or among multiple applications. For example, in certain cases, you want to share a database server or LDAP server, rather than purchasing additional licenses for each.

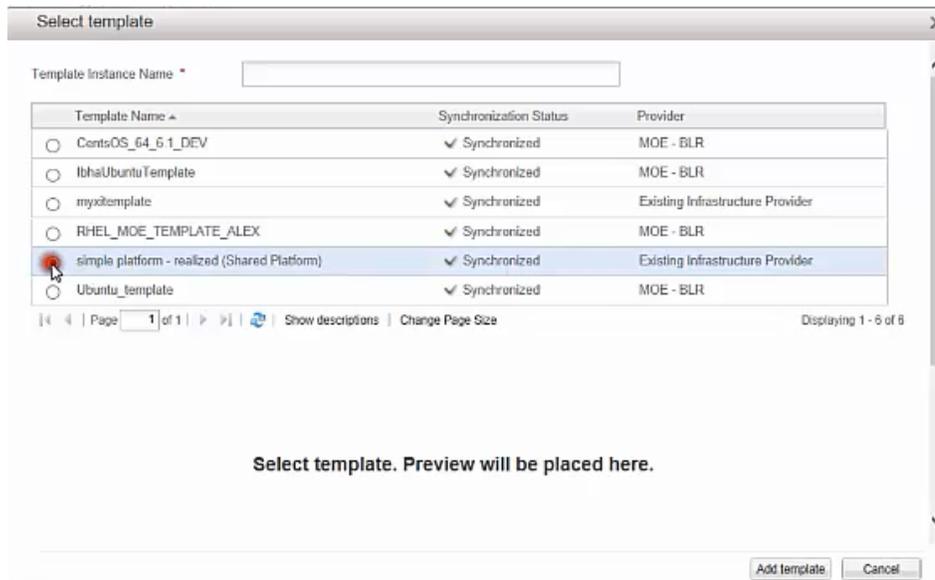
To share a platform:

1. Navigate to the **My Provisioned Platforms** page and select the platform that you want to share.

2. On the details page for the specific provisioned platform, scroll to the Platform Sharing section, and select the checkbox for Share this platform.

Note: Notice that after selecting the Share this platform checkbox, the Deprovision link on the right-hand side of your screen disappears. You cannot deprovision a shared platform until it is unshared, and you cannot unshare a platform until all platforms that are using it are deprovisioned first, with all references removed.

3. Navigate to the **My Platforms** page, and click the **Create Platform** link. Fill in the fields Basic Properties, and click **Next**
4. In the Select template dialog, click **Add Template**. Note that when you specified that the platform should be shared, HP CDA created a template and made it available for import into other composite platform designs. The Select template dialog lists all templates available for use. Locate the specific template you wish to use. Any shared templates are identified as such here.



5. Here we select `simple platform - realized`, which is a shared platform.
6. In the Template Preview dialog, click **Add template**. Then click **Next**.
7. In the Summary dialog, review the properties and verify your platform template selections. Then click **Finish**.
8. Upon completion, you are taken to the details page for the newly-created platform. Here, you can see that the template used to create this platform is a shared one. If you return to the original platform that served as the template for this new platform, you can see that it is now identified as a shared platform. All other platforms using this particular template are identified

here as well. You cannot unshare the platform until *all* references to this shared platform are removed.

9. You can now proceed with the provisioning of the newly-created platform.

## Unsharing a Platform

Note: You can unshare a platform *only* if it is no longer being used.

To view all platforms that are using a particular shared platform, select the platform and navigate to the Platform Sharing section on the details page. Links to all platforms that use this original platform are displayed. If you click any of these links, you are taken to the details page for that particular platform, where you can also see that the platform is currently being shared.

To unshare a platform:

1. If you have not already done so, deprovision *all* other platforms that share the platform you wish to unshare. Also, remove the shared template reference from all the logical platforms either by removing the shared template or deleting the platform.
2. Navigate to the **My Provisioned Platforms** page and select the platform that you wish to unshare.
3. On the details page for the specific provisioned platform, scroll to the Platform Sharing section, and deselect the checkbox for Share this platform.

# Infrastructure

Infrastructure templates serve as the foundation for platform models. Infrastructure templates define the base compute, software, storage, and network characteristics upon which a platform model is built.

**Note:** If you have Administrator permission to add and configure infrastructure providers, you see a note and **here** link in the Application Flow box. Click **here** to open the Administration Plugin tab where you can add plugins and configurations.

To be used by HP CDA, an infrastructure template must be:

- **Searched:** HP CDA can search for all infrastructure templates that reside on integrated infrastructure providers.

**Note:** Searches return all templates that the provider has made available (public). Search does not filter by any template status. However, HP CDA offers an edit function that allows you to view and work with a template in its native interface on the provider application.

- **Imported:** You must import an infrastructure template into HP CDA to use it in platform creation. Imports bring in the current version of the infrastructure template, so the templates are synchronized at the time of import.
- **Synchronized:** The HP CDA infrastructure template must match the version of the template on its home server. If someone makes a remote change to the infrastructure template, HP CDA recognizes the change and displays a status of **Remote Change**. You can manually synchronize a template at any time.

**Note:** When you use the CLI to edit or delete templates, HP CDA does not detect the change until the Template Status Updater task runs. Templates can briefly appear to be synchronized, although they have been changed or deleted.

The following topics provide information about tasks platform developers perform when working with infrastructure templates.

Topic	Information Included
<a href="#">"Launching a Provider Designer " on next page</a>	How to access an infrastructure provider designer web site where you can create infrastructure templates.

Topic	Information Included
<a href="#">"Launching a Provider Manager " on the facing page</a>	How to access an infrastructure provider management console where you can: <ul style="list-style-type: none"><li>• monitor requests for provisioned systems</li><li>• query and delete provisioned systems</li><li>• manage an infrastructure template and its components</li></ul>
<a href="#">"Creating Infrastructure Templates from the CLI " on the facing page</a>	How to use the Command Line Interface (CLI) to create infrastructure templates that use existing infrastructure.
<a href="#">"Creating Infrastructure Templates from the CLI, an Example " on page 132</a>	How to use the CLI to create an existing infrastructure template for Pet Clinic.
<a href="#">"Importing Infrastructure Templates " on page 133</a>	How to search for and import an infrastructure template from an integrated infrastructure provider.
<a href="#">"Opening Infrastructure Template Windows " on page 135</a>	How to open windows that show information about infrastructure templates.
<a href="#">"Manually Synchronizing Infrastructure Templates " on page 136</a>	How to synchronize an infrastructure template that has been imported, but subsequently changed on its home server.
<a href="#">"Using the Synchronize Template Report " on page 137</a>	How to monitor the progress of an infrastructure template import and view status messages afterward.
<a href="#">"Editing Infrastructure Templates " on page 138</a>	How to launch an interface where you can edit an infrastructure template on its home server.
<a href="#">"Deleting Infrastructure Templates " on page 138</a>	How to delete an infrastructure template from use in HP CDA.

## Launching a Provider Designer

HP CDA uses infrastructure templates that are created in a separate provider designer interface. Infrastructure developers can launch a provider designer from HP CDA, and make remote edits to infrastructure templates.

To launch an infrastructure provider designer, complete the following steps:

1. Click the **Infrastructure** tab to open the Infrastructure Home window.
2. On the Provider Designers menu, chose a provider to open its designer interface.

3. Use the designer interface to create a template (and bindings, in the case of HP Open Stack).

For documentation on HP provider designers, see [manuals](#).

4. Click to **Sign Out** or close the window.

**Note:** If you have Administrator permission to add and configure infrastructure providers, you see a note and **here** link in the Application Flow box. Click **here** to open the Administration Plugin tab where you can add plugins and configurations.

## Launching a Provider Manager

Infrastructure developers can launch a provider manager from HP CDA. Provider manager consoles typically allow you to query existing provisioned systems and monitor the status of ongoing provision requests.

To launch an infrastructure provider manager, complete the following steps:

1. Click the **Infrastructure** tab to open the Infrastructure Home window.
2. On the Provider Designers menu, chose a provider to open its designer interface.
3. Use the service portal as needed.

For documentation on HP provider managers, see [manuals](#).

4. Click to **Sign Out** or close the window.

## Creating Infrastructure Templates from the CLI

You can use HP CDA's Existing Infrastructure provider to create infrastructure templates that use existing compute resources. Compute resources (virtual, physical, or cloud-based) that have been provisioned outside of HP CDA form the basis of Existing Infrastructure templates.

Like any other infrastructure template, these templates can be used as the basis for platform models.

The basic steps involved in creating an infrastructure template are as follows:

- Represent existing compute resources as logical objects called servers.
- Group compute resources that have similar properties, such as the same OS and memory size, together by defining logical objects called server groups.

- Create logical objects that represent the infrastructure template in HP CDA, and attach the server group objects 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. We assume that 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. For information about HP CDA CLI see "[HP CDA Command Line Interface \(CLI\)](#)" on page 276

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

1. Identify the existing compute resources that will be added to the infrastructure template. 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 groups.

**Note:** Server groups are groupings of compute resources with 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 servergroup` command to create logical objects in HP CDA to represent each server group, and to add compute resource objects to the server group objects.

**Note:** The `-attachserver` option of the `existinginfrastructure add servergroup` command can be used to add compute resource objects to a server group object as it is created, or they can be added to the server group object subsequently using the `existinginfrastructure set servergroup` command.

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

**Note:** The `-attachservergroup` option of the `existinginfrastructure add template` command can be used to add server group 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 a template to HP CDA, it displays when you select **Existing Infrastructure** as the Template Provider.

## Creating Infrastructure Templates from the CLI, an Example

The following example shows how to create an infrastructure template for an application called Pet Clinic. In the example, Pet Clinic requires an infrastructure template that contains two server groups:

- App, with a single provisioned server named `app511.syslab.adam.qa.hp.com`
- DB, with a single provisioned server named `db511.syslab.adam.qa.hp.com`

Existing Infrastructure environments can be created in a variety of ways; this example uses the HP CDA CLI `set` command. For information about the HP CDA CLI see "[HP CDA Command Line Interface \(CLI\)](#)" on page 276

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

1. To create the logical template, logical server groups, and logical servers for the Pet Clinic example, run the following commands:

```
cdaexec existinginfrastructure add template --name  
PetClinicTemplate
```

```
cdaexec existinginfrastructure add servergroup --name App --ostype  
Linux
```

```
cdaexec existinginfrastructure add servergroup --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 groups with the existing template, run the following command:

```
cdaexec existinginfrastructure set template PetClinicTemplate --  
attachservergroup App --attachservergroup.DB
```

3. To associate the two existing servers with the existing server groups, run the following

commands:

```
cdaexec existinginfrastructure set servergroup App --attachserver  
app511.syslab.adam.qa.hp.com
```

```
cdaexec existinginfrastructure set servergroup 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 group 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 servergroup 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 servergroup DB --attachserver  
db512.syslab.adam.qa.hp.com --attachserver  
db513.syslab.adam.qa.hp.com
```

## Importing Infrastructure Templates

Before you can use an infrastructure template to create a platform, you must import the template from its home sever into HP CDA.

When you import, you bring in the current version of the infrastructure template, so its status is **Synchronized**.

**Note:** Searches return all templates that the provider has made available (public). Search does not filter by any template status. However, HP CDA offers an edit function that allows you to view and work with a template in its native interface on the provider application.

To import an infrastructure template, complete the following steps:

1. Click the **Infrastructure** tab to open the Infrastructure Home window.
2. Click **Import** to open the Synchronize template(s) dialog box.
3. Search for the infrastructure template:
  - a. Choose an infrastructure provider from the Template Provider menu.
  - b. *Optional:* Type the beginning of a template **Name** or **Description**. HP CDA treats missing, trailing characters as a wildcard match. Use an asterisk (\*) at any point within the search string to represent any number of wildcard characters.
  - c. Click **Search**.
  - d. You see the Templates that match your search criteria. The templates' Synchronize Status is one of the following:

**Not Synchronized Yet:** The template has been discovered on the infrastructure provider, but not imported and synchronized with HP CDA.

**Synchronized:** The template has been imported into HP CDA and matches the latest version on the template provider.

**Remote Change:** The template has been imported into HP CDA and later changed on the infrastructure provider. The template used by HP CDA is not synchronized.

**Note:** When you use the CLI to edit or delete templates, HP CDA does not detect the change until the Template Status Updater task runs. Templates can briefly appear to be synchronized, although they have been changed or deleted.

4. Select one or more templates.
5. Click **Synchronize templates**.
6. The template displays under **Imported infrastructure designs**.

*Optional:* Use the Synchronize Template Report to view import steps and status messages. Click [here](#) for information.

When you import or synchronize an infrastructure template, HP CDA provides a report of that activity.

During the import, the report lists each associated task when it begins and the task status (successful or not) when it ends.

After synchronization, the report provides an overall status of the synchronization (successful or not).

To view the **Synchronize Template Report**, complete the following steps:

1. Scroll down to the Latest Events section on the Infrastructure Home window.
2. Click the template report name.

**Note:** The report name will vary according to the status of the synchronize activity.

The report window opens.

## Opening Infrastructure Template Windows

Infrastructure templates display as one of the following:

- **Not Synchronized Yet:** The template has been discovered on an integrated server, but not imported and synchronized with HP CDA.
- **Synchronized:** the template used by HP CDA matches the version on the template's home server. You can provision a platform that uses this template.
- **Remote Change:** the template has been changed on its home server and the template used by HP CDA is out of date. You should not provision a platform using the template while it is out of synch. You must manually synchronize the template to make it usable.
- **Deleted:** The template has been deleted on the infrastructure provider. HP CDA still has a copy of the template, but when you try to provision a platform that uses the template, provisioning fails on the remote server. You should delete infrastructure templates with a status of Deleted from platform designs.

**Note:** When you use the CLI to edit or delete templates, HP CDA does not detect the change until the Template Status Updater task runs. Templates can briefly appear to be synchronized, although they have been changed or deleted.

To display infrastructure templates that have been imported into HP CDA, complete the following steps:

1. Click the **Infrastructure** tab to open the Infrastructure Home window.

You see imported infrastructure templates under the Imported infrastructure designs heading.

2. *Optional:* You can sort the list by any of the column headings (Template Name, Provider, etc.)

Infrastructure templates display the following tabs:

- **Overview:** Provides a graphic representation of the template where you can see and change certain disk and server properties. You can also access the template's native interface to edit the template. The process is the same here as in the platform Designer, except that Designer provides fields for deployer information entry, and the Overview tab does not. See "[Modifying Infrastructure Design](#)" on page 81
- **Platforms:** View a list of platforms that use the template.
- **Specification:** Template information that you want to save and share with other users.
- **Relationships:** Add or remove outgoing and incoming relationships.
- **Access Rights:** Manage read and write access rights to the template.
- **History:** View template revisions.
- **Events:** View information about the major events (such as creation and modification) for the template.

To open a list of discovered infrastructure templates, complete the following steps:

1. Click the **Infrastructure** tab to open the Infrastructure Home window.
2. Click **Import** to open the Synchronize template(s) dialog box.
3. Choose an infrastructure provider on the Template Provider menu.
4. *Optional:* Type the beginning of a template **Name** or **Description**. HP CDA treats missing, trailing characters as a wildcard match.
5. Click **Search**.

You see templates that match your search criteria.

## Manually Synchronizing Infrastructure Templates

To manually synchronize an infrastructure template, complete the following steps:

1. Click the **Infrastructure** tab to open the Infrastructure Home window.
2. Select the template and click **Synchronize template(s)**.
3. Click **OK** in the warning window to confirm the import.

You see the infrastructure template with a Synchronization Status of Synchronized.

*Optional:* Use the Synchronize Template Report to view import and synchronization steps and status messages. Click [here](#) for information.

When you import or synchronize an infrastructure template, HP CDA provides a report of that activity.

During the import, the report lists each associated task when it begins and the task status (successful or not) when it ends.

After synchronization, the report provides an overall status of the synchronization (successful or not).

To view the **Synchronize Template Report**, complete the following steps:

1. Scroll down to the Latest Events section on the Infrastructure Home window.
2. Click the template report name.

**Note:** The report name will vary according to the status of the synchronize activity.

The report window opens.

**Note:** Searches return all templates that the provider has made available (public). Search does not filter by any template status. However, HP CDA offers an edit function that allows you to view and work with a template in its native interface on the provider application.

## Using the Synchronize Template Report

When you import or synchronize an infrastructure template, HP CDA provides a report of that activity.

During the import, the report lists each associated task when it begins and the task status (successful or not) when it ends.

After synchronization, the report provides an overall status of the synchronization (successful or not).

To view the **Synchronize Template Report**, complete the following steps:

1. Scroll down to the Latest Events section on the Infrastructure Home window.
2. Click the template report name.

**Note:** The report name will vary according to the status of the synchronize activity.

The report window opens.

## Editing Infrastructure Templates

HP CDA provides access to templates' native interface, where you can edit the templates. You can access the interface from the platform Designer tab or the Infrastructure tab.

To edit an infrastructure template from the Infrastructure tab, complete the following steps:

1. Click the **Infrastructure** tab to open the Infrastructure Home window.
2. Click the infrastructure template's name to open its details window.
3. You see the Overview tab, which displays a graphic representation of the template.
4. Click **Edit in Designer** on the right.
5. The template opens in its native designer, and you can make edits.
6. Click to **Sign Out** or close the designer window.

**Note:** After you edit an infrastructure template, its HP CDA status will be **Remote Change**.

## Deleting Infrastructure Templates

You can delete an infrastructure template from HP CDA if it is not in use. The template remains unchanged in its native interface (such as HP Cloudsystem Matrix).

You cannot delete an infrastructure template that is used in a platform. You must first delete the template from all platform designs that use it.

To delete an infrastructure template, complete the following steps:

1. Click the **Infrastructure** tab to open the Infrastructure Home window.
2. Select the template and click **Delete**.
3. Click **OK** in the warning window to confirm.

The infrastructure template is removed from the list of imported infrastructure designs.

## Monitoring Administration

Application monitoring shows you the current operational state of your deployed applications and provides information that helps ensure that those applications perform effectively, and as expected.

With HP Continuous Delivery Automation (HP CDA), you can create and deploy a *monitoring policy*, which is a unit of deployment that contains multiple monitors and, possibly, pre-written advisory rules. A *monitor* generates status events when measurements cross defined threshold boundaries, and provide a way to keep data collection in synch. A *monitoring tool* is the actual software that performs the monitoring (such as HP Diagnostics or HP SiteScope). One monitoring policy may specify that HP CDA monitor CPU utilization, while another generates events if the specified latency thresholds are exceeded. A *monitoring server* is a server that provides monitoring services and is the context from which you define connection parameters to your monitoring tools.

The HP CDA application model and the monitoring policy are integrated and deployed/un-deployed together with the infrastructure and application components.

The following topics provide information about tasks monitor administrators perform when working with monitoring tools.

Topic	Information Included
<a href="#">"Using Agent-Based vs. Agent-Less Monitoring Tools " below</a>	Information about monitoring tools.
<a href="#">"Implementing Monitoring " on the facing page</a>	How to set up monitoring tools in your environment.
<a href="#">"Monitoring Applications in Amazon Elastic Compute Cloud " on page 156</a>	How to define connections between HP CDA, and HP Diagnostics and HP SiteScope to monitor Amazon Elastic Compute Cloud (Amazon EC2) environments.

## Using Agent-Based vs. Agent-Less Monitoring Tools

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

- Agent-less

Agent-less tools 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

Agent-based tools require you to install software (i.e., the agent) on the monitored system.

Each approach has its strengths, and using a combination of monitoring tools is often desirable. For example, agent-based solutions tend to require fewer configuration tasks than agent-less solutions. However, this is a broad categorization and some monitoring tools exhibit aspects of both agent-based and agent-less approaches.

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.

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.

HP CDA is the deployment mechanism for agents, and automates this deployment. HP CDA uses its monitoring configuration to dynamically configure the underlying monitoring tools.

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 is updated and displayed.

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

## Implementing Monitoring

To configure monitoring, complete the following tasks:

["Step 1: Identify the Monitoring Tools in Your Environment " on next page](#)

["Step 2: Configure Your Monitoring Tool to Send Events to HP CDA " on next page](#)

["Step 3: Configure HP CDA to Communicate with the Monitoring Tool " on next page](#)

["Step 4: Define Monitors " on page 142](#)

["Step 5: Define Monitoring Policies " on page 145](#)

["Step 6: Assign Monitoring Policies to Application Topologies " on page 150](#)

["Step 7: Review the State of Monitored Applications " on page 155](#)

## Step 1: Identify the Monitoring Tools in Your Environment

There are several monitoring tools available that enable you to monitor application availability and performance. HP CDA Monitoring is designed to be open and extensible, and integrates with the following monitoring tools:

- HP SiteScope
- HP Diagnostics
- HP Operations Manager (HP OM)
- Nagios

**Note:** For integrations with additional monitoring tools, contact your HP representative.

## Step 2: Configure Your Monitoring Tool to Send Events to HP CDA

Click the monitoring tool you are using to view information for configuring the tool to send events to HP CDA:

- ["Configuring HP Diagnostics to Send Events to HP CDA " on page 343.](#)
- ["Configuring HP SiteScope to Send Events to HP CDA " on page 344.](#)
- ["Configuring Nagios to Integrate with HP CDA " on page 348.](#)
- ["Configuring HP Operations Manager to Send Events to HP CDA " on page 346.](#)

## Step 3: Configure HP CDA to Communicate with the Monitoring Tool

After configuring your monitoring tools to send events to CDA, you must create a monitor server for each tool as well. Monitor Servers contain parameters that enable HP CDA to connect to a monitoring tool.

To create a monitor server:

1. In HP CDA, navigate to the Monitoring tab.
2. Click **Monitor Servers** in the left panel.
3. Click **Create**.
4. Enter a Name for the monitor server, and optionally, enter a Description.
5. Click **Select**.
6. Select the Monitor Provider Artifact (monitoring tool) being used for this particular definition.
7. Click **Add**.
8. Verify that the Monitor Provider Definition displays the monitoring tool you selected in step 6.
9. Click **Next**.
10. Enter the connection parameter values for the tool-specific connection.
11. Click **Finish**.

The monitor server is now available for the specified domain.

## Step 4: Define Monitors

HP CDA provides two ways to configure monitors:

- Use SiteScope monitors, which are created by importing HP SiteScope templates
- Use manually-configured monitors, including HP Diagnostics, HP OM, and Nagios

To define monitors for HP SiteScope, you must first import the SiteScope monitoring template.

## *Creating Monitors by Importing HP SiteScope Monitoring Templates*

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

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

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, then they must have a tag set for sending events to HP CDA. They must also use HP CDA Event Mapping for HP CDA to receive status events from the HP SiteScope monitors.

To import a monitoring template from an HP SiteScope server:

1. In HP CDA, navigate to the Monitoring tab.
2. Click **SiteScope Templates**.
3. Click **Import Templates**.
4. Browse and select the individual templates you wish to import. HP SiteScope templates are imported from the designated HP SiteScope Server.

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

5. Click **Next**.
6. Click **Finish**.
7. To view and verify that the templates you selected were imported successfully, click **SiteScope Templates** in the left pane.
8. Click an imported template to view its parameters. HP CDA parameters have been created for each HP SiteScope variable in the template. The syntax for HP SiteScope variable names has been altered to accommodate the HP CDA format.

HP 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 an HP SiteScope template cannot be shared. Only the entire template can be shared. This follows the HP 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. You cannot change the name of any parameters in the template.

## ***Creating Monitors Manually***

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

Before creating monitors, you should have expert knowledge of your monitoring tool. HP CDA includes out-of-box monitors for each supported tool. Before creating a monitor, check to see if any of the provided monitors—which have been tested and include examples—will serve your needs.

**Note:** You can manually create a monitor for HP Diagnostics, HP OM, and Nagios.

To create a monitor manually:

1. In HP CDA, navigate to the **Monitoring** tab.
2. Click **Monitors** in the left panel.
3. Click **Create**.
4. Specify a Name for the monitor
5. Specify a name in 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 text in 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 in HP CDA. 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}](http://@{diag_server}:@{diag_port}/topaz/appmon/tcAPI?action=showTxBd&customer_name=Default%20Client&view=virtual_machines_screen&select_entity=@{probeid})

**Note:** There are two types of Diagnostics monitors: *infrastructure* (for example, CPU and Memory) and *application* (for example, Latency). To create the contextual launch URL, you must provide a hostname for infrastructure type monitors, whereas for application monitors, the `probeid` must be provided. Thus, there are two types of contextual launch URL templates, one ending with `view=hosts_screen&select_entity=@{hostname}`, the other ending with `view=virtual_machines_screen&select_entity=@{probeid}`.

Because the values of `@{hostname}` and `@{probeid}` are automatically provided (substituted) by HP CDA internally at deployment time, you do *not* need to provide them here. For more information about URL launching, refer to the documentation of the specific monitoring tool. For more information about HP Diagnostics, see the *HP Diagnostics User's Guide* or Online Help that describes 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 application 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 wish to add.
12. Click **Add**.
13. Click **Next**.

The Deployment Parameters display. These parameters apply to all monitors for the tool being used. Some tools, such as HP Diagnostics, collect data in various ways, so when you create a monitor with HP Diagnostics, you can have a deployer for system metrics, another for monitor metrics and so on, as a way to help organize all the parameters and metrics for that category.

14. Enter values for the Deployment Parameters.

If the monitor will require additional parameters to be filled in when it is deployed, then you can add them in the Additional Parameters section. For example, if you want a monitor to generate threshold events, then you must add a parameter that specifies the high and low values for threshold events. Some parameters are specific to the type of monitor being used. For example, for all Nagios monitors, you need to be familiar with the `nag_check` parameter.

15. Click **Finish**.

## Step 5: Define Monitoring Policies

Monitoring policies define the set of monitors (and monitor servers) to use when you deploy an application, and 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 the monitors that it will reference. After you define a policy, the monitors can be referred to (shared) by multiple monitoring policies.

**Note:** When using out-of-the-box monitoring policies, be aware that you cannot specify a value of zero for either event observed duration or event evaluation duration. Also note that the value

used in both of these system settings (mon.event.observed.duration and mon.event.evaluation.duration) is *milliseconds*.

To define monitoring policies:

1. Navigate to the Monitoring tab.
2. Click **Monitoring Policies** in the left panel.
3. Click **Create**.
4. Enter a unique Name.
5. Click **Next**.
6. Click **Add** to add one or more monitors (also referred to as "Monitor Artifact Type") to the policy.
7. Check the box next to each monitor (or HP SiteScope Template) you wish to add to the policy.
8. Click **Add**.
9. Click **Finish**.

## Out-of-Box Monitoring Policies

HP CDA provides numerous out-of-box monitoring policies that enable users to implement common monitoring tasks without having to define, build, and create new monitoring policies from scratch. Of course, you can explicitly create new policies as well. The following table identifies all of the out-of-box monitoring policies provided with HP CDA.

Policy Name	Description	Monitors Included
<b>HP Diagnostics Policies</b>		
ASP.NET Memory Utilization	Use to monitor memory utilization.	<ul style="list-style-type: none"> <li>• ASP.NET GC Time Spent in Collections</li> <li>• System Memory Utilization (percent)</li> </ul>

Policy Name	Description	Monitors Included
ASP.NET Requests	Use with HP Diagnostics version 9.25 or later. This policy monitors requests per second and rejected requests. For example, when a user requests a web page, the request may be rejected for a variety of reasons. This policy monitors the rate of rejection of such requests.	<ul style="list-style-type: none"> <li>• ASP.NET Requests/Second</li> <li>• ASP.NET Requests Rejected</li> </ul>
ASP.NET Scaling	Use to monitor CPU, latency, and requests per second.	<ul style="list-style-type: none"> <li>• ASP.NET Requests/Second</li> <li>• System CPU</li> <li>• Web application visit latency (milliseconds)</li> </ul>
System CPU with Duration Rules	<p>Use to monitor CPU. This policy includes two duration rules:</p> <ol style="list-style-type: none"> <li>1. This rule advises users to increase the number of servers in the corresponding server group. Specifically, it advises users to increase the number of servers if the CPU usage is high (accumulatively) within the observed time window of the evaluation period.</li> <li>2. This rule advises users to decrease the number of servers if CPU is low (accumulatively) within the observed time window of the evaluation period.</li> </ol> <p>You can set the observed and evaluation time periods in <b>Administration &gt; Configuration &gt; System Variables</b>.</p>	System CPU

Policy Name	Description	Monitors Included
Web Application Scaling with Aggregation Rules	<p>Use to monitor tier scaling for HP Diagnostics. This policy includes two aggregation rules:</p> <ol style="list-style-type: none"> <li>1. This rule advises users to increase the number of servers in the corresponding server group if all CPUs and request/second rates are high and latency is high.</li> <li>2. This rule advises users to decrease the number of servers if all CPUs and request/second rates are low and none of the latency is high.</li> </ol> <p>These rules will not be triggered if any of the CPU, latency, and request/second monitors are removed from this policy.</p>	<ul style="list-style-type: none"> <li>• System CPU</li> <li>• Web application visit latency (milliseconds)</li> <li>• Web application visit rate (count)</li> </ul>
<b>Nagios Policies</b>		
Nagios Linux	Use to monitor CPU, memory, and disk utilization on the Linux platform.	<ul style="list-style-type: none"> <li>• Nagios CPU Utilization (Linux)</li> <li>• Nagios Disk Utilization (Linux)</li> <li>• Nagios Memory Utilization (Linux)</li> </ul>
Nagios MySQL Linux	Use to test connections to MySQL on the Linux platform.	MySQL Connection
Nagios Tomcat Linux	Use to check the Tomcat/manager/status page for excessive memory usage or an excessive number of threads being used.	Tomcat Memory and Threads Usage
Nagios Windows	Use to monitor CPU, memory, and disk utilization on the Windows platform.	<ul style="list-style-type: none"> <li>• Nagios CPU Utilization (Windows)</li> <li>• Nagios Disk Utilization (Windows)</li> <li>• Nagios Memory Utilization (Windows)</li> </ul>
<b>HP SiteScope Policies</b>		

Policy Name	Description	Monitors Included
Web Application Scaling with Ratio Rules	<p>Use for tier scaling using HP SiteScope. This policy includes two aggregation rules with ratios:</p> <ol style="list-style-type: none"> <li>1. This rule advises the users to increase the number of servers in the corresponding server group if 50% of the CPUs and Memory are high.</li> <li>2. This rule advises users to decrease the number of servers if 70% of the CPUs and Memory are low.</li> </ol> <p>These two rules will not be triggered if any of the CPU and Memory monitors are removed from this policy.</p>	Basic Ubuntu template
Web Application Scaling with Single Monitor Rules	<p>Use for tier scaling using HP SiteScope Latency monitor only. The policy contains scale out and scale in rules based on the single monitor.</p> <ol style="list-style-type: none"> <li>1. This rule specifies that the user is should increase the number of servers if the latency is high.</li> <li>2. This rule specifies that the user should decrease the number of servers if the latency is low.</li> </ol>	URL Latency
<b>HP Operations Manager Policies</b>		
Oracle Database Health Status	<p>This policy is part of the HP Operations Manager Oracle Smart Plugin (SPI), and monitors the overall health status of the Oracle instance running on the managed node.</p>	Oracle Database Monitor
System Infrastructure Health Status	<p>This policy is part of the HP Operations Manager Infrastructure SPI, and shows the overall health status of the system infrastructure.</p>	System Infrastructure Monitor

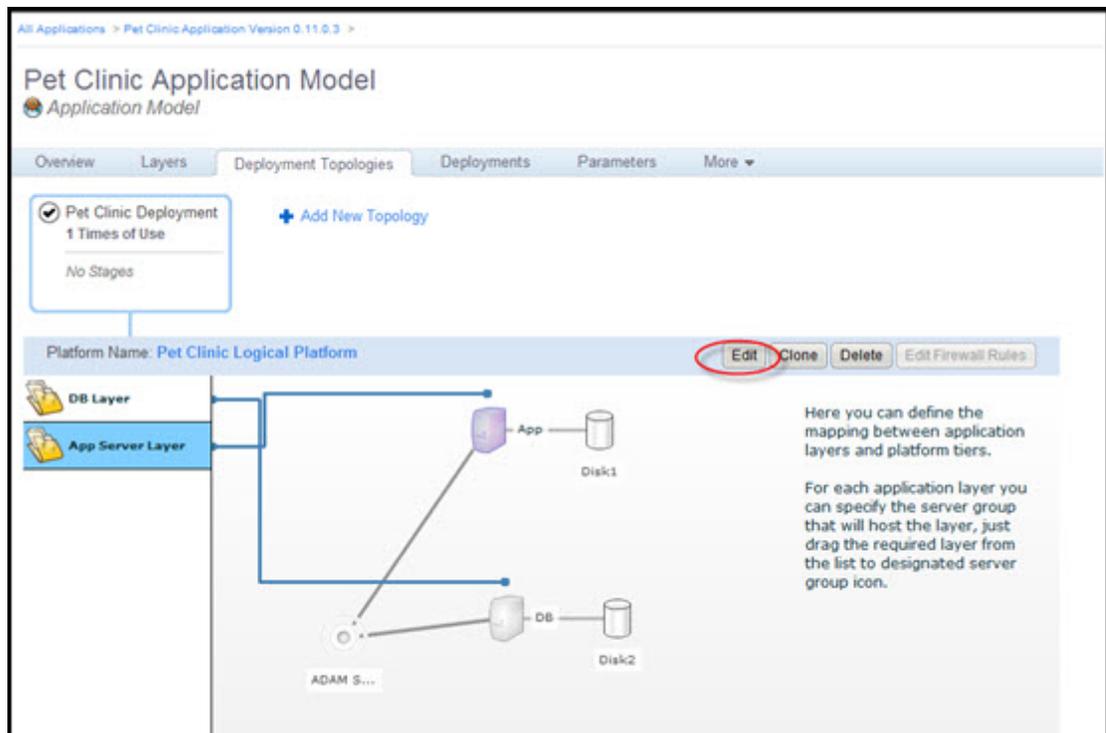
Policy Name	Description	Monitors Included
WebLogic Application Server Health Status	This policy is part of the HP Operations Manager WebLogic Application Server SPI, and monitors the overall health status of the WebLogic Application Server.	<ul style="list-style-type: none"> <li>Client Time Out for EJB</li> <li>WebLogic Application Server Monitor</li> </ul>
JBoss Application Server	This policy is part of the HP Operations Manager JBoss SPI (JBSSPI), and is configured with the JBoss monitor, and shows the application server health status and its resources.	JBoss Application Server Health Status Monitor
Microsoft SQL Server Health Status	This policy is part of the HP Operations Manager Microsoft SQL Server SPI, and shows the overall health status of the Microsoft SQL Server instance running on the managed node.	Microsoft SQL Server Monitor
WebSphere Application Server	This policy is part of the HP Operations Manager WebSphere Application Server Health Status SPI (WBSSPI), and is configured with the WebSphere monitor, which shows the application server health status and resources.	WebSphere Application Server Health Status Monitor

## Step 6: Assign Monitoring Policies to Application Topologies

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

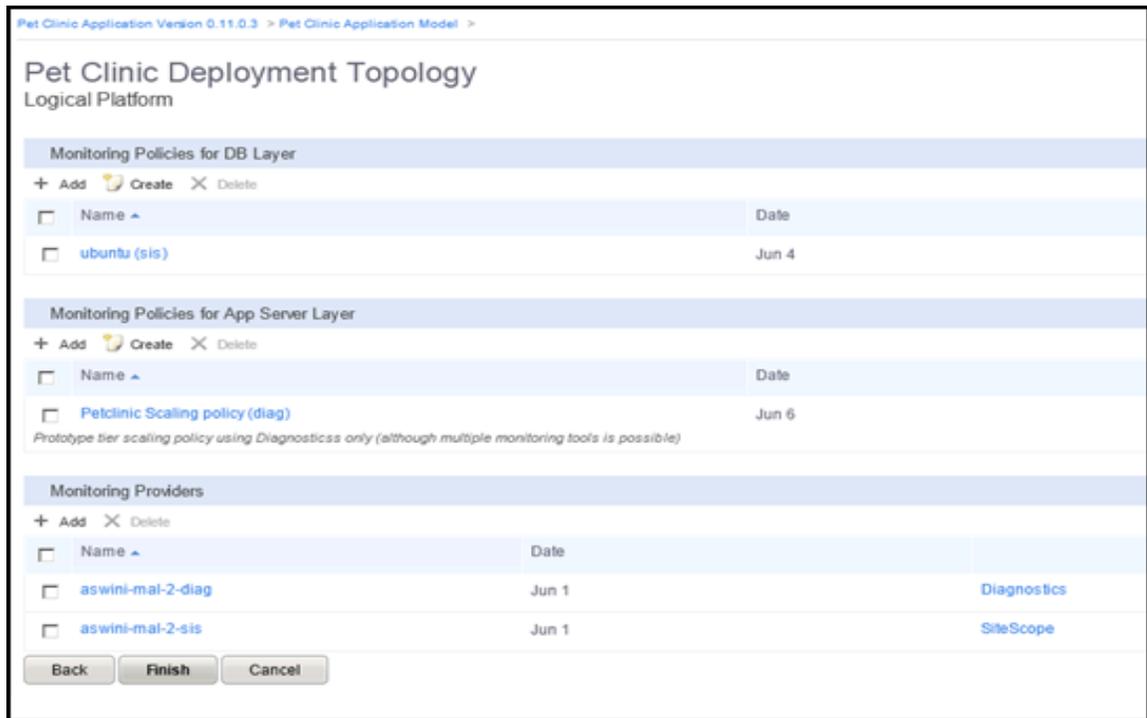
1. Navigate to the Applications tab.
2. In the left panel, click **All Applications**.
3. Select an Application version from the list.
4. Click **Application Models**.
5. Select the Application.
6. Click **Deployment Topologies**.

7. Select a Deployment topology.
8. Click **Edit**.



A wizard appears on the right panel.

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

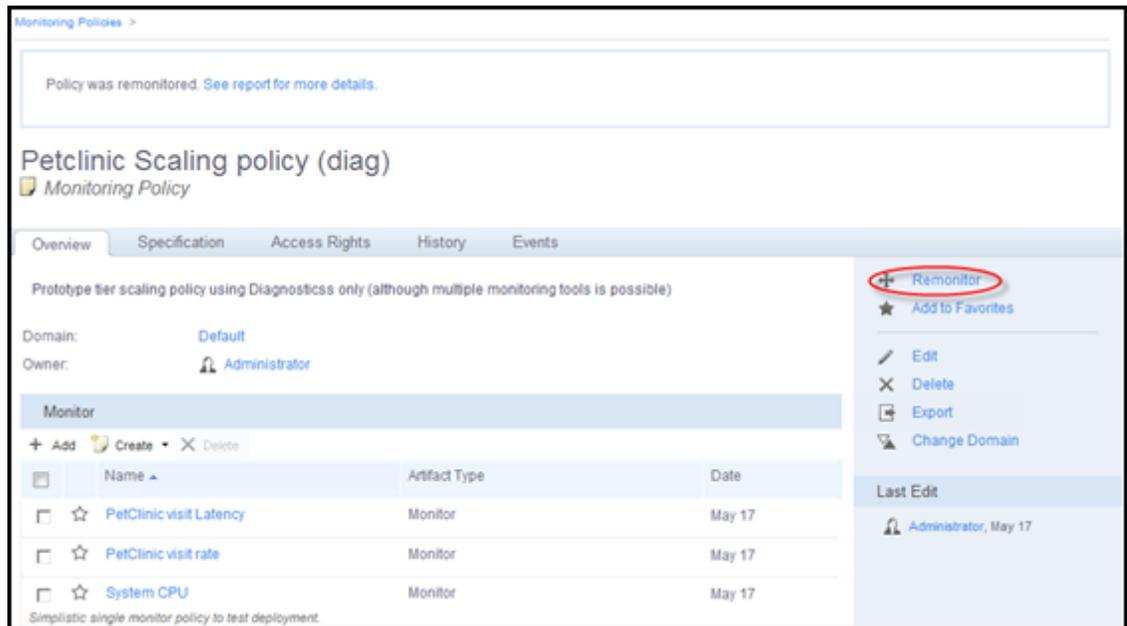


## Remonitoring

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. Navigate to the Monitoring tab.
2. Click **Monitoring Policies** in the left panel.
3. Click the policy you wish 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 parameters for the monitors in the policy.

The policy will be redeployed everywhere that it has been deployed previously.

## ***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 return the layers of the application to a consistent state.

## ***Aggregating Monitoring Metrics within Tiers***

After you have already provisioned and deployed an application, you can specify that the monitoring metrics from all servers are aggregated into a single server group view for the entire application tier.

In cases where you have many server groups—perhaps dozens or hundreds of servers—it can be a challenge to keep track of the monitors for each and every one. When you aggregate your monitoring metrics, all individual status indicators and events are presented together, thereby making it easier to scan the HP CDA user interface to quickly identify any resource issues.

To aggregate monitoring metrics across the application tier, simply use a monitoring policy that has already been configured *with* aggregation rules.

You can modify the properties (in this case, the directory path) used by the aggregate advisory script by navigating to **Administration > Configuration > System Settings** and editing the values in the system parameters `monitor.action.script.file.path` and `monitor.action.script.working.directory`.

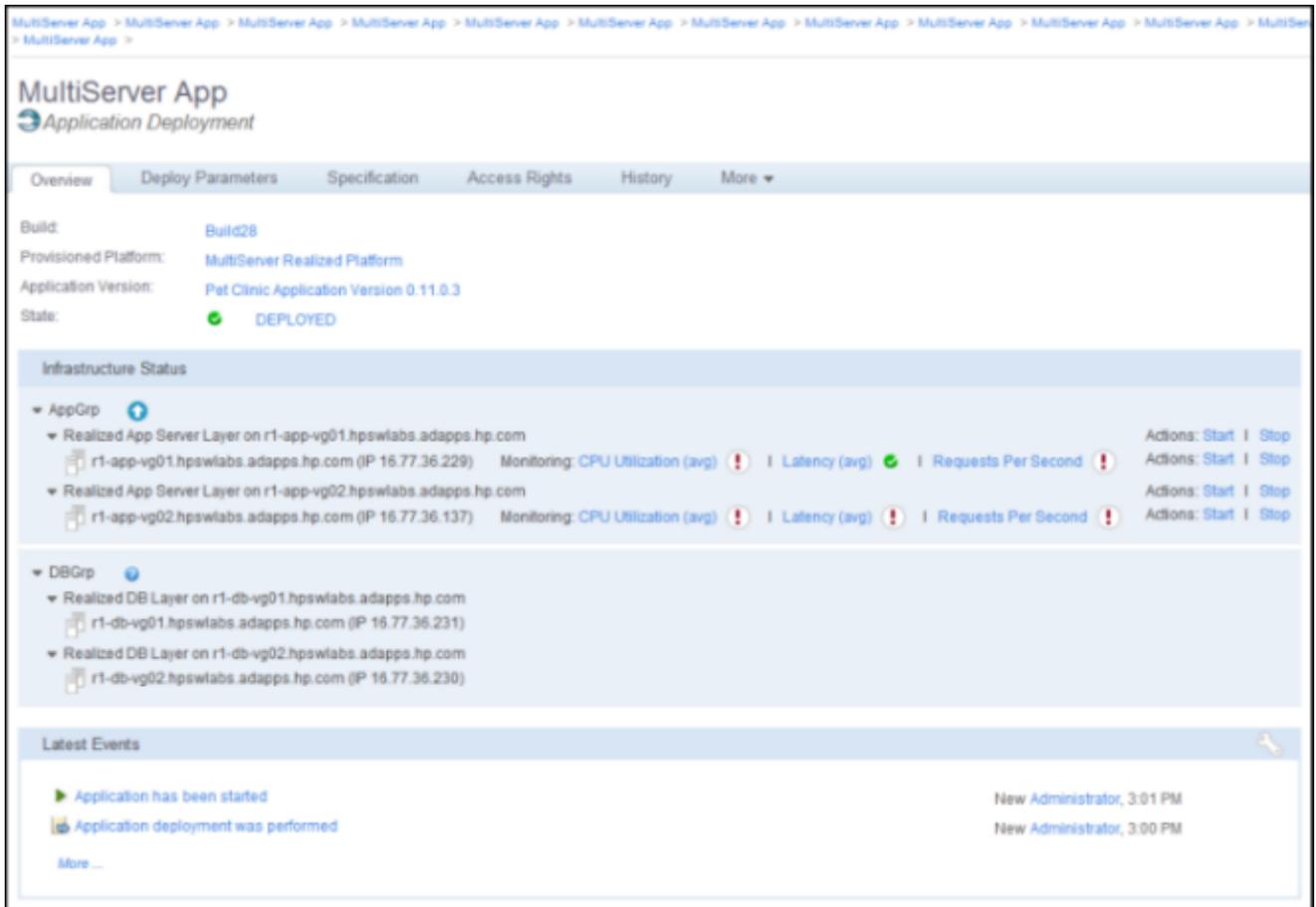
**Note:** You cannot aggregate metrics across the application tier using a monitoring policy that does not have rules defined.

### ***Scenario: Monitoring Metrics Aggregation***

In this scenario, the MultiServer App group has been deployed with monitors where thresholds have been defined for CPU utilization, and latency requests per second (RPS). You can see that for the AppGrp, the defined CPU thresholds have been exceeded for both servers, and that the defined latency threshold for one server has been exceeded as well. The up arrow icon is an indicator of the aggregate status for the entire group. In this case, the status for the group advises the user to add additional servers to the server group.

Here, it is easy for persons viewing the status to quickly identify an event and address it by increasing CPU or server resources.

In the second group, DBGrp, you can see that no monitors have been deployed; hence, no advisory appears.



## Step 7: Review the State of Monitored Applications

To review the state of monitored applications:

1. Navigate to the Applications tab.
2. Click **All Applications** in the left panel.
3. Select an application.
4. Select a Deployment (assuming the application has been previously deployed).
5. Under the Infrastructure Status section, monitors display for each system upon which the application is deployed. These systems are organized by the layer of the application. Monitor status (displayed as status change events) is received from the monitoring tools. The following table displays the monitoring status definitions displayed in HP CDA.

Icon	Status	Description & Corrective Action
	High	The monitored value exceeds the defined high threshold value.
	Normal	The monitored value is between the high and low threshold values that were configured.
	Low	The monitored value is below the defined low threshold value.
	Unknown	Unknown status. HP CDA has not received a status event for this monitor.

The following table displays the *aggregate* status definitions displayed in HP CDA.

Icon	Status	Description & Corrective Action
	Increase	Increase or add server resources.
	Maintain	The server group status is in a maintain state based on the correlation rules defined in the monitoring policy—there is no need to modify resources at this time.
	Decrease	Remove server resources.
	Unknown	Unknown status. Either HP CDA has not received a status event for this group, or no monitor is deployed for this group.

- (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.

**Provision Platform**

**Provision settings**

Provisioned Platform Name: \*

Stage:

Force Automatic Backout and cleanup on any failure:

**Parameters**

■ rmoe.ftc.adam.qa.hp.com

arm.io.hostname.replacement.text: \*

arm.io.notification.email:

arm.io.serverpool.list:

arm.io.service.lease.period:

arm.io.service.name: \*

■ Diagnostics Probe

diagnostics.application.name: \*  ...

diagnostics.server.host:  ...

diagnostics.server.port: \*  ...

## 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 rely on VPNs, which 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.

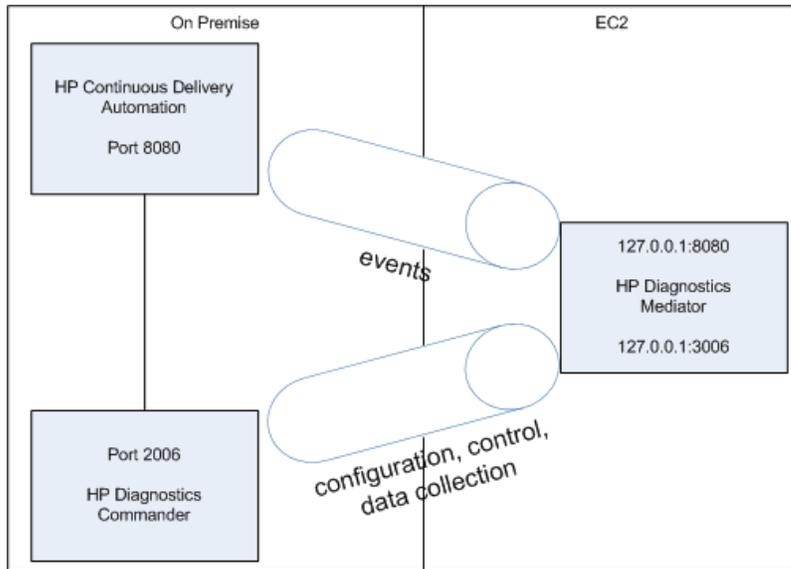
Any ports on an Amazon EC2 server that will be accessed from an on-premise system must be opened in an Amazon EC2 Security Group, otherwise firewalls will block access. The default port for SSH is 22.

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.



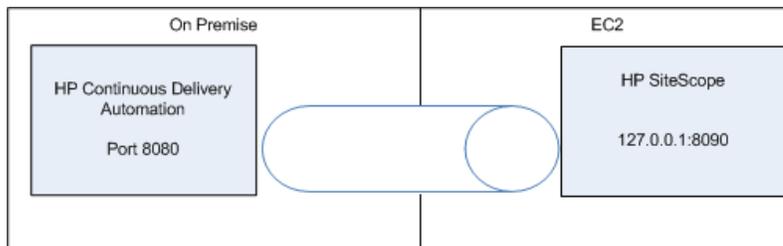
## Using SSH Tunnels with HP SiteScope

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.

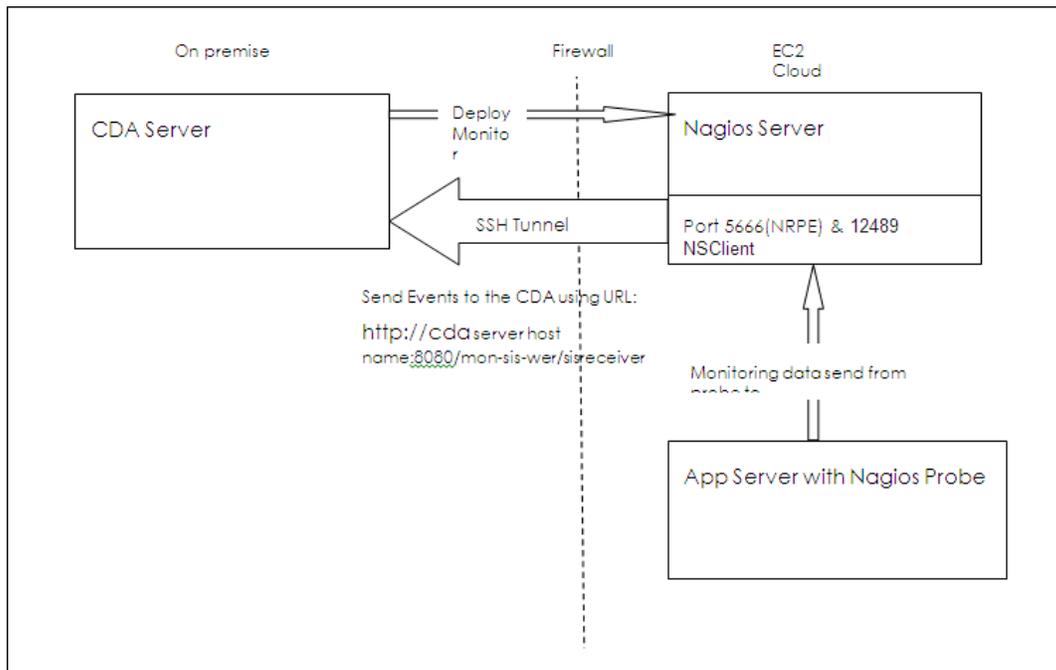
The following diagram depicts HP SiteScope sending events to an HTTP Recipient at localhost (127.0.0.1) port 8090. This traffic is forwarded to port 8080 on the HP CDA system via the SSH tunnel.



## 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.



# Administration

HP CDA Administration can be broadly divided into the following categories:

- **Managing Content:** Administrators create tenant and domain structures, and maintain the data within those structures.
- **Managing Users:** Administrators create the artifacts that represent users and assign them to roles and groups.
- **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 assign users are 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.
- **Integrating External Services:** Administrators upload and maintain plugins for platform provisioning and application deployment. Administrators are responsible for the configuration of provisioning and deployment. Administrators can also use the Command Line Interface (CLI) to integrate existing infrastructure.
- **Managing Configurations and Systems:** Administrators can use customizable configuration to control nearly every aspect of HP CDA's behavior.

The following topics provide information about administrator tasks.

Topic	Information Included
<a href="#">"Managing Content " on next page</a>	How to work with tenants, domains, archives, lifecycles, policies and assertions, tags and default administrator tasks. Includes information about working with domains when HP CDA and HP ALM are integrated.
<a href="#">"Managing Users, Groups and Roles " on page 211</a>	How to enable and disable users, and grant them roles and permissions. How to create groups and add members. How to create roles that are in addition to HP CDA default roles.
<a href="#">"Managing Security and Access Control " on page 231</a>	How to manage domain and role-based access rights.

<a href="#">"Integrating External Services " on page 232</a>	How to upload and configure application deployment and platform provisioning plugins. Includes information on using the Command Line Integration to integrate existing infrastructure.
<a href="#">"Managing Configurations and Systems " on page 251</a>	How to use customizable configuration that can control nearly every aspect of HP CDA's behavior.
<a href="#">"Running Administration Scripts " on page 256</a>	How to run Administration scripts.

## Managing Content

The most important content management concept in HP CDA is the use of domains. Administrators can create a domain structure that represents their 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 catalog 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.

The following topics provide information about tasks administrators perform to manage content.

Topic	Information Included
<a href="#">"Organizing by Tenant and Domain " on the facing page</a>	How to create, delete and change properties of tenants and domains. Includes information about how to work with domains when HP CDA and HP ALM are integrated.
<a href="#">"Importing and Exporting Archives " on page 177</a>	How to import and export model archives.
<a href="#">"Managing Lifecycle Processes " on page 180</a>	How to create and define (stages, tasks, approvers, etc.) lifecycle processes.
<a href="#">"Managing Policies and Assertions " on page 190</a>	How to create and delete technical policies.
<a href="#">"Adding, Deleting and Modifying Tags " on page 194</a>	How to create tags to describe and define requirements and capabilities of applications and platforms.
<a href="#">"Adding and Running Administration Tasks " on page 207</a>	How to create default administration tasks for deployment, deployment data, change management and custom tasks.

## Organizing by Tenant and Domain

You can configure HP CDA for multi-tenancy. With multi-tenancy, you can create tenant structures to service multiple client organizations (tenants) by partitioning client specific data and configurations. Multi-tenancy must be enabled at install time. Refer to “HP CDA Installation and Configuration Guide”.

Administrators can assign users to specific roles in specific tenants. Client organizations can use assignments 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.

### Managing Domains

As an administrator, you can use HP CDA to create a domain structure that reflects your business organization. Each artifact in your catalog belongs specifically to one domain, and you can assign users to specific roles in specific domains. When you compartmentalize your catalog in this way, you restrict the access and visibility of data to only the users who need it.

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 an administrator can create and delete domains.
- **Managing a Domain:** An 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 domain's default role

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. An 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 that users who are not assigned to a specific working domain can 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 desired domain.

The following topics contain information for administrators when they work with tenant and domain tasks.

Topic	Information Included
<a href="#">"Creating and Deleting Tenants " below</a>	How to create and delete tenants.
<a href="#">"Creating and Deleting Domains " on the facing page</a>	How to create and delete domains.
<a href="#">"Using Domains and Repositories " on page 168</a>	Information about domain structure and repository data.
<a href="#">"Using Domains When HP CDA and HP ALM are Integrated " on page 171</a>	Information about how to create artifacts and deployments in HP CDA so that HP ALM can discover them.
<a href="#">"Disabling Default Domain Sign-In " on page 171</a>	How to change sign-in rules so that users do not sign in to a default domain—users need a specific domain assignment to sign in.
<a href="#">"Setting Default Tenant or Domain Access Rights " on page 172</a>	How to set read and write permissions at either the tenant or domain level.
<a href="#">"Adding Parameters to Tenants and Domains " on page 175</a>	How to add tenant and domain level parameters.
<a href="#">"Adding Provisioning Keys to a Centralized Key Store " on page 175</a>	How to add authentication keys to a centralized store. The keys can subsequently be added to platform plugin configurations.
<a href="#">"Exporting Tenant or Domain Content " on page 176</a>	How to export domain content.

## ***Creating and Deleting Tenants***

To create tenants, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click **Create Tenant** to open the Create Tenant window.
3. Type tenant information in the text boxes:

**Name:** for the new tenant.

**Description:** an optional description.

Select a **Tenant Administrator:** option, either:

- **Create a new domain administrator:** Your **Name** entry becomes the administrator.
- **Select existing user:** You are prompted to search and enter an existing user after the **Save**.

**Login Name:** following the on-screen instructions.

**Password** and **Retype Password:** following the on-screen instructions.

4. Click **Save**.

To delete a tenant, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, select a tenant.
3. Click **Delete**.
4. Click **OK** in the **Delete Tenant** warning box.

## ***Creating and Deleting Domains***

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

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 add domains, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Domains** tab to open the Domains window.
4. Click **Create Domain** to open the Create Domain dialog box.
5. Type a name and (optional) description for the domain in the text boxes.
6. Click **Save** to create the new domain.

To delete a domain, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant Name to open its details window.
3. Click a tenant name to open its details window.
4. Click the **Domains** tab to see a list of tenant domains.
5. Select the domains to delete.
6. Click **Delete**, and confirm your choice to delete the selected domains.

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

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

To assign users and groups to roles for a domain:

1. Click the Administration tab to open the Administration Home window.  
Under the Tenants heading, click the required tenant name.
2. Click the **Domains** tab to see a list of the tenant's domains.
3. Click the **Roles** tab.
4. Click the role that you want to add users or groups to.
5. Click **Add Member**.

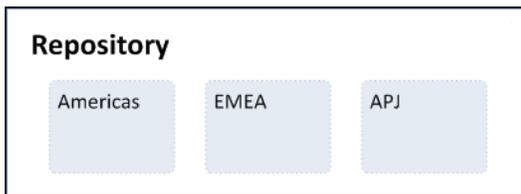
The Add Member dialog opens.

6. Do one of the following:
  - Click the **Users** tab, and select the users to add. Optionally, use the filter to locate a particular user.
  - Click 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.

## Using Domains and Repositories

Domains provide a logical separation of data within a 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.

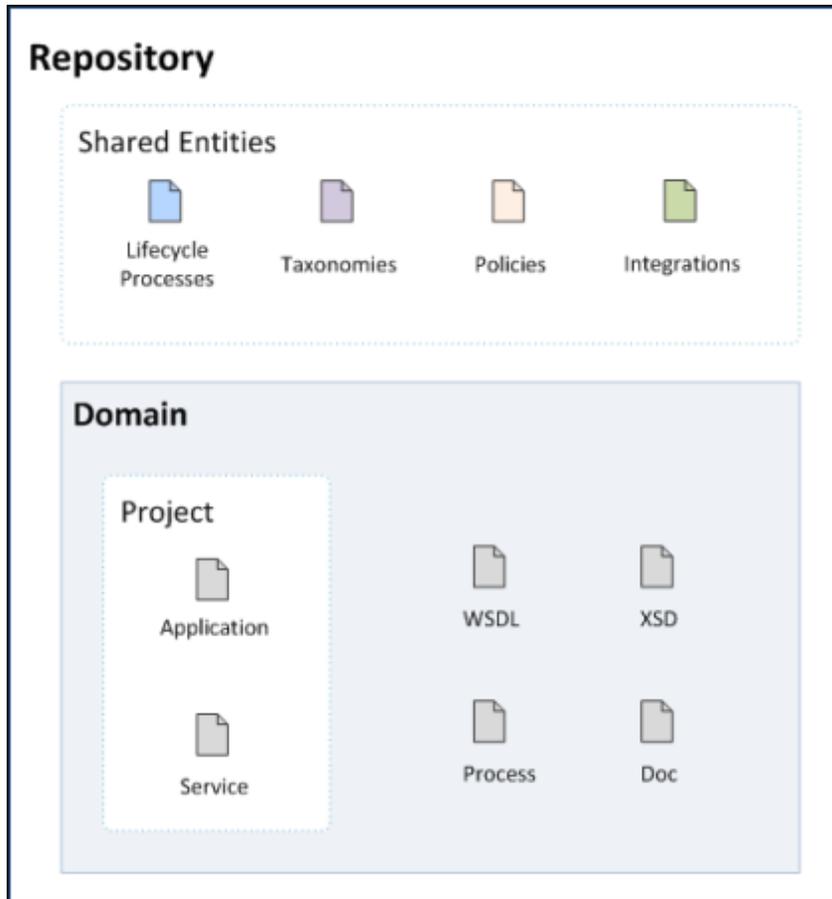
In this release, HP CDA provides support for a single layer of domains within a global top-level repository domain. Following is an example, a domain structure representing organizational regions, Americas, EMEA, and APJ.



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

**Note:** An 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. This data applies across all domains, and each working domain contains the specific data relevant to users of that domain.



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.

Within HP CDA, users perform specific functions based on the roles assigned to them. The user interface restricts users' access to functionality and artifacts based on 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:** When users are assigned to the administrator role in a domain, they are responsible for the following:

Domain management for the domain

Administrative task management within the domain

**Note:** Domain administrators should grant tenant administrator permissions if the multi-tenancy option is enabled.

- **Domain Users:** Users assigned to a specific role within a domain with specific functionality associated with that role. One user can access different domains in different roles.

Each artifact belongs to exactly one domain. The 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 a 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.

## ***Using Domains When HP CDA and HP ALM are Integrated***

You can create and use multiple domains in HP CDA. However, an HP ALM project can only access one (a single) HP CDA domain.

If you create artifacts and deployments in HP CDA that you intend to access from HP ALM, create those artifacts and deployments in a single domain. Configure HP ALM to point to that domain.

If you create artifacts and deployments in HP CDA that you cannot access in HP ALM, check whether your HP ALM arguments use the correct HP CDA domain.

**Caution:** If you intend to use an HP ALM project to access artifacts and deployments created in HP CDA, you must use a single HP CDA domain and configure HP ALM to point to that domain. Otherwise, the integration will fail.

## ***Disabling Default Domain Sign-In***

After you install HP CDA, it contains a default domain where users can sign in if they are not assigned to any working domain.

You can use the Administration UI to change the default domain, but in certain circumstances (for example, when you are dealing with thousands of LDAP users) you might want to disable default domain functionality. When the default domain sign-in is disabled, users cannot sign in until they are assigned to a working 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, an administrator must enable users by creating user artifacts or importing them from the user store. The administrator then assigns the artifacts to a specific domain.

## Setting Default Tenant or Domain 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 permission for artifact types.

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

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenant heading click a tenant name to open its details window.
3. *Optional:* Click the **Domains** tab, and click a domain name to set access rights at the domain level.
4. Click the **Default Access Rights** tab to see the current artifact creation, and read and write access permissions.

**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.

### Managing Artifact Creation Permissions

Perform the following tasks in the Artifact Creation table.

To add artifact creation rules, complete the following steps:

1. Click **Add Rule** to open the Add Artifact Creation Rule window.
2. Choose the required artifact type to add creation rights for from the To Artifact Type menu.
3. In the Granted To table, click **Add Member** to open the Add Member dialog box.
4. Click **Browse Address Book** and select the roles to add creation rights for the selected

artifact type.

*Optional:* Use the dialog Search feature to find a particular role.

5. Click **Select** to add the selected roles to the Granted To table.
6. Click **Save** to add the rule for the selected artifact type to the Artifact Creation table.

To edit artifact creation rules, complete the following steps:

1. Click **Edit** for the rule to open the Edit Artifact Creation Rule window.
2. Use the Granted To table to add and remove roles from the rule.
3. Click **Save** to apply the rule changes.

To remove artifact creation rules, complete the following steps:

1. Select the rules to remove.
2. Click **Remove**, and confirm your decision.

### **Managing Read and Write Permissions**

Perform the following tasks in the Read /Write Access table.

To manage default read and write access, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Domains** tab to see a list of tenant domains.
4. Click a domain name to open its details window.
5. Click the **Default Access Rights** tab to see the current permissions for the domain.
6. In the Read / Write Access table, perform your desired task.

To add read and write access rules:

1. Under the Read / Write Access heading, click **Add Rule** to open the Add Read / Write Access Rule window.
2. Select to add **Read only** or **Read / Write** access.
3. Select the artifact type to add access for from the **To Artifact Type** drop-down list.
4. In the Granted To table, click **Add Member** to open the Add Member dialog box.
5. Select the roles, users, or groups to add read or write access for the selected artifact type. Optionally, use the dialog Search function to find a particular role, user, or group.
6. Click **Select** to add the selected roles, users, and groups to the Granted To table.
7. *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.
8. Click **Save** to add the rule for the selected artifact type to the Read / Write Access table.

To edit read and write access rules:

1. Click **Edit** for the rule you want to edit, to open the Edit Read / Write Access Rule window.
2. Select **Read only** or **Read / Write access**.
3. Use the Granted To table to add and remove roles, users, and groups from the rule.
4. *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.
5. Click **Save** to apply the changes to the rule to the Read / Write Access table.

To remove read and write access rules:

1. Select the rules to remove.
2. 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.

## ***Adding Parameters to Tenants and Domains***

Administrators can add parameters at the tenant and domain levels. The parameters control how users perform some functions in HP CDA.

See ["Enforcing a Base Path "](#) below.

### ***Enforcing a Base Path***

Administrators can set the Enforce Base Path parameter at the tenant or domain level. When set, the parameter limits the targets for Placed File and Placed Directory programming operations. The advantage of using this parameter is that it captures standard paths and locations, so users do not need to manually enter locations. It saves time and eliminates the risk of a typing error.

To set the Enforce Base Path parameter, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. *Optional:* Click the **Domains** tab and click a domain name to set the parameter at the domain level.
4. Click the **Parameters** tab (It might be on the More tab.) to open a window where you can see and manage parameters.
5. Select the **Enforce Base Path** option.

## ***Adding Provisioning Keys to a Centralized Key Store***

Administrators can add authentication private key files to a centralized location in HP CDA. The key files can subsequently be added to platform plugin configurations.

To add keys to a centralized store, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. *Optional:* Click the **Domains** tab and click a domain name to add keys at the domain level.
4. Click the **Keys** tab to see keys. The tab might be on the More tab drop-down menu.
5. Click **Add** to open the **Add key** dialog box, and type entry:

**Name:** for the key file

**Key File:** location

**Description:** optional description

Click **OK** to save entry and create the key store.

## Exporting Tenant or Domain Content

An administrator can export the content of an entire tenant or domain. When you export the top-level domain—you export the entire content of the catalog.

To export content, complete the following steps:

1. In the tenant or domain Overview tab, click **Export** 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
Include Sub-Domains	Select to include all sub-domains.
Data	Select to export all artifacts that are assigned to the domain.
Artifact History	Select to include artifact history.
System Settings	Select which domain settings to export.

4. Click **Export** to create a .zip archive that contains the 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.

The archive is available for import using the Import Repository Archive functionality.

## Importing and Exporting Archives

You can 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.

The following topics provide information about tasks administrators perform to import and export archives.

Topic	Information Included
<a href="#">"Exporting Single Models to a ZIP Archive File " below</a>	How to export a single model archive.
<a href="#">"Exporting Multiple Models to a ZIP Archive File " on the facing page</a>	How to export multiple model archives.
<a href="#">"Importing Model Archives " on page 179</a>	How to import model archives.

### ***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 window that lists the models of the desired type (for example, My Platforms), select the model that you want to export, and then click **Export** above the listing table.
  - In the details screen for the model that you want to export, click **Export** on the right side of the screen.
2. *Optional:* Change the name of the .zip file in the Archive Name box.

**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 **Export** to start the export operation.

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

Your download should start automatically. Click **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 **Save** 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 **Download Contents** near the top of the report.
8. Click **Save** in the File Download dialog box, 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. On the window that lists the models of the desired type (for example, My Platforms), select the models that you want to export, and then click **Export** above the listing.
2. *Optional:* 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, click **Advanced Options** and clear the **Include Dependencies** option.
4. By default, HP CDA does not export the revision history of the models. To export the revision history, click **Advanced Options** and select the **Include History** checkbox.
5. Click **Export** to start the export operation.

A progress bar appears:

6. *Optional:* Click **this direct link** if your download does not start automatically or you want to download the archive later
7. If you want to download the .zip file now, continue with the next step. If you want to navigate away from the window and download the .zip file at a later time, skip to Step 11.

8. Click **this direct link** in the message if a File Download dialog box does not appear.
9. Click **Save** in the File Download dialog box, and then use the Save As dialog box to save the .zip file to a desired location.
10. *Optional:* If you navigate away from the screen where the export operation was initiated and later want to download the .zip file, open the report for the export operation, and click **Download Contents** (near the top of the report).
11. Click **Save** in the File Download dialog box, and then use the Save As dialog box to save the .zip file to a desired location.

**Note:** An 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 **Model Archive** to open the Repository Archive Import window.

**Note:** The Model Archive link appears on the Import menu on the Applications, Platforms and Monitoring tabs.

2. Click to import from **File** or **URL**.
3. Either enter the URL where the file is located or click **Browse** to browse and select the file to upload (import).
4. Select one of the following Conflicting Data options:
  - **Preserve Data in Repository:** The properties defined in the imported models *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 imported models *will overwrite* the properties of existing models in the repository that have the same name.

The following table lists model archive import advanced options.

**Note:** The options only take effect if you have the required permissions.

Advanced Option	Description
-----------------	-------------

Conflicting Relationships	Selects whether to merge or overwrite relationship conflicts. <b>Merge</b> adds any new relationships defined in the archive for existing models to any existing relationships. <b>Overwrite</b> 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 (Click <b>Same as Source</b> .) or to the current domain (Click <b>Current</b> .) If you select to use domains defined in the archive and the domain does not exist, the models are not imported.
Artifact Ownership	Selects whether to use ownership defined in the archive (Click <b>Same as Source</b> .) or to assign ownership of imported content to the current user (Click <b>Current</b> .)
System Settings	Select the checkboxes for the system settings that you want to import if the archive is a domain archive.

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

The archive import process executes as a bulk operation. A progress bar appears at the top of the window to inform you that the operation is in progress. You can click **Stop** to stop the operation or **Notify Me** to 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 might impact the performance of some data queries. Consult your database administrator.

When the operation completes, a report is generated. On the report window, you can click **Undo Changes** to undo 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 roll back those models.

## Managing Lifecycle Processes

Before a user can work with a lifecycle process, the process must be defined and published by an administrator.

Administrators can also copy a lifecycle process, maintain certain values and modify others, rather than defining the process from the beginning.

The following topics provide information about tasks administrators perform to create and manage lifecycle processes.

Topic	Information Included
<a href="#">"Using Lifecycle Best Practices "</a>	List of best practices for managing lifecycle processes in HP CDA.
<a href="#">"Creating Lifecycle Processes " on the facing page</a>	How to define a lifecycle process. This is the first step in artifact lifecycle management.
<a href="#">"Defining Lifecycle Stages " on page 183</a>	How to define lifecycle stages and their order.
<a href="#">"Defining Lifecycle Transitions " on page 185</a>	How to define the transition types that are part of stages.
<a href="#">"Defining Lifecycle Tasks " on page 185</a>	How to define a set of manual tasks to be performed as part of a lifecycle stage.
<a href="#">"Defining Lifecycle Policies "</a>	How to define a set of policies to validate artifact compliance as part of a lifecycle stage.
<a href="#">"Adding, Deleting and Editing Lifecycle Approvers " on page 187</a>	How to define a unique set of approvers for each lifecycle stage in a lifecycle process.
<a href="#">"Defining Lifecycle Automatic Actions " on page 188</a>	How to define, edit and delete automatic actions related to artifacts in different lifecycle stages.
<a href="#">"Defining Lifecycle Permissions " on page 188</a>	How to define permissions for ownership, access rights, approver status and whether artifacts are shared during different lifecycle stages.
<a href="#">"Publishing a Lifecycle Process " on page 189</a>	How to make a lifecycle process available for the governance of artifacts.
<a href="#">"Exporting a Lifecycle Process " on page 190</a>	How to export a lifecycle process.

## ***Using Lifecycle Best Practices***

Following are lifecycle best practices:

- **Govern all artifacts:** HP recommends governing artifacts whenever possible. We are aware that in some situations (for example, for system artifacts) there might 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 artifacts. Instead, an Administrator should create an appropriate lifecycle process that is assigned to the artifact automatically after artifact creation.

You can use Action Set Lifecycle Process / Stage to assign any process to an artifact. Administrators can even assign a specific stage and mark it as approved. This action is intended mainly for administrators who import data to a repository. If the data is suitable, administrators can use this action to assign, for example, a production stage.

- **Separate lifecycle processes:** HP recommends separating lifecycle processes and defining different lifecycle process for different artifact types. You can use policies to connect processes, but it is crucial to not govern all artifacts in one process.

As an example, you can review the default lifecycle processes included in this topic. There is a special lifecycle process for Project artifacts containing only Project artifacts and their documents. Similarly, there are separated lifecycle processes for Applications, Services, and Processes artifacts.

## Creating Lifecycle Processes

The first step in artifact lifecycle management is the creation (definition) of a lifecycle process.

To create a new lifecycle process, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Lifecycle Processes** to open the Lifecycle processes window, and then click **Create Lifecycle Process**.
3. Make entry in the **Create Lifecycle Process** window that opens:
  - a. Type a name and (optional) description for the lifecycle process.
  - b. Define the lifecycle stages for the lifecycle process as follows:

Choose the **Root Artifact Type** from the menu.

*Optional:* To select one or more sub-artifact types, click **Select** to open a window that lists sub-artifact types. Select the required types, and click **OK**.

If you want the new process to be automatically assigned to artifacts, select the **Sub-Artifact Type(s) Assigned Automatically** option, and select 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.

**Applicable To**  
Root Artifact Type: \* Placed Directory  
Sub-Artifact Type(s): Select..  
Assigned Automatically:   
Categorized as: Add Category  
Not Categorized as: Add Category  
In Domains: Edit Domains  
Save Cancel

**Categorized as:** Artifacts must contain these categories to be automatically assigned to the lifecycle process.

**Not Categorized as:** Artifacts must not contain these categories to be automatically assigned to the lifecycle process.

**In Domains:** Select which domains the lifecycle process will be used in.

**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.

4. Click **Save**.

The status of a newly created lifecycle process is set to Draft. The Overview tab opens and you can define lifecycle stages.

## Defining Lifecycle Stages

Lifecycle stages represent important milestones in a 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. To set an initial stage definition for artifacts, click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Lifecycle Processes** to open a window that lists lifecycle processes.

3. Click a lifecycle process name to open a lifecycle process details window.
4. Click **Add Stage** to open the Add Initial Stage dialog box.
5. Choose a stage on the Name menu or type a custom name in the box.
6. *Optional:* Click **Share Artifacts**, and choose when the artifact will be shared. The After Approval option is not available if the transition to the next stage is automatic. (Sharing can be edited on the **Permissions** tab.)
7. Click **OK**.
8. *Optional:* To add more stages, click **Add Stage**, and choose a stage **Name** and **Transition from Stage #** type from the menus.

**Note:** You can add the same stage into a process multiple times. These are called referral stages.

9. The lifecycle process details window provides options to edit, clone, export, deprecate or delete the lifecycle process.
10. Before publishing the lifecycle process, click the **Stages** tab and do at least one of the following:

**Note:** A valid stage definition must contain at least one of the following conditions.

- Set transitions: See ["Defining Lifecycle Transitions "](#) on next page.
  - Set voters for approval: See ["Adding, Deleting and Editing Lifecycle Approvers "](#) on page 187.
  - Define tasks to complete before stage approval: See ["Defining Lifecycle Tasks "](#) on next page.
  - Set policies to comply with before stage approval: See ["Defining Lifecycle Policies "](#) on page 186.
  - Define automatic actions to execute when a lifecycle transition occurs: See ["Defining Lifecycle Automatic Actions "](#) on page 188.
  - Specify whether to automatically promote on stage approval: See ["Defining Lifecycle Permissions "](#) on page 188.
11. Click **Save** to save the stage definition and open the Lifecycle process window.
  12. When the process is complete, you must publish the process to make it active. See ["Publishing a Lifecycle Process "](#) on page 189.

**Note:** Stage layout cannot be changed after the process is published.

## Defining Lifecycle Transitions

When you add a lifecycle stage, you must also add a transition type. Choose from one of the following options:

- **Automatically, When stage\_name is Approved**  
The stage will transition to the next stage as soon as approved by all designated approvers.
- **Manually, After stage\_name is Approved**  
The stage requires approval from all designated approvers before it can be manually advanced to the next stage.
- **Manually, Anytime**  
The stage does not require approval from all designated users before being manually advanced

To edit a transition, hover over the stage or the transition between stages, and click the pencil icon.

**Caution:** If a stage has several next stages, only one of the transitions can be automatic and the rest must be set as **Manually, Anytime**. In case of conflict, you are 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 **Administration** tab to open the Administration Home window.
2. On the Administration menu, click **Lifecycle Processes** to open a window that lists lifecycle processes.
3. Click a lifecycle process name to open its details window.
4. Click the **Stages** tab.
5. Click the stage where you want to add a task.
6. Click **Add Task** to open the Add Task dialog box, and make entry:

- a. Type a name and (optional) description for the task.
- b. Click **Select** next to the Assigned To: label to assign a task to a role, user or group. Click **Browse Addressbook** to reveal lists of roles, users and groups that can have assigned tasks.
- c. Click **Select** next to the Verified by Policy label to create a task that is verified by a policy. A list of existing policies that can be applied appears.

**Note:** Only one policy can be associated with a task at a time. Only policies not used within the same stage can be selected (either in tasks or policies).

- d. Click **OK** to add the new task to the Tasks list.

## ***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 the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Lifecycle Processes** to open a window that lists lifecycle processes.
3. Click a lifecycle name to open its details window.
4. Click the **Stages** tab.
5. Click the stage that requires a policy.
6. Click **Add Policy** to open the Select policy dialog box, and make entry.
  - a. You can use the Search function or select a policy from the list.

**Note:** Only policies that are not used within the *same* stage can be selected (either in tasks or policies).

- b. Click **Select**.

You can use the next window to change your choice of policy, the artifact type to which it is applicable and whether the policy is required to approve the stage.

- c. Click **OK** to apply the policy.

## ***Adding, Deleting and Editing 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 add approvers:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Lifecycle Processes** to open a window that lists lifecycle processes.
3. Click a lifecycle process name to open its details window.
4. Click the **Stages** tab. Select the approval stage and make entry:
  - a. Click **Browse Addressbook**.
  - b. To find principals, select them from the lists or type a name in the Search text box and click **Search**.
  - c. Select the Role, User or Group option, and click **Select** to add your selection as an approver.
  - d. If the selected approver is a Role or Group (not an individual), select the number of votes required for that Role or Group.
  - e. You can select the **Passive Approval** option and type a number of days before automatic approval.
  - f. Click **OK** to add the approver.

To remove an approver:

- On the Stages tab, click **Delete** next to the approver's name.

To edit a Passive Approval option:

- On the Stages tab, click **Edit** next to the approver's name.

**Note:** Approvers can also be viewed, edited or added on the Permissions tab.

## Defining Lifecycle Automatic Actions

To define automatic actions, click **Add Automatic Actions** at the bottom of the stage details window on the Stages tab.

The following actions might be available, depending on the artifact type:

- **Parent Request Approval** - Creates an approval request for a parent artifact. Parent Artifact Type and Parent Stage must be selected. Click **OK** to add automatic action. The approval on the parent's artifact is requested automatically.
- **Enable Consumption** - Enables the creation of new contracts.
- **Disable Consumption** - Prevents the creation of new contracts.
- **Remove Comments** - After the artifact goes into production, all comments are removed.
- **Delete from Registries** - Perform a 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** - Performs an 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** - You can use HP CDA 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 can include Lifecycle Status and an attached text.

The possible recipients of the notification are listed in the following table.

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.

Click **Delete** to delete automatic actions. Click **Edit** to edit Notifications and Parent Request Approval actions.

## 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 and approver status, and whether artifacts are shared.

**Note:** Permission settings are not carried forward to the next stage.

To define a lifecycle permission, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Lifecycle Processes** to open a window that lists lifecycle processes.
3. Click a lifecycle process name to open its details window.
4. Click the **Permissions** tab.
5. Select a stage, and make the following permissions changes as required:
  - Click **Add Read Access** or **Add Write Access** to assign permissions to users, roles or groups. You can also click the corresponding value in the permissions matrix to edit permissions.
  - Click **Set Owner** or the **Ownership** matrix value to assign a stage owner.
  - Select a value from the **Share Artifacts** list to set whether artifacts can be shared.
  - Click **Clear** to remove a stage's permissions.
  - Click **Copy** to open the Copy Stage Permissions dialog box. Select the stages or roles where you want to copy, and click **Copy**.

## ***Publishing a Lifecycle Process***

After you create a lifecycle process, your next step is to make it available for the governance of artifacts by publishing it.

To publish a lifecycle process, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Lifecycle Processes** to open a window that lists lifecycle processes.
3. Select the lifecycle processes to publish, and click **Publish**.

The lifecycle process is now available to be used in the governance of root and sub-artifacts.

**Note:** You can also publish on the Overview or Stages tab of a process.

To deprecate a lifecycle process, complete the following steps:

- On the Lifecycle processes window, select the processes to deprecate, and click **Deprecate**.

**Note:** You can also deprecate from the Overview or Stages tab of a Process.

## ***Exporting a Lifecycle Process***

You can export a lifecycle process on either its Overview or Stages tab.

To export a lifecycle process, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Lifecycle Processes** to open a window that lists lifecycle processes.
3. Click a lifecycle process name to open its details window.
4. Click **Export** on the right to open the Export dialog box:
  - a. Accept or type a name for the exported file.
  - b. Click **Export**.
5. Click to either **Open** or **Save** (and provide a location for) the exported file.

## **Managing Policies and Assertions**

You can 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. When you reference other technical policies, you can collect a set of technical policies together into a larger policy, so you can validate them collectively.
- **Assertion:** An assertion is a validation check for a single piece of data that 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 that is categorized as a business specification document before the service is approved at the candidate lifecycle stage.

**Data Content:** Some artifact types should contain attached content. An assertion can verify that an 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).

Use the Assertion Editor to create and edit assertions. For details, see the *Assertion Editor Guide*.

You can use technical policies in the following ways:

- **Lifecycle Stage Validation:** The stages in a lifecycle process can have a set of policies associated with them that serves 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.

Policies are automatically validated when a user requests stage approval or you can manually validate policies. The collective status of policies gives a Compliance Status for an artifact. The status measures the current percentage of required policies that pass validation. (Administrators assign policies to Lifecycle Stages at the time of their creation.)

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 function of the Reports tab to perform ad-hoc manual validation. You can set up a policy report that validates a selected set of artifacts against a set of selected technical policies.
- **Artifact Form Validation:** Administrators 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.

The following topics provide information about tasks Administrators perform to create and manage policies.

Topic	Information Included
<a href="#">"Creating and Deleting Technical Policies " below</a>	How to create and delete technical policies.
<a href="#">"Default Technical Policies " on page 194</a>	List of prepackaged of technical polices.

## ***Creating and Deleting Technical Policies***

To create a technical policy, complete the following steps:

1. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, choose **Technical Policies** to open a window that lists technical policies.
3. Click **Create Technical Policy** to open the Technical Policy window and define the policy:
  - a. Type basic information, name and (optional) description.
  - b. Click to add **Technical Policies** and **Assertions**.
  - c. Click **Save** to save the technical policy.

To delete a technical policy, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Technical Policies** to open a window that lists technical policies.
3. *Optional:* Use the **Applicable To** column to sort technical policies by their applicable artifact type.
4. Select a technical policy.
5. Click **Delete**.
6. Click **OK** to confirm the deletion.

The technical policy details window provides the standard edit, delete, and export artifact management tasks.

To access a technical policy details window, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Technical Policies** to open a window that lists technical policies.
3. Click a technical policy name to open its details window where you can perform management tasks.

You can collect a set of technical policies together under a single technical policy (a parent). Collection enables you use the parent policy to perform validation against the entire set of policies.

To collect a set of related policies under one policy, complete the following steps:

- Click **Add Technical Policy** to add the first technical policy, and then use the standard table functionality for related artifacts.

To add technical policy assertions, complete the following steps:

1. Click **Add Assertion**, and use the standard table functionality.
2. After you select an assertion to add, you must set its scope within the technical policy. Select 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.

3. *Optional:* To change the severity and parameters for an assertion, click **Edit** for the assertion.

The Create and Edit technical policy windows 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 <b>Add Technical Policy</b> function to manage referenced technical policies.
Assertions	Use the <b>Add Assertion</b> function to manage referenced assertions.

## Default Technical Policies

HP CDA includes a prepackaged set of technical polices.

These technical polices are listed in the following tables organized by the type of validation they perform.

Technical Policy	Definition
Related Artifact Validity	Policies that verify that relevant related artifacts are present for a service, such as documentation
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.

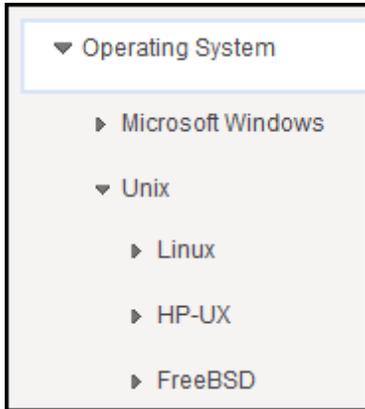
## Adding, Deleting and Modifying Tags

Tags describe and define platform and application requirements and capabilities.

An application model requires specific types of servers, storage, networking, etc. on the platforms upon which it is deployed. A specific platform provides various capabilities, such as CPU, memory, operating system type and version, and other middleware (such as Tomcat or MySQL).

You can add tags to application models and platforms to associate application model requirements with platform capabilities in a uniform manner. For example, when an application with tags is deployed, HP CDA offers only those servers that contain the tagged requirements as available servers. For example, an application needs 512MB RAM, Linux and JBoss 5.0 or newer. This application can be deployed on a platform with 2GB, Red Hat Enterprise 4 and JBoss 7.0.

You can group and describe tags in a hierarchical manner by creating sibling and child tags. For example, you could create a parent tag **Operating System** that includes the following child tags: **Microsoft Windows** and **Unix**. The **Unix** tag might contain the child tags **Linux**, **HP-UX** and **FreeBSD**. (**Linux** and **HP-UX** are sibling tags.) The following graphic shows the tags as they appear on the Tag Management window.



Some tags have parameters. Application designers can use those parameters to define applications, independent of a platform. For example, tag **JBoss** has the **jboss.home** parameter. An application designer can set **JBoss** as an application requirement and use the **jboss.home** parameter in installation scripts. Later, when a platform designer defines a platform, he fills the parameter with concrete value (for example `c:\Program Files\jboss-as\`).

The following topics provide information about working with tags.

Topic	Information Included
<a href="#">"Opening a Tag Management Window " below</a>	How to open a window where you can see and work with tags.
<a href="#">"Importing and Exporting Tag Archives " on page 198</a>	How to import and export tags. Tags are not exported automatically during application or platform export.
<a href="#">"Adding Tags " on page 202</a>	How to add tags that you can use to assign requirements and capabilities when you create software and platforms.
<a href="#">"Editing Tags " on page 202</a>	How to make changes to editable tags. HP CDA default tags are not editable.
<a href="#">"Further Defining Tags " on page 203</a>	How to add parameters, compatible-version sequences and translation information for imported infrastructure templates.
<a href="#">"Removing Tags " on page 207</a>	How to delete tags that are not in use.

## ***Opening a Tag Management Window***

Use tags to specify:

- requirements and capabilities for platforms and platform software
- requirements for application layers

To open a window where you can see and work with tags, complete the following steps.

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Tag Management**.

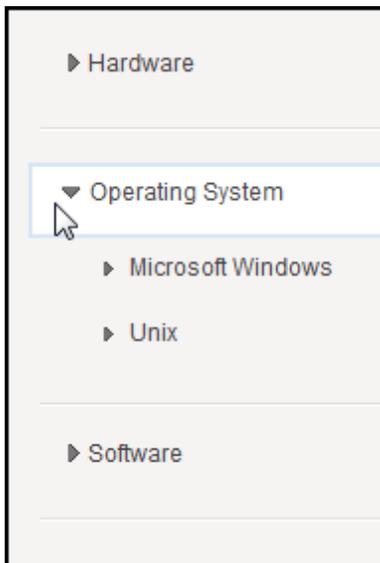
On the left, you see a menu where you can choose actions to perform.

On the right is a window where you can make entry.

HP CDA installs common, default tag menu options, such as Operating System > Microsoft Windows and Operating System > Unix.

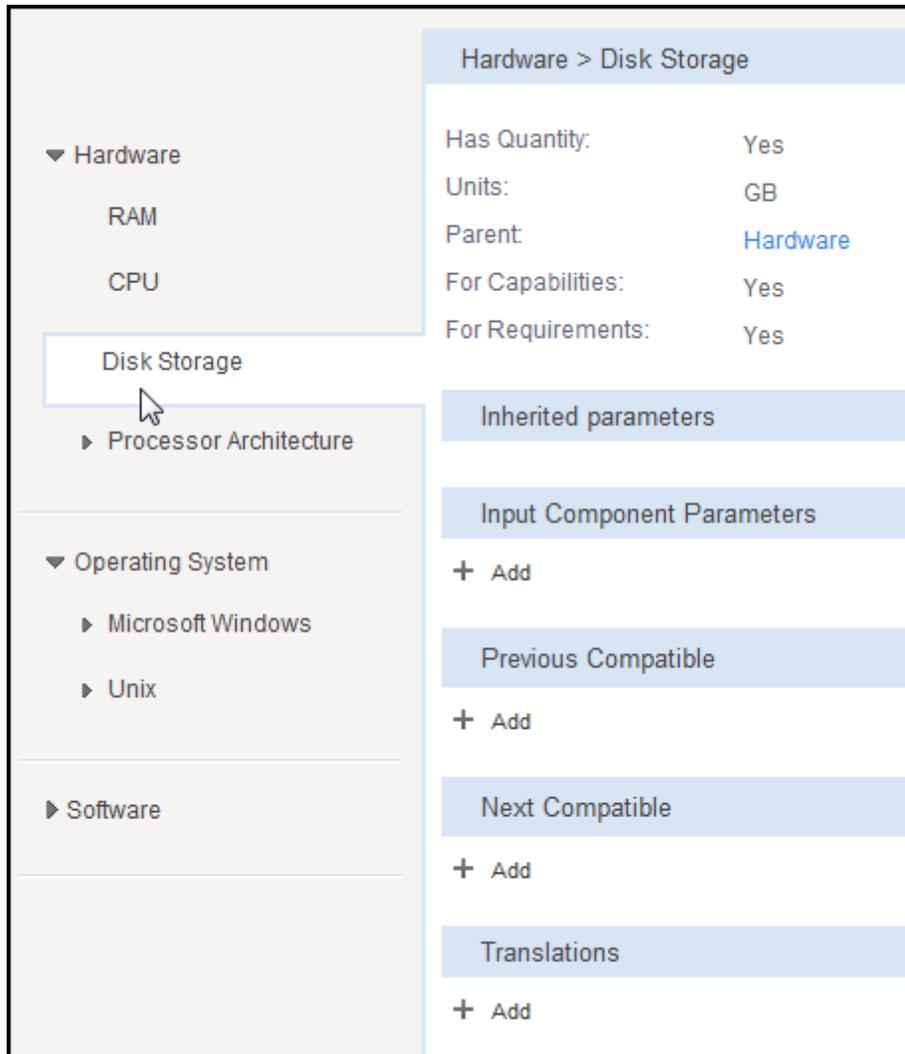
To expand the menu options, complete the following steps:

- Click the caret to the left of the tag menu option.



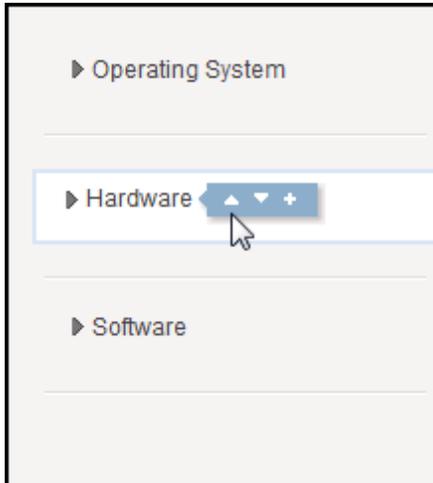
To open a window where you can see and edit tags, complete the following steps:

- Click the name of the tag menu option, as shown in the following graphic.



To perform actions, complete the following steps:

1. Choose an action from the menu to the right of the tag menu option, as shown in the following graphic.
2. You can click:
  - The up carat to move the tag menu option up in the list. This action changes only how the list displays.
  - The down carat to move the tag menu option down in the list. This action changes only how the list displays.
  - Plus (+) to add a tag. See ["Adding Tags " on page 202.](#)



## ***Importing and Exporting Tag Archives***

Tags are not exported automatically during application or platform export. Administrators must separately export and then import tags. Otherwise, you could invalidate objects (in the application or platform), because those objects contain references to tags that no longer exist.

Another use case is that you might create tags in one instance of HP CDA that would be useful in a different instance of HP CDA. In that case, you can export the existing tags and import them into the system where they are needed.

To export tags, complete the following steps:

1. Click the **Administration** tab to open the Administration home window.
2. On the Administration menu, choose **Tag Management**.
3. Click **Export** to open the Export Tags window.
  - a. Accept the default or type in an **Archive Name**.
  - b. Click **Export**. An HP CDA report provides information about the export operation.

To import tags, complete the following steps:

1. Click the **Administration** tab to open the Administration home window.
2. On the Administration menu, choose **Tag Management**.
3. Click **Import** to open the Repository Archive Import window.

- a. Select an **Import from** option.
- b. Type or browse for the import **File** location.
- c. **Select a Conflicting Data** option:

**Preserve Data in Repository:** If the tag already exists in HP CDA, the archive tag is not imported.

**Overwrite Data in Repository:** If the tag already exists in HP CDA, the archive tag is imported and overwrites the existing tag.

- d. Select an Artifact UUIDs option:

**Note:** Every tag has unique identifier (UUID). This option selects whether the identifiers remain the same after import. HP strongly recommends that you select **Keep from Archive**. Otherwise, import can result in duplicate tags.

**Keep From Archive:** Use UUIDs from the archive.

**Generate New:** Generate a new UUID for each tag.

- e. *Optional:* Select **Advance Options**.
- f. Click **Import**. An HP CDA report provides information about the import operation.

## ***Exporting and Importing Tags as XML***

Tags can be exported from the HP CDA repository as an XML file, which can be modified if desired and imported to the repository. The XML that is exported is in an easy to read and understand format to facilitate editing in a standard XML browser or text editor.

The following topics explain how to export and import tags.

<b>Topic</b>	<b>Information Contained</b>
<a href="#">"Exporting Tags as XML " on the facing page</a> <a href="#">"Exporting Tags as XML " on the facing page</a>	How to export tags from the HP CDA repository as XML files.
<a href="#">"Importing Tags as XML" on page 201</a> <a href="#">"Importing Tags as XML" on page 201</a>	How to import a tag XML file into the HP CDA repository.

## Exporting Tags as XML

You can export tags from the HP CDA repository as XML. This feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

To export tags as XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Browse the tags by clicking the Tag Management link in the Administration section of the HP CDA user interface, and copy the exact name of the tag that you want to export.
3. Open the REST client, and issue a GET request with the following characteristics:

- **URL:**

```
https://<cda_
address>:8443-
/cda/remote/rest/v1/administration/tags?treeRoot=<tag_name> (Secure
HTTP)
```

```
http://<cda_
address>:8080-
/cda/remote/rest/v1/administration/tags?treeRoot=<tag_name>
(Standard HTTP)
```

Where <cda\_address> is the IP address or FQDN of the HP CDA instance, and <tag\_name> is the name of the tag that you want to export.

**Note:** To export the XML for all tags, issue a GET request to the above URL without the ?treeRoot=<tag\_name> parameter.

- **Content Type:** application/xml

If the GET request succeeds (a status code of 200 OK is returned), the XML for the tag will be returned.

4. Copy the tag XML to an editor for modification, or save it to a text file for later editing and import.

## Importing Tags as XML

You can import tag XML to the HP CDA repository. The XML that you import has been exported using the procedure ["Exporting Tags as XML" on previous page](#) and typically has been stored to a text file.

If you add new tags in the XML before importing it, you must ensure that the tag is included within the proper tag hierarchy. For example, if a JBoss application server tag named "JBoss New Version" is added as shown in the following, the `<parentPath>` must reflect the hierarchy in which the new tag will be placed, in this case Software/Application Server/JBoss.

```
<tags>
  <tag name="JBoss New Version" hasQuantity="false" hasArchitecture="false"
forRequirements="true" forCapabilities="true">
  <parentPath>Software/Application Server/JBoss</parentPath>
  <previousCompatible>Software/Application Server/JBoss/JBoss 5.0</previousCompatible>
</tag>
</tags>
```

The tag import feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

To import tag XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Open the REST client and issue a POST request with the following characteristics:

- **URL:**

```
https://<cda_
address>:8443-
/cda/remote/rest/v1/administration/tags?treeRoot=<tag_name> (Secure
HTTP)
```

```
http://<cda_
address>:8080-
/cda/remote/rest/v1/administration/tags?treeRoot=<tag_name>
(Standard HTTP)
```

Where `<cda_address>` is the IP address or FQDN of the HP CDA instance, and `<tag_name>` is the name of the tag that you want to import.

- **Header:** application/xml
- **Body:** The tag XML to be imported to the repository.

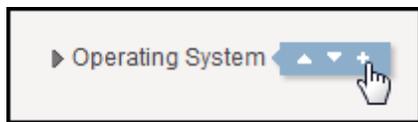
If the POST request succeeds (a status code of 200 OK is returned), the XML for the tag has been imported into the HP CDA repository.

## Adding Tags

After you create tags, they are available for selection on the Requirements and Capabilities window sections. You use those sections to create and define software (Applications tab) and server platforms (Platforms tab).

To add a tag, complete the following steps:

1. Choose the tag menu option and click Plus (+) to the right of it as shown in the following graphic.



2. Make entry in the window that opens on the right.
  - a. Type a **Name**.
  - b. Select **Has Quantity**, if the tag has a requirements or capability quantity, and type the **Units** in the text box.

**Note:** You can specify whether a tag is used to define capabilities or whether it defines requirements (or both). For example: You can use the “Microsoft Windows” tag as a requirement. You would not use it as a capability, because a capability should be a specific version of an installed system (such as “Windows 7”). But a designer can specify that his application runs on all versions of Microsoft Windows.

- c. Select **For Capabilities**, to define the tag as a capability.
- d. Select **For Requirements**, to define the tag as a requirement.
- e. *Optional:* Add further refinement (input component parameters, previous compatible tags, and so on), as required. See ["Further Defining Tags " on next page.](#)
- f. Click **Save**.

## Editing Tags

You cannot edit (or delete) the HP CDA default tags (such as Software, Microsoft Windows and RAM). You can edit (delete) all other tags.

To edit a tag, complete the following steps:

1. Click the **Administration** tab to open the Administration home window.
2. On the Administration menu, choose **Tag Management**.
3. Click the tag name to open a details window.
  - a. Make edits.
  - b. Click **Save**.

## ***Further Defining Tags***

### **Inherited parameters**

If you define parameters on a tag, all of the children and next compatible tags inherit the parameters. This read-only list shows those inherited parameters.

### **Input Component Parameters**

Application designers can use these parameters for application definition independent of a platform.

To learn more, [click here](#).

You can add tags to application models and platforms to associate application model requirements with platform capabilities in a uniform manner. For example, when an application with tags is deployed, HP CDA offers only those servers that contain the tagged requirements as available servers. For example, an application needs 512MB RAM, Linux and JBoss 5.0 or newer. This application can be deployed on a platform with 2GB, Red Hat Enterprise 4 and JBoss 7.0.

### **Previous Compatible Tags and Next Compatible**

Your entry in Previous Compatible Tags or Next Compatible defines a sequence of compatible product versions. Whether you define the sequence from newest to oldest (previous-compatible) or oldest to newest (next-compatible), the result is the same.

HP CDA needs compatible version information for proper matching between applications and platforms. For example, if an application requires Oracle 8i or newer—the application can be deployed to a platform with Oracle 11g. HP CDA detects the match by the tag compatible sequence **Oracle8i -> Oracle9i -> Oracle 10g -> Oracle 11g**.

The Select Artifact dialog box shows a list of all existing tags, which you can use to create version sequences.

HP strongly recommends that you define a compatible tag sequence for all tags that have several versions.

### **Translations**

HP CDA can import infrastructure templates from different systems (HP Cloudsystem Matrix, HP Cloud Connector). Immediately after import, the templates have some capabilities. (Users do not need set those capabilities.)

HP CDA needs information to translate between infrastructure templates and tags. For example, HP Cloudsystem Matrix uses **WINDOWS** as a software capability that translates to the **Microsoft Windows** HP CDA tag.

- **Translation:** name of the software in the infrastructure template.
- **Provider:** the infrastructure template provider.

### Interface

HP CDA allows you to create custom workflows whereby a workflow is delegated or shared amongst layers, or within platform software. You can delegate platform software workflows using the interface definition(s) specified here. See ["Adding an Interface Operation to a Tag "](#) below for more information.

## ***Adding an Interface Operation to a Tag***

Interface operations (referred to as *Interfaces* in the HP CDA user interface), when defined on a Tag and added to platform software as a Capability, creates a workflow in that platform software that implements the interface. The interface and thus the workflow can then be accessed from multiple applications and application layers. Interface operations are used when delegating control from an application layer workflow to the workflow in platform software when the application is being deployed, started, stopped, or redeployed.

For example, you can define an Interface named "deployApplicationToJbossServer" on a Tag named "Jboss 5.0" and add that tag as a Capability to platform software named "Jboss 5.0 Software." This automatically adds a workflow named "deployApplicationToJbossServer" to the Jboss 5.0 Software to which you can add the programming operations necessary to deploy an application to a JBoss server. If the "Jboss 5.0" tag is then added to an application layer as a Requirement, you can add an Invoke Workflow programming operation to that layer which references the Interface operation. If the application layer is mapped to a platform tier which has the JBoss 5.0 software added to it, the Invoke Workflow operation will delegate control to the "deployApplicationToJbossServer" workflow when the application operation executes, in this case deploying the application to the JBoss server.

To add an Interface operation to a tag, complete the following steps:

1. Click the **Administration** tab to open the Administration home window.
2. On the Administration menu, choose **Tag Management**.
3. Navigate to and click the name of the tag on which you want to define the Interface operation, or add the tag within the pertinent tag hierarchy if it does not yet exist.

4. In the Interface section of the details for the tag, click **+ Add** to open the Operation window.
5. Enter a name for the Interface operation and an optional description. The name you enter here will be the name of the workflow that is automatically added to any platform software to which the tag is added as a Capability.
6. Click **+ Add** under Input Component Parameters to open the Add dialog box and add any input component parameters that will be necessary to pass parameter values during workflow delegation. The fields in the Add dialog box are as follows:

Field	Description
Label	This label identifies the parameter—it is not the actual name of the parameter used in references and substitution variables. Your entry here automatically populates Name.
Name	<p>This field becomes the character-for-character string that is used in references and substitution variables. This field is automatically populated by Label entry, but you can change it before you click Save. After you save, you cannot change Name.</p> <div style="background-color: #f0f0f0; padding: 5px; border: 1px solid #ccc;"> <p><b>Note:</b> You can use a workaround to change the name by deleting your original parameter and recreating it with a new name. If you rename a parameter in this way, you must change the parameter name in places where it is referenced, such as in substitution variables.</p> </div> <p>Parameters within the same parameter group cannot use the same name. However, parameter names are case-sensitive. You can create a parameter named <code>my.parameter</code> and another named <code>my.Parameter</code> in the same group.</p> <p>The duplicate name restriction does not apply to parameters that are created in different parameter groups.</p>
Description	An optional description for the parameter.
Encrypted	To mask the parameter value where it appears or is entered on interface windows.
Read Only	If you want the parameter value to be un-editable on interface windows. (The Add and Edit parameter dialog box will allow edits.)
Private Scope	To limit the parameter scope to within the area where it was created. For example, a parameter with a private scope might be listed on the Parameters tab of the Application Model window, but not on the Deploy Application wizard.

Parameter Type	<p>Following are the available parameter types:</p> <ul style="list-style-type: none"><li>• <b>String:</b><ul style="list-style-type: none"><li>- <i>Minimal Length</i>: The string's minimum character length.</li><li>- <i>Maximum Length</i>: The string's maximum character length.</li><li>- <i>Allowed text</i>: Enter a text string here to establish a regular expression for the string. User input must match the regular expression.</li></ul><p>HP CDA supports a common subset of the Java and JavaScript regular expression syntax.</p><p>For information on Java regular expression syntax, search <a href="http://docs.oracle.com">http://docs.oracle.com</a>.</p><p>For information on JavaScript regular expressions, search <a href="http://www.ecma-international.org/publications">http://www.ecma-international.org/publications</a> .</p><p>If you leave this text box empty, it is the same as establishing the default “.” regular expression. In that case, users are allowed to enter any string.</p></li><li>• <b>Number::</b><ul style="list-style-type: none"><li>- <i>Minimal Value</i>: The number's minimal value.</li><li>- <i>Maximum Value</i>: The number's maximum value.</li></ul></li><li>• <b>List:</b> Click <b>Add Value</b> to add additional values that appear in or populate the List.  Choose the <b>Enable MultiSelect</b> option to enter more than one parameter value, separated by delimiters.  Accept a comma (,) as the <b>Delimiter Character</b> or type a different character in the text box.</li><li>• <b>Email:</b> No additional input required.</li><li>• <b>Boolean:</b> Use entry here to define the values for <i>True</i> and <i>False</i>.</li><li>• <b>Key:</b> No additional input required.</li><li>• <b>Directory:</b> Choose Directory to enforce that files are copied to predefined directories (using the <code>BASE_PATH</code> parameter). If an administrator has defined a base path for the Directory, users cannot edit the base path.</li></ul>
----------------	--

Default Value	The default value of the parameter, which can be one of the following: <ul style="list-style-type: none"><li>• <b>Blank:</b> When no default value is desired.</li><li>• <b>A textual value:</b> Type a textual value in the text box.</li></ul>
---------------	--

7. Click **Save** in the Operation window to save the Interface operation.
8. Click **Save** that is under the details for the tag to save the changes to the tag.

## Removing Tags

You should not delete a tag that is in use. After you delete a tag, HP CDA performs a validation. It marks any application or software *invalid*, if it uses the deleted tag.

To delete a tag, complete the following steps:

1. Click the **Administration** tab to open the Administration home window.
2. On the Administration menu, choose **Tag Management**.
3. Click the tag name to open a details window.
4. Click **Delete**.
5. Click **OK** in the warning window to confirm the delete.

## Adding and Running Administration Tasks

HP CDA includes a number of default administration tasks to help administrators manage their deployment and the data it contains. In addition, administrators can configure additional tasks to perform change management or custom tasks included in an extension.

To display and manage administration tasks, complete the following steps:

1. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, click **Tasks** to open the Tasks window.

The initial view of the administrator Tasks window 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 on 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 on 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 on the Build Cleanup tab for each domain.
- **Data File Cleaner Task:** Runs a data file cleanup job that removes file artifacts.
- **Delegation Updater Task:** Runs the Delegation Update job.
- **Send Delayed Notifications Task:** Sends all expired delayed notifications.

Administrators can execute tasks immediately or schedule them to run on a periodic basis. In addition to the default tasks, administrators can add change management tasks and custom tasks.

The following topics provide information about tasks Administrators perform to create and manage tasks.

Topic	Information Included
<a href="#">"Adding Custom Tasks " below</a>	How to add and schedule custom tasks.
<a href="#">" Adding Artifact Sync Tasks " on next page</a>	How to add tasks to synchronize artifacts in the repository with those imported from an external source.
<a href="#">"Running Tasks " on page 210</a>	How to manually execute a task.

## ***Adding Custom Tasks***

HP CDA includes an extension that contains custom task implementations. You can add and schedule custom tasks on the Tasks window.

To create custom tasks, complete the following steps:

1. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, choose **Tasks** to open the Tasks window.
3. Click **Add Custom Task** on the right to open the Add Custom Task window. Define the task:
  - a. Select a **Task Implementation**.
  - b. Type a name and (optional) description.
  - c. *Optional:* Click **Set Schedule** to set a schedule for running the task.
  - d. *Optional:* Type **Task Parameters** in the text box.

**Note:** Task Parameters input requirements vary, depending on how the task is composed.

- e. *Optional:* Select a **Process Artifacts Defined** option.
- f. Click **Save** to save the custom task.

The Task window opens and you see the task in the list. You can complete the following steps:

1. *Optional:* Click **Edit Schedule** to change the schedule for running the task, or click **Run** to run it immediately.
2. *Optional:* Click **Edit** to edit the task.

## ***Adding Artifact Sync Tasks***

Administrators can use sync tasks to synchronize artifacts in the repository with those imported from an external source.

To create sync tasks, complete the following steps:

1. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, choose **Tasks** to open the Tasks window.
3. Click **Add Artifact Sync Task** on the right to open the Add Artifact Sync window. Define the sync task:
  - a. Type a name and (optional) description for the sync task.

**Note:** You can use either the Search function or select artifacts from the list.

- b. Click **Save** to save the sync task.

The Task window opens and you see the sync task in the list. You can complete the following steps:

1. *Optional:* Click **Edit Schedule** to set a schedule for the running the new sync task, or click **Run** to run the task immediately.
2. *Optional:* Click **Edit** to edit the sync task.

## Running Tasks

You might want to run a task immediately. If you have sufficient permissions, you see a **Run** option on the Tasks window and on the task details window.

To manually run tasks, complete the following steps:

- Click **Run** on either the Task window or the task details window.

### Scheduling Tasks

You can run HP CDA 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. On the Task window or task details window, click **Edit Schedule** to open the Edit Schedule dialog box.
2. Define the new schedule:
  - a. Select a start date and start time for the task.
  - b. On the **Recurrence** menu, choose how frequently the task will run.
  - c. Select the task schedule **End** from the following options:

**Indefinitely**

**On:** Set a finish date and time

**times:** Type a numeral for how many times the task will run.

- d. *Optional:* Use **Advanced Options** to select whether the schedule runs during downtime or waits until HP CDA is running.
- e. Click **OK** to set the schedule for the task.

## Managing Users, Groups and Roles

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 assigned user or an administrator can manage. Users can create additional Contact artifacts to represent external contacts who do not use the product—to associate external contacts with artifacts in the catalog.

If required, 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.

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

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

The following topics provide information about tasks administrators perform to manage people (both users and administrators), groups and roles.

Topic	Information Included
<a href="#">"Managing Users " on the facing page</a>	How to enable and disable users, assign them to groups and roles, set their default domain, and change the default ownership of artifacts they create.
<a href="#">"Managing Groups " on page 220</a>	How to create, delete and retire groups. How to set a default domain for a group and add the group to additional domains.
<a href="#">"Managing Roles " on page 224</a>	How to create and delete roles, and change the Sharing Principal. Information includes listings of user interface and lifecycle roles.

## Managing Users

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 on 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.

The following topics provide information about tasks administrators perform to manage people—both users and administrators.

Topic	Information Included
<a href="#">"Creating and Importing Users and Administrators " on next page</a>	How to create and import users and administrators under a tenant and domain.
<a href="#">"Editing Users and Administrators " on page 214</a>	How to edit users and administrators.
<a href="#">"Assigning Users to Roles " on page 214</a>	How to assign users to additional roles within their domains.
<a href="#">"Assigning Users to Groups " on page 215</a>	How to assign users to additional groups within their domains.
<a href="#">"Setting Default Domains for Users " on page 216</a>	How to set the default domain that members of a group sign in to.

<a href="#">"Setting a User's Default Domain Role " on page 217</a>	How to set the default role that HP CDA assigns to any user who signs into a domain, but who does not have a role in that domain.
<a href="#">"Setting New Artifact Ownership " on page 217</a>	How to set the default ownership of artifacts created by a user to a specified role or group.
<a href="#">"Enabling and Disabling and Users " on page 218</a>	How to grant or block user access to HP CDA.
<a href="#">"Retiring Users in Multi-Tenant Environments " on page 219</a>	How to retire users when HP CDA is installed as a multi-tenant environment.
<a href="#">"Retiring Users in Environments without Tenants " on page 220</a>	How to retire users when HP CDA is installed in an environment without a tenant structure.

## ***Creating and Importing Users and Administrators***

Perform the following functions on the Members tab. Complete the following steps to open the tab:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users and groups.

To create a user or administrator, complete the following steps:

1. Click **Create User / Create Administrator** on the right to open the Create User / Create Administrator window.

The window lists rules for name and password entry. If a rule is violated, the rule displays in red.

2. Type:

**Login name:** following the rules and prompts that you see.

**Password:** following the rules and prompts that you see.

**Re-type Password:** to confirm your password entry.

3. Click **Create**.

The User window opens.

The new user or administrator appears on the Members tab.

To import users, complete the following steps:

1. Click **Import Users** to open the Import Users dialog box.
2. Choose an **Import Mode** from the menu:

**Replace**

**Add or Update**

3. Browse to select a CSV file to import.
4. Click **OK**.

## ***Editing Users and Administrators***

To edit users and administrators, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users and groups.
4. Click a user name to open a user details window.
5. Click **Edit** on the right to open a window where you can add, edit or delete information about the person.
6. Type or select changes to the information.
7. Click **Save**.

## ***Assigning Users to Roles***

**Note:** This functionality is only available if the person is already assigned roles in a domain.

To assign people (either users or administrators) to additional roles, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users and groups.
4. Click a user name to open a user details window.
5. Click the **Groups and Roles** tab to see the role details for that person.
6. In the Roles by Domain section, click **Change** for the required domain.
7. Make entry in the **Change Role Membership** dialog box that opens:
  - a. Select additional roles to assign to the person in this domain.
  - b. Clear role selections that you want to remove.
  - c. Click **Set** to apply the changes.

**Note:** You can also manage role assignments at the roles level. See ["Adding and Removing Roles Members "](#) on page 230.

## ***Assigning Users to Groups***

An administrator can assign a user to be a member of multiple groups.

To assign users to groups, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users and groups.
4. Click a user name to open a user details window.
5. Click the **Groups and Roles** tab to see the group details for the user.
6. Use the Groups table to add or remove a user's group assignments.

To add a user to groups, complete the following steps:

1. Click **Add to Group** to open the Add Groups to User dialog box.
2. Select **Groups** options for group assignments you want to make.

3. Click **Select** to add the user to the groups.

To remove a user from groups, complete the following steps:

1. Select the **Groups** option for groups to remove.
2. Click **Remove** and confirm your decision.

**Note:** You can also manage group assignments at the group level. See ["Adding and Removing Group Members"](#) on page 221.

## ***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, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users and groups.
4. Click a user name to open a user details window.
5. Click the **Overview** tab, if it is not already selected.
6. Under the Access Security heading, click Default Domain **Change**.
7. In the Set Default Domain dialog box that opens, make entry:
  - a. Select a default domain.
  - b. Click **Select** to set the domain.

When 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 a User's Default Domain Role***

HP CDA provides a default role that it assigns to any user who signs in to 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. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Domains** tab to open the Domains window, which lists domains.
4. Click the domain name to open its details window.
5. Click the **Overview** tab, if it is not already selected.
6. Under the Default Role heading, click **(Change)** to open the Choose Role dialog box and make entry:
  - a. Select the role from the list. Optionally, use the filter to find a particular role.
  - b. Click **Select** to set the selected role as the default role for the 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 and write permission for the new artifact.

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

To set new artifact ownership for a single user, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users and groups.
4. Click a user name to open the user details window.
5. Click the **Overview** tab, if it is not already selected.
6. Under the Access Security heading, click New Artifacts Ownership: **Change**.
7. Make entry in the Select New Artifacts Ownership dialog box:

- a. Type a role or group term in the text box and use the **Search** function, or click **Browse Address Book** and select from the list of roles or groups.

**Note:** The user must be a member of the selected group or assigned to the selected role in at least one domain.

- b. Click **Select** to apply the new artifact ownership assignment to the user.

To set new artifact ownership for multiple users:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, click **Users** to open the Users browse page.
3. Select the users to whom you want to apply new artifact ownership.
4. Expand Manage Members, and select **Set New Artifacts Ownership** to open the Set New Artifacts Ownership dialog box.
5. Click **Change** to open the dialog box.
6. Type a role or group term in the text box and use the Search feature, 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.

## ***Enabling and Disabling and Users***

Administrators can block access to HP CDA by disabling users.

To disable a user, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users and groups.
4. Click a user name to open a user details window.

5. Click **Disable** on the right.
6. Click **OK** to confirm the action.

To enable a user, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users and groups.
4. Click a user name to open a user details window.
5. Click the **Overview** tab, if it is not already selected.
6. Click **Enable** on the right.
7. Click **OK** to confirm the action.

## ***Retiring Users in Multi-Tenant Environments***

The following steps explain how to retire a user when you have installed HP CDA as a multi-tenant environment.

You cannot retire an active user. The **Retire** option does not appear until you disable the user. You must first disable and then retire the user.

To retire a user, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Members** tab to see tenant users.
4. Click a user name to open a details window.
5. On the Overview tab, click **Disable**. The user status changes to Disabled and the **Retire** option appears.
6. Click **Retire** to open the Retire dialog box.
7. Click **Change** to open a dialog where you can transfer the user's artifact ownership to a new owner. Select an owner and click **Select**.
8. In the Retire dialog, click **Retire**. The user status changes to Retired.

To learn how to retire users when HP CDA is installed in an environment without a tenant structure, see ["Retiring Users in Environments without Tenants " below](#).

## ***Retiring Users in Environments without Tenants***

The following steps explain how to retire a user when you have *not* installed HP CDA as a multi-tenant environment.

You cannot retire an active user. If you attempt to do so, you receive an error message, but the message does not provide detail. You must first disable a user, and then retire the user.

To retire a user, complete the following steps:

1. Click the **Administration** tab to open the Administration home window.
2. On the Administration menu, choose **Users** to open the Users window.
3. Select a user and click **Disable**. On the Disable dialog, click **OK**. The user status changes to Disabled.
4. Select the disabled user and click **Retire** to open the Retire dialog.
5. Click **Change** to open a dialog where you can transfer the user's artifact ownership to a new owner. Select an owner and click **Select**.
6. In the Retire dialog, click **Retire**. The user status changes to Retired.

To learn how to retire users when HP CDA is installed in a multi-tenant environment, see ["Retiring Users in Multi-Tenant Environments " on previous page](#)

## **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.

The following topics provide information about tasks Administrators perform to manage groups.

<b>Topic</b>	<b>Information Included</b>
<a href="#">"Creating Groups " on next page</a>	How to create a group.

<a href="#">"Adding and Removing Group Members " below</a>	How to add and remove members from a group.
<a href="#">"Assigning Groups to Additional Domain Roles " on the facing page</a>	How to assign a group to have member roles within a specific domain.
<a href="#">"Setting Default Domains for Groups " on the facing page</a>	How to set the default domain that members of groups sign in to.
<a href="#">"Retiring and Deleting Groups " on page 223</a>	How to retire and then delete groups.

## Creating Groups

To create groups, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Groups** to open the Groups window.
3. Click **Create Group** to open the Create Group window.
4. Type a **Name** and (optional) **Description**.
5. *Optional:* Under the Members heading, add group members as described in ["Adding and Removing Group Members " below](#).
6. Click **Save** to save the group.

## Adding and Removing Group Members

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. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, choose **Groups** to open the Groups window.
3. Click a group name to open its details window.
4. Click **Edit** on the right to open an edit window.
5. Use the Members table to add or remove members.

To add users to the group, complete the following steps:

1. Click **Add Member** to open the Add user to Group dialog box.
2. Select the user to add to the group.
3. Click **Select** to add the selected user to the group.
4. Click **Save** to apply your group membership changes.

To remove members from the group, complete the following steps:

1. Select the users to remove from the group.
2. Click **Remove** and confirm your decision.

**Note:** Administrators can also manage the membership of groups from at the user level. See ["Assigning Users to Groups "](#) on page 215.

## ***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. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, choose **Groups** to open the Groups window.
3. Click the group name to open its detail page.
4. Under the Roles by Domain heading, 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. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, choose **Groups** to open the Groups window.
3. Click the group name 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.

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 the groups.

To retire multiple groups, complete the following steps:

1. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, choose **Groups** to open the Groups window.
3. Select the groups to retire.
4. Click **Retire** and confirm your decision.

To retire a single group, complete the following steps:

1. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, click **Groups** to open the Groups window.
3. Click the group **Name** to open its details window.
4. Click **Retire Group** to open the Retire Group dialog box.
5. Click **(Change)** to select a new owner for any artifacts owned by members of the group.

**Note:** You must retire a group before you can delete it.

To delete multiple groups, complete the following steps:

1. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, click **Groups** to open the Groups window.
3. Select the groups to delete.
4. Click **Delete** and confirm your decision.

To delete a single group, complete the following steps:

1. Click the Administration tab to open the Administration Home window.
2. On the Administration menu, click **Groups** to open the Groups window.
3. Click a group **Name** to open its details window.
4. Click **Delete Group**.

## Managing Roles

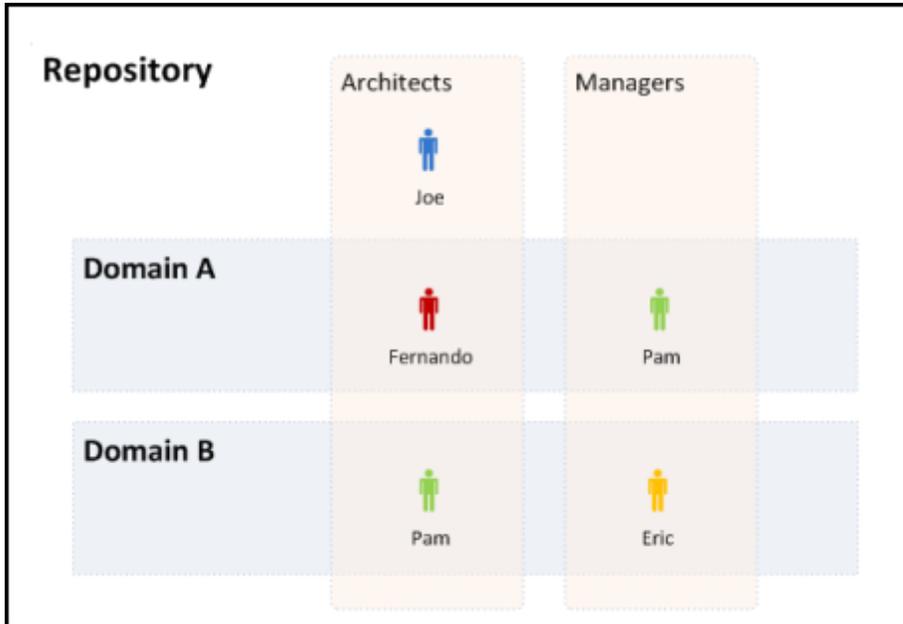
Administrators use HP CDA to assign users to specific roles related to the users' job functions. These roles restrict user's access to artifacts, and limit their functionality to only what is appropriate to their role.

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.

HP CDA offers functionality across the entire service development lifecycle. In most organizations, development lifecycle functions are performed by many individuals and teams—each with specific permissions. Administrators use roles to define and assign permissions. Permissions focus each user or group on specific functionality and tasks, and limit access to only the artifacts appropriate to the 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.



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.

The following topics provide information about tasks administrators perform to manage roles.

Topic	Information Included
<a href="#">"Using User Interface Roles " on the facing page</a>	List of user interface roles.
<a href="#">"Using Lifecycle Roles " below</a>	List of lifecycle roles.

<a href="#">"Creating and Deleting Roles " on page 228</a>	How to create and delete roles that are in addition to the default roles.
<a href="#">"Changing the Sharing Principal " on page 229</a>	How to add and remove members from the Sharing Principal role.
<a href="#">"Adding and Removing Group Members " on page 221</a>	How to add and remove members from a group.

## Using 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 applications 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 application models.
- **Monitoring Administrator:** (default role) Monitoring administrators have access to the Monitoring tab, so they can manage application and infrastructure monitoring.
- **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 use the Administration tab to manage users, groups, roles, domains and other system artifacts.

## Using 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.

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

## ***Creating and Deleting Roles***

Administrators have responsibility for managing roles in HP CDA.

To view and manage roles, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Roles** to open the Roles window.
3. Use the Roles window to work with roles.

**Note:** You cannot delete default roles.

To delete roles, complete the following steps:

1. Select the roles.
2. Click **Delete** to open the Select Entity to Transfer Ownership to dialog box.
3. Use the Search function or browse to make a selection, and click **Select**.
4. Read the warning message in the Delete dialog box that opens.
5. Click **Delete**.

To edit a role, complete the following steps:

1. Click the role name to open its details window.
2. Click **Edit** to open an edit window.
3. Make the required changes.

**Note:** For system roles, you can only change UI Access details.

4. Click **Save**.

To create a role, complete the following steps:

1. Click **Create Role** to open the Create Role window.
2. Type a name and (optional) description.
3. Click **Save**.
4. *Optional:* Use the role details page that opens to edit or delete the role.

The role is available for user and group assignment, for use in lifecycle processes, and for default access rights.

The Create and Edit role windows contain the following text boxes.

Field	Definition
Name	Role name.
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	Enable users in the new role to access the UI.

## ***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. Click the **Administration** tab to open the Administration Home window.
2. Under the Tenants heading, click a tenant name to open its details window.
3. Click the **Roles** tab to see roles and their members.
4. Click **Sharing Principal** to see its membership.

5. Use **Remove** and **Add Member** to edit the membership of the role.

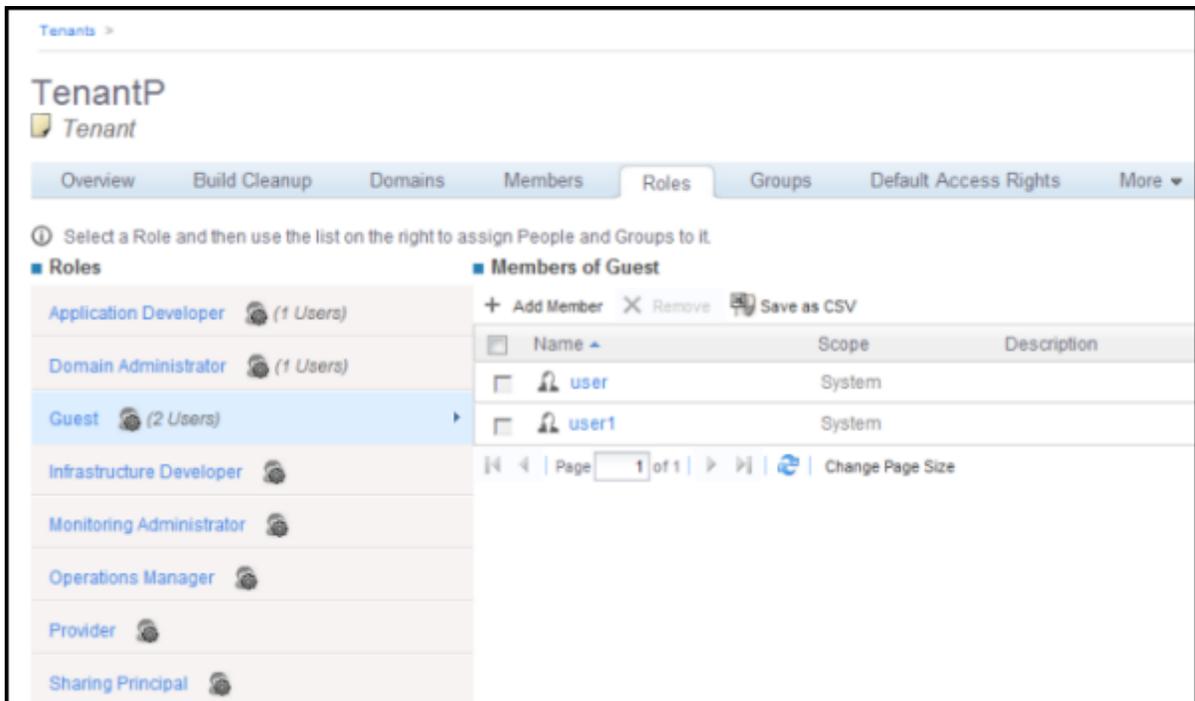
## Adding and Removing Roles Members

Administrators can view all a tenant's roles and members, and manage membership in one place.

To view the roles and members, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. Under the **Tenants** heading, click a tenant name to open its details window.
3. Click the **Roles** tab to see roles and their members.

Click to select a role and work with its members. In the graphic, the Guest role is selected.



To add a member or group to a role, complete the following steps:

1. Click a role to open the Members of dialog box.
2. Click **Add Member** to open the Add Member dialog box.
3. To search for contacts, enter part of a contact name and click **Search**.

4. To browse for the tenant's users and groups, click **Browse Address Book**. Select users and or groups to add to the role.
5. Click **Select**.

The Members of dialog box displays the new role members.

To remove a member or group from a role, complete the following steps:

1. Click a role to open the Members of dialog box.
2. Select one or more users or groups.
3. Click **Remove**. The Members of dialog box displays the updated role listing.

## Managing 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 where the artifact belongs.
- **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 roles that have permission 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. An administrator can assign rights and permissions to particular roles for each stage of a lifecycle process. Within a domain, the 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, an administrator can define which roles can read or write particular artifact types within a domain. Within the domain, the users in roles that have permission 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 permission to artifacts for users, groups, or roles individually or in bulk.

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

## Integrating External Services

You can use plugins to load information about external services. A plugin is a way to communicate with a remote system.

Plugins provide a robust way to manage your environment by enabling:

- Plugin updates while the HP CDA system is running—without requiring a server restart.
- Various upgrade models based on the exposed risks in the environment.
- Immediate, complete replacement of a plugin, so you can hot fix critical defects.
- Plugin testing in a limited environment (because of controlled integration).
- Phased rollouts (by repointing various models to the new plugin to limit risk).
- Immediate replacement of the plugin (by making plugin updates in place).

The following topics provide information about tasks administrators perform when integrating external services.

Topic	Information Included
<a href="#">"Integrating Artifact Plugins " below</a>	How to configure artifact plugins for CVS, Jenkins, servers accessible using SFTP, the HP CDA server file system, and SVN.
<a href="#">"Integrating Application Deployer Plugins " on page 238</a>	How to define servers and plugins. How to test, upload and update a deployer plugin.
<a href="#">"Integrating Platform Provisioning Plugins " on page 241</a>	How to test, upload, edit, update, delete and configure a platform provisioning plugin.
<a href="#">"Integrating Existing Infrastructure " on page 246</a>	How to use the Command Line Interface to integrate existing infrastructure.

## Integrating Artifact Plugins

HP CDA provides four artifact provider plugins that integrate with external artifact repositories:

- CVS Artifact Provider Plugin: For integration with CVS source repositories.
- Jenkins Artifact Provider Plugin: For integration with Jenkins build systems.
- Server File System Artifact Provider Plugin: For integration with the file system on the HP CDA server.
- SFTP Artifact Provider Plugin: For integration with artifact servers using secure FTP (SFTP).
- SVN Artifact Provider Plugin: For integration with Subversion repositories.

Topic	Information Included
<a href="#">"Configuring a CVS Artifact Provider Plugin " on the facing page</a>	How to add, edit and delete a CVS artifact provider plugin configuration.
<a href="#">"Configuring a Jenkins Artifact Provider Plugin " on page 235</a>	How to add, edit and delete a Jenkins artifact provider plugin configuration.
<a href="#">"Configuring a Server File System Artifact Provider Plugin " on page 236</a>	How to add, edit and delete a Server File System artifact provider plugin configuration.

<a href="#">"Configuring an SFTP Artifact Provider Plugin " on page 236</a>	How to add, edit and delete an SFTP artifact provider plugin configuration.
<a href="#">"Configuring an SVN Artifact Provider Plugin " on page 238</a>	How to add, edit and delete an SVN artifact provider plugin configuration.

## Configuring a CVS Artifact Provider Plugin

**Note:** A CVS client (version 1.12.13 or higher for Linux CDA servers, CVSNT 2.5.03 or higher for Windows CDA servers) must be installed on the CDA server to enable the CVS Artifact Provider Plugin to communicate with the CVS server.

To configure a CVS artifact provider plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the "CVS Artifact Provider Plugin" entry in the Name list.

**Note:** The Configurations column displays how many configurations currently exist for the plugin.

4. *Optional:* Click the arrows to sort the list in ascending or descending order.
5. Click **Add Configuration** to open the Create Provider window.
6. In the text boxes, type:
  - Name:** for the configuration.
  - Description:** optional description.
  - CVS Host:** the fully qualified hostname to the desired CVS server.
  - CVS User:** to connect with the CVS server.
  - CVS Password:** optional password to connect with the CVS server.
  - CVS Repository:** the CVS repository base directory on the CVS server. Generally in the form of `/repositoryName`.
7. Click **Test Connection** to verify the connection.

**Note:** If the connection test hangs indefinitely, the CVS client installed on the CDA server might be incompatible with the CVS Artifact Provider Plugin implementation. To force the test connection to exit, look for the running CVS client executable on the CDA server and stop the process manually.

8. Click **Save** after verifying the connection.

## ***Configuring a Jenkins Artifact Provider Plugin***

**Note:** This plug-in requires that the HP ALI plug-in is installed and configured on the Jenkins build system. To accomplish this, download the plugin file `ali-hudson-plugin-*.hpi` to the Jenkins client and install it, and then add "ALI Integration" as a post-build action in Jenkins for the project. The plugin is contained in the `<CDA_HOME>/buildSystemPlugins` directory on the HP CDA system.

To configure a Jenkins artifact provider plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the "Jenkins Artifact Provider Plugin" entry in the Name list.

**Note:** The Configurations column displays how many configurations currently exist for the plugin.

4. *Optional:* Click the arrows to sort the list in ascending or descending order.
5. Click **Add Configuration** to open the Create Provider window.
6. In the text boxes, type:
  - Host:** URL for the Jenkins server.
  - User:** optional user name for the configuration (if a login is required).
  - Password:** optional password for the configuration (if a login is required).
7. Click **Test Connection** to verify the connection.
8. Click **Save** after verifying the connection.

## ***Configuring a Server File System Artifact Provider Plugin***

To configure a Server File System artifact provider plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the "Server File System Artifact Provider Plugin" entry in the Name list.

**Note:** The Configurations column displays how many configurations currently exist for the plugin.

4. *Optional:* Click the arrows to sort the list in ascending or descending order.
5. Click **Add Configuration** to open the Create Provider window.
6. In the text boxes, type:

**Name:** A name for this server file system artifact provider integration. This is the name that will appear in user interface elements such as drop-down lists when different artifact providers can be chosen for a given operation.

**Description:** A description for the server file system artifact provider integration (optional).

**Root/Parent Directory:** The base directory on the HP CDA server in which the artifacts are contained . The base directory can be a local drive or folder or a mapped drive in the HP CDA server

7. Click **Test Connection** to verify the connection.
8. Click **Save** after verifying the connection.

## ***Configuring an SFTP Artifact Provider Plugin***

To configure an SFTP artifact provider plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the "SFTP Artifact Provider Plugin" entry in the Name list.

**Note:** The Configurations column displays how many configurations currently exist for the plugin.

4. *Optional:* Click the arrows to sort the list in ascending or descending order.
5. Click **Add Configuration** to open the Create Provider window.
6. In the text boxes, type:

**Name:** A name for this SFTP artifact provider integration. This is the name that will appear in user interface elements such as drop-down lists when different artifact providers can be chosen for a given operation.

**Description:** A description for the SFTP server being integrated (optional).

**SFTP Server Hostname:** The host name or IP address of the SFTP server.

**User Name:** The user name required by the SFTP server for authentication.

**Note:** For SFTP authentication, either an entry must be made into the Password field for basic (User Name/Password) authentication or an authentication key must be chosen in the Authentication Key field for key-based (User Name/Authentication Key) authentication. Both fields cannot be left unfilled.

**Password:** The password to be used for authentication, if basic authentication is used.

**Authentication Key:** The name of the key file to be used for authentication, if key-based authentication is used.

**Passphrase:** The passphrase that was entered when the authentication key file was generated, if a passphrase was specified.

**Root/Parent Directory:** The base (root or parent) directory on the server in which the artifacts are contained. Enter the base directory into this field in this format: */c/<DirectoryName>* when the SFTP server is Windows.

**Timeout:** Specifies the timeout (in milliseconds) used when connecting to the SFTP server.

7. Click **Test Connection** to verify the connection.
8. Click **Save** after verifying the connection.

## Configuring an SVN Artifact Provider Plugin

**Note:** The SVN artifact provider plugin requires that an SVN command line client is installed on the CDA server.

To configure an SVN artifact provider plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the "SVN Artifact Provider Plugin" entry in the Name list.

**Note:** The Configurations column displays how many configurations currently exist for the plugin.

4. *Optional:* Click the arrows to sort the list in ascending or descending order.
5. Click **Add Configuration** to open the Create Provider window.
6. In the text boxes, type:

**SVN Repository URL:** URL for the SVN repository.

**User:** optional user name for the configuration (if a login is required).

**Password:** optional password for the configuration (if a login is required).

**Path:** full path to the SVN client executable on the HP CDA server machine, enclosed in quotes (for example, "C:\Program Files\svn\")

7. Click **Test Connection** to verify the connection.
8. Click **Save** after verifying the connection.

## Integrating Application Deployer Plugins

HP CDA uses plugins to load information about external services. A plugin is a way to communicate with a remote system.

HP CDA provides the following predefined application deployer plugins:

- OO Deployer
- DMA Deployer
- DMA 10 Deployer
- SA Deployer
- Chef Deployer
- A generic, "mock" multipurpose plugin

The following topics provide information for administrators who work with deployer plugins.

Topic	Information Included
<a href="#">"Defining a Deployer Plugin " below</a>	How to define a server plugin.
<a href="#">"Testing a Plugin Prior to Updating " on the facing page</a>	How to test a server plugin.
<a href="#">"Uploading New Plugins " on the facing page</a>	How to upload a server plugin.
<a href="#">"Updating Plugins " on the facing page</a>	How to update a server plugin.

## ***Defining a Deployer Plugin***

To define a deployer plugin:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the deployer plugin in the list.
4. Click **Add configuration** in that row to open the Create Provider window.
5. In the text boxes, type the name, (optional) description, and connection parameters to access the server.
6. Click **Test Connection** to make sure that the parameters are entered correctly.
7. Click **Save**. The server now appears in the list of available plugin configurations.

In platform Designer, for each server group in the design, the plugin appears in the drop-down list under the Deployer tab.

Select the desired deployer from the deployer list in the platform Designer.

## ***Testing a Plugin Prior to Updating***

To test a plugin prior to updating it, see "[Uploading New Plugins](#) " below. Add configurations and providers, and then create a binding or test platform and application, or update an application to use the new plugin configuration. With this process, you can test the plugin with anew configuration—without affecting the binding to the rest of the models.

After you test the plugin, you can update models by performing one of the following actions:

- Move existing models to the new plugin configurations in a phased manner.
- Re-point the plugin back to the original configuration, and then follow the procedure in "[Uploading New Plugins](#) " below to upgrade all environments to the new plugin.

## ***Uploading New Plugins***

To upload a new plugin:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Click **Upload new** to open the Upload new plugin dialog box.
4. Select either the **File** or **URL** option for **Source**.
5. Browse and select the **File** and **Server Folder** for the plugin.
6. Click **Upload**.

## ***Updating Plugins***

To update a plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the plugin from the list, and click **Update** in that row to open the Update plugin dialog box.

4. Select either the **File** or **URL** option for **Source**.
5. Browse and select the **File** and **Server Folder** for the plugin.
6. Click **Update**.

## Integrating Platform Provisioning Plugins

Currently, HP CDA platform plugins include the following:

- HP Cloudsystem Matrix
- HP Cloud Connector
- Existing Infrastructure

The following topics provide information for administrators who work with platform provisioning plugins.

Topic	Information Included
<a href="#">"Uploading a Platform Provisioning Plugin " below</a>	How to upload a platform provisioning plugin.
<a href="#">"Editing a Platform Provisioning Plugin " on the facing page</a>	How to edit a platform provisioning plugin.
<a href="#">"Updating a Platform Provisioning Plugin " on the facing page</a>	How to update a platform provisioning plugin.
<a href="#">"Deleting a Platform Provisioning Plugin " on page 243</a>	How to delete a platform provisioning plugin.
<a href="#">"Configuring a Platform Plugin " on page 243</a>	How to configure a platform provisioning plugin.

### ***Uploading a Platform Provisioning Plugin***

To upload a platform provisioning plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.

**Note:** The Plugins window lists all the plugins (platform, application, etc.) in HP CDA.

3. Click **Upload new** to open the Upload new plugin dialog box.
4. Select either the **File** or **URL** option for the **Source**.
5. Browse and select the **File** and **Server Folder** for the plugin.
6. Click **Upload**. The plugin appears in the plugin window.

## ***Editing a Platform Provisioning Plugin***

To edit basic information about a provisioning plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Click a plugin name to open its details window.
4. Click **Edit** on the right to open an edit window.
5. In the text boxes, type changes to name and description as required.
6. Click **Save**. The Plugin window displays the changes.

## ***Updating a Platform Provisioning Plugin***

**Note:** This update changes the plugin location and can influence referenced artifacts. Read and follow the warning instructions you see.

To delete a platform provisioning plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the plugin you want, and click **Update** in that row to open the Update plugin dialog box. Read the warning message.
4. Do one of the following:

Click **Cancel** and proceed using an alternate update method.

Make changes and click **Update**.

## ***Deleting a Platform Provisioning Plugin***

You must be an Administrator to perform the following task.

To delete a provisioning plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Select the plugin you want to delete.
4. Click **Delete**.
5. In the **Confirm Delete Operation** warning box, select **Advanced Options**:

**Include Subartifacts:** If checked, will delete subartifacts, such as a plugin configuration.

**Ignore Usage:** If checked, will delete the plugin, even if it is in use by another artifact.

**Caution:** When you delete with **Ignore Usage**, you might break artifacts that rely on the plugin. For example, if you select **Ignore Usage** a plugin that is used (through configuration) by an infrastructure template will be deleted. When you attempt to provision a platform using that infrastructure template, provisioning fails.

6. Click **OK**. The **Plugins** window opens and the plugin is no longer listed.

## ***Configuring a Platform Plugin***

Following are topics that provide information for administrators when they work with platform plugin configurations.

<b>Topic</b>	<b>Information Included</b>
<a href="#">"Adding a Platform Plugin Configuration " on the facing page</a>	How to add parameters to platform plugin to create a plugin configuration.
<a href="#">"Adding Authentication Keys to a Platform Plugin Configuration " on the facing page</a>	How to add and associate authentication private keys (from a centralized CDA key store) with a platform plugin configuration.

Topic	Information Included
<a href="#">"Deleting Authentication Keys from a Platform Plugin Configuration " on next page</a>	How to remove authentication keys from a platform plugin configuration.
<a href="#">"Editing a Platform Plugin Configuration " on page 246</a>	How to edit a platform plugin configuration's basic information and parameters.
<a href="#">"Deleting a Platform Plugin Configuration " on page 246</a>	How to delete a platform plugin configuration.

## ***Adding a Platform Plugin Configuration***

You must be an administrator to perform the following task.

To add a configuration to a provisioning plugin, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugins** to open the Plugins window.
3. Locate the plugin in the list and click **Add configuration** in that row to open the Create Provider window.

**Note:** The Configurations column displays how many configurations currently exist for the plugin.

4. *Optional:* Click the arrows to sort the list in ascending or descending order.
5. In the basic information text boxes, type a name and optional description for the configuration.
6. In the **Parameter Values** text boxes, enter information as requested.

**Note:** You can click **Test Connection** either before or after you save the configuration.

7. Click **Save**.

## ***Adding Authentication Keys to a Platform Plugin Configuration***

Administrators can add authentication keys to a platform plugin configuration.

**Note:** The keys are created in a separate task and reside in a centralized pool. See ["Adding Provisioning Keys to a Centralized Key Store "](#) on page 175.

To add keys to a provisioning configuration, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugin Configurations** to open the Plugin configurations window.
3. Click a plugin configuration name to open its details window.
4. Under the Authentication Keys heading, click **Add** to open the Add authentication key dialog box.
5. In the dialog box:

Choose a **Capacity pool** from the menu.

**Note:** Server groups within a template belong to a single capacity pool and will use the same private key for authentication across all servers from that server group. After provisioning, the deployers will select the corresponding private key based on the server pool name associated with the server group.

Choose an **Authentication key** file from the menu.

Click **OK**.

The dialog box closes and the key file displays in the **Authentication Keys** listing.

## ***Deleting Authentication Keys from a Platform Plugin Configuration***

Administrators can delete authentication keys from a platform plugin configuration.

To delete keys, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugin Configurations** to open the Plugin configurations window.
3. Click a plugin configuration name to open its details window.
4. Under the Authentication Keys heading, select a key.
5. Click **Delete**.
6. In the confirmation window, click **OK**.

## ***Editing a Platform Plugin Configuration***

You must be an Administrator to perform the following task.

To edit a platform plugin configuration, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugin Configurations** to open the Plugin configurations window.
3. Click a plugin configuration name to open its details window.
4. Click **Edit** on the right.
5. Make edits to the basic information or parameter values.
6. Click **Save**. The Provider window displays the edits.

## ***Deleting a Platform Plugin Configuration***

To delete a platform plugin configuration, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Plugin Configurations** to open the Plugin configurations window.
3. Select the configuration to delete.
4. Click **Delete**.
5. Click **OK** in the confirmation dialog box.

## **Integrating Existing Infrastructure**

HP CDA provides the ability to integrate using a Command Line Interface (CLI). You can use the interface to perform actions such as the following:

- List application model related artifacts.
- Run jobs related to the following:
  - Platform provisioning and de-provisioning

- Application deployment and undeployment
- Application start and stop
- Execution of an application custom operation
- Job progress
- Create and manage platforms that use existing infrastructure available in the environment.

The parameters and options that govern the usage of this tool are described below.

<b>Commands</b>	<b>Usage</b>	<b>Details</b>
<b>Model</b> - to list artifacts that are part of an application model	<code>cdaexec m list &lt;options&gt;</code>	Use this parameter to list artifacts that are part of an application model. These commands return the Name and respective UUID (Universally Unique Identifier) of the artifact— which you can use with the Job parameter.
<b>Job</b> - to run specific jobs available at platform and application level	<code>cdaexec j &lt;options&gt;</code>	Use this parameter to run jobs related to platform provisioning\deprovisioning, application deployment\undeployment, application start\stop, check job progress as well as running any custom operation defined in the application model.

<p><b>Existing Infrastructure</b> - to create and manage existing infrastructure platforms</p>	<p>cdaexec xi &lt;verb&gt; &lt;artifact&gt;</p> <p>Supported verbs - add, delete, set, get &amp; list</p> <p>Supported artifacts – server, server group &amp; template</p>	<p>You can use HP CDA to configure existing infrastructure (servers) available in the environment as a target platform for application deployment.</p> <p>With this unique feature, you can:</p> <ul style="list-style-type: none"> <li>• Create a server artifact for every existing server available in the network and ready to be used with HP CDA</li> <li>• Group these servers to be part of one or more server group artifacts</li> <li>• Create one or more Existing Infrastructure templates that are a collection of server groups</li> <li>• Display the list of services that are provisioned instances of templates</li> </ul> <p>Existing Infrastructure templates can be viewed and imported by going to the Infrastructure tab. After a template is imported, you can use it to create a platform in the same way you use templates from providers.</p>
--	--	--

**Note:** For command line options and their use, see "[HP CDA Command Line Interface \(CLI\)](#)" on page 276

## Exporting Plugins

You can export all plugins from the HP CDA repository as XML. This feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

**Note:** The ability to import plugins is not available.

To export plugins as XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Open the REST client, and issue a GET request with the following characteristics:
  - **URL:**  

```
https://<cda_  
address>:8443/cda/remote/rest/v1/administration/plugins (Secure  
HTTP)
```

```
http://<cda_  
address>:8080/cda/remote/rest/v1/administration/plugins (Standard  
HTTP)
```

Where <cda\_address> is the IP address or FQDN of the HP CDA instance.
  - **Content Type:** application/xml

If the GET request succeeds (a status code of 200 OK is returned), the XML for the plugins will be returned.
3. Copy the plugins XML to an editor for modification, or save it to a text file for later editing and import.

## Exporting and Importing Plugin Configurations

All plugin configurations can be exported from the HP CDA repository as an XML file, which can be modified if desired and imported to the repository. The XML that is exported is in an easy to read and understand format to facilitate editing in a standard XML browser or text editor.

The following topics explain how to export and import plugin configurations.

Topic	Information Contained
<a href="#">"Exporting Plugin Configurations " below</a>	How to export plugin configurations from the HP CDA repository as an XML file.
<a href="#">"Importing Plugin Configurations " on the facing page</a>	How to import a plugin configuration XML file into the HP CDA repository.

## Exporting Plugin Configurations

You can export all plugin configurations from the HP CDA repository as XML. This feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

To export all plugin configurations as XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Open the REST client, and issue a GET request with the following characteristics:

- **URL:**

```
https://<cda_
address>:8443/cda/remote/rest/v1/administration/configurations
(Secure HTTP)
```

```
http://<cda_
address>:8080/cda/remote/rest/v1/administration/configurations
(Standard HTTP)
```

Where <cda\_address> is the IP address or FQDN of the HP CDA instance.

- **Content Type:** application/xml

If the GET request succeeds (a status code of 200 OK is returned), the XML for the plugin configurations will be returned.

3. Copy the plugin configuration XML to an editor for modification, or save it to a text file for later editing and import.

## Importing Plugin Configurations

You can import plugin configuration XML to the HP CDA repository. The XML that you import has been exported using the procedure ["Exporting Plugin Configurations " on previous page](#) and typically has been stored to a text file. The plugin configuration import feature uses the HP CDA REST APIs, so you need to have a REST client available to issue requests. Many standard Web browsers have REST Client plugins available that you can download.

To import plugin configuration XML, complete the following steps:

1. Log into HP CDA on the Web browser on which the REST client is installed.
2. Open the REST client and issue a POST request with the following characteristics:

- **URL:**

```
https://<cda_
address>:8443/cda/remote/rest/v1/administration/configurations
(Secure HTTP)
```

```
https://<cda_
address>:8443-
/cda/r-
emote/rest/v1/administration/configurations?passphrase=<pass_
phrase> (Secure HTTP)
```

```
http://<cda_
address>:8080/cda/remote/rest/v1/administration/configurations
(Standard HTTP)
```

```
http://<cda_
address>:8080-
/cda/r-
emote/rest/v1/administration/configurations?passphrase=<pass_
phrase> (Standard HTTP)
```

Where <cda\_address> is the IP address or FQDN of the HP CDA instance, and <pass\_phrase> is the pass phrase that was used when the plugin configuration XML was exported.

- *Header:* application/xml
- *Body:* The plugin configuration XML to be imported to the repository.

If the POST request succeeds (a status code of 200 OK is returned), the XML for the plugin configurations has been imported into the HP CDA repository.

## Managing Configurations and Systems

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

To set configuration options, 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:
    - See "[Create Empty Data Artifacts](#) " on the facing page.
    - See "[Customizing SSL](#) " on the facing page.

- **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. See ["Changing System Settings " on next page](#).
- **License:** A summary page detailing the terms of the current license with an option to enter a new license key and manage licensed users. See ["Changing the License System " on page 254](#).
- **Self-Test:** Details the status of HP CDA with an option to disable self-test. See ["Running a Self-Test " on page 255](#).
- **Artifact Validation:** Configure policies to validate artifacts when users create or edit them. See ["Changing Policies for Artifact Form Validation " on page 256](#).
- **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.

## 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.

Select 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.

## Customizing SSL

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 settings to their defaults.

## Changing System Settings

Administrators 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. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Configuration** to open the Configuration window.
3. Click the **System Settings** tab to see settings.
4. Click **New Property** to open the New Property dialog box.
5. *Optional:* Click **Change** to alter the domain that the setting applies to.
6. Type 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. On 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. Type 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. On 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:

- Type a new value, and click **Save** to set a new value.
- Click **Load Default** to reset the value to its default installation value.

To delete system settings, complete the following steps:

1. On the System Settings tab, select the properties to delete.
2. Click **Remove** and confirm your decision.

## Changing the License System

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.

An 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. On the Administration tab Administration menu, choose **Configuration** to open the Configuration page.
3. Click the **License** tab to see current license details.
4. Click **Enter License Key** to open the Enter License Key window.
5. Click **Browse** to select the license file on your local disk.
6. Click **Finish**.

The Configuration page opens with the License tab showing the details of the new license.

## Running a Self-Test

Administrators can use a Self-Test application to run a series of checks on HP CDA's installation and configuration.

To use the self-test application, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Configuration** to open the Configuration window.
3. Click the **Self-Test** tab to see a summary, and then the details of the self-test checks.
4. Click **Enable** or **Disable**.

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*.

## Changing Policies for Artifact Form Validation

Administrators 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 access artifact validation policies, complete the following steps:

1. Click the **Administration** tab to open the Administration Home window.
2. On the Administration menu, choose **Configuration** to open the Configuration window.
1. Click the **Artifact Validation** tab to see a list of artifact types.

**Note:** Numbers next to artifact type names indicate validation policies in use.

3. Click the artifact type that you want to validate for creation and edit. The Validated By table refreshes and you see the policies used to validate the artifact type.

To add policies, complete the following steps:

1. Click **Add Policy** to open the Add Policy dialog box.
2. Use the search criteria to locate the technical policies you want to use.
3. Select the policies to use, and click **Select** to add the policies to the Validate By table.

To remove policies, complete the following steps:

1. Select the policies to remove.
2. Click **Remove Selected** and confirm your decision.

**Note:** You can extend form validation to also validate data attachments to artifacts. Set property `platform.autovalidation.validateData` to `true`.

## Running Administration Scripts

HP CDA contains several scripts that are located in the bin directory under the installation directory on the HP CDA server machine. These scripts are either batch (\*.bat) or shell (\*.sh) scripts, depending on the HP CDA server operating system.

In addition to scripts that you can run, the env script is used by other scripts to set system variables.

**Note:** Do not run the `env` script directly.

The following topics provide information about running administration scripts.

Topic	Information Included
<a href="#">"Running the <code>cdaexec</code> Administration Script " below</a>	How to run a script to run an HP CDA Command Line Interface (CLI) command.
<a href="#">"Running the <code>export</code> Administration Script " on page 259</a>	How to run a script to create and export a data image for specified components of HP CDA.
<a href="#">"Running the <code>import</code> Administration Script " on page 261</a>	How to run a script to import a data image for specified components of HP CDA.
<a href="#">"Running the <code>reset</code> Administration Script " on page 264</a>	How to run a script to reset the data for specified components of HP CDA.
<a href="#">"Running the <code>serverstart</code> Administration Script " on page 264</a>	How to run a script to start the application server on the HP CDA server machine, thus starting the HP CDA server.
<a href="#">"Running the <code>serverstop</code> Administration Script " on page 265</a>	How to run a script to stop the application server on the HP CDA server machine, thus stopping the HP CDA server.
<a href="#">"Running the <code>servicestart</code> Administration Script " on page 265</a>	How to run a script to start the HP CDA server as an OS service.
<a href="#">"Running the <code>servicestop</code> Administration Script " on page 266</a>	How to run a script to stop running HP CDA as an OS service.
<a href="#">"Running the <code>setup</code> Administration Script " on page 266</a>	How to run a script that can perform several post-installation HP CDA server platform setup tasks.
<a href="#">"Running the <code>ssltool</code> Administration Script " on page 273</a>	How to run a script to set up client-side SSL for a deployed HP CDA application, enable you to print SSL server certificates, and download the SSL server certificate chain.

## Running the `cdaexec` Administration Script

Use the `cdaexec` script to run HP CDA Command Line Interface (CLI) commands.

**Note:** If you want to run the `cdaexec` script on a remote machine (not the machine on which the HP CDA server is installed) some configuration on the remote machine is required. For more information, see ["Deploying Local Developer Builds" on page 73](#)

Script syntax is:

```
cdaexec <cli_command> [OPTIONS]
```

Where `<cli_command>` is a CLI command and `[OPTIONS]` are the general options listed in the following table. The CLI commands that can be run are explained in detail in ["HP CDA Command Line Interface \(CLI\)" on page 276](#)

The following general options are common for all CLI commands.

Option	Description
<code>-s, --server &lt;server&gt;<sup>1,2</sup></code>	The URL of the HP CDA instance for which the CLI command is being run (for example, <code>http://localhost:8080/cda</code> )
<code>-u, --user &lt;user_name&gt;<sup>1</sup></code>	A user name used to log into the above-mentioned HP CDA instance.
<code>-p, --password &lt;password&gt;<sup>1</sup></code>	The password of the HP CDA user specified by the <code>--user</code> option.
<code>-c, --config &lt;config_file&gt;</code>	The path and file name (for example, <code>c:\cda-cli\cda-cli.conf</code> ) of a file containing default general option values. If not specified in the command line, the <code>cliforcda.properties</code> file in the current user's home directory is used if it exists.

<sup>1</sup> These general options must either be pre-defined in a configuration file or defined in the command line.

<sup>2</sup> If a valid configuration file has been created and is being used, the server will default to localhost if the server option is not pre-defined in the configuration file. If no configuration file exists, then the `-s` or `--server` option must be included on the command line or the command will not run.

### Configuration File for General Options

When an HP CDA CLI command is run using `cdaexec`, you must provide the URL of the HP CDA instance for which the command is to be run as well as a user name and password used to log into that instance. You can provide that information on the command line each time you run a CLI command, or you can store the information in a configuration file for use when running CLI commands.

A configuration file can be the following:

- **cliforcda.properties file:** This is a user-created file in the home directory of the user under which the HP CDA server machine is running. If a cliforcda.properties file exists in that user's home directory and no alternate configuration file is specified on the command line, its contents will be used to run the command.
- **Alternate configuration file:** This is a user-created file that can have any file name and can exist in any directory on the HP CDA server machine. If the alternate configuration file is specified on the command line using the `-c` or `--config` option, then it will be used instead of the cliforcda.properties file during command execution.

Either configuration file must be a text only file having contents in the format `<name>=<value>`, where `<name>` is server, user, or password as in the following example:

```
server=http://localhost:8080/cda
user=admin
password=123456
```

**Note:** You can use the `configuration encryptproperties` CLI command to encrypt the password in a configuration file. Running the command with no options will encrypt the `cliforcda.properties` file, and running the command with the `--config` option will encrypt the specified alternate configuration file.

## Running the export Administration Script

Use the export script to export HP CDA images so that the data can be imported at a later date using the import script if necessary.

**Note:** HP CDA must not be running when you use the export script.

The syntax is:

```
export --image <image> [OPTIONS]
```

Where `<image>` is the target directory under which the images are to be placed (for example, `c:\cdaImages\image_2-02`), or the target directory and name of a Zip file that will contain the images (for example, `c:\cdaImages\image_2-02.zip`), and `[OPTIONS]` are the options listed in the following table.

**Note:** The target directory must be empty or the export operation will fail.

Option	Description
--------	-------------

<pre>--components &lt;c1&gt;, ..., &lt;cn&gt;</pre>	<p>A comma-separated list of the components to be exported. The following component options are available:</p> <ul style="list-style-type: none"> <li>• <b>all</b>: Exports the configuration and data except for the license. This is the default if you omit <code>--components</code>.</li> <li>• <b>Configuration</b>: Exports the configuration data.</li> <li>• <b>Content</b>: Exports all data without the configuration and security data.</li> <li>• <b>Security</b>: Exports the security configuration, including user profiles, groups, and roles and the default ACLs for newly created resources.</li> </ul>
<pre>--quiet</pre>	<p>Runs the script without needing a confirmation request.</p>
<pre>--silent</pre>	<p>Provides less verbose console output.</p>
<pre>--passphrase &lt;passphrase&gt;</pre>	<p>Specifies the master passphrase that was entered during HP CDA installation and configuration if HP CDA was installed with password encryption enabled.</p>
<pre>--executions-flat-limit &lt;N&gt;</pre>	<p>Specifies the maximum report or item count to be exported with a flat layout.</p> <p>By default, execution reports are exported in a flat layout up to a predefined number of reports and items. If the limit is exceeded, a hashed layout is used instead to prevent potential excess directory errors.</p> <p>Use this option to define your own limit for flat layout exports.</p>
<pre>--configuration-all</pre>	<p>Exports the complete configuration.</p>
<pre>--configuration-application</pre>	<p>Exports the application configuration.</p>
<pre>--configuration-system</pre>	<p>Exports the system configuration.</p>
<pre>--configuration-license</pre>	<p>Exports license details.</p>
<pre>--policyManager-validations</pre>	<p>Exports policy manager validation data.</p>
<pre>--userEvents-dontSkip</pre>	<p>Exports user event data (not exported by default).</p>

An export operation creates the directory or Zip file specified by `<image>` containing the following items, depending on the script options used:

- **image.properties:** A file containing the export execution properties and a list of the data sets exported.
- **configuration:** A directory containing the configuration properties files. The directory also contains the license details if you use the `--configuration-license` option.
- **dist:** A directory containing properties specific to a particular distribution.
- **executions:** A directory containing the execution report results of asynchronous tasks.
- **lifecycle:** A directory containing the lifecycle data.
- **platform:** A directory containing the catalog data.
- **policyManager:** A directory containing the policy data.
- **reporting:** A directory containing the reporting definitions.
- **security:** A directory containing the security configuration.
- **userEvents:** A directory containing user specific event information.

## Running the import Administration Script

Use the import script to import HP CDA images that were previously exported using the export script.

**Note:** HP CDA must not be running when you use the export script.

**Note:** When importing an image with governed artifacts using the default options, lifecycle processes with identical UUIDs in the image are not imported. To avoid inconsistencies in the lifecycle of artifacts, use the import script with the `--reset` option which deletes all existing artifacts and lifecycle processes.

The syntax is:

```
import --image <image> [OPTIONS]
```

Where **<image>** is the directory where the HP CDA image to be imported is stored (for example, `c:\cdaImages\image_2-02`), or the directory and name of the Zip file that contains the image (for example, `c:\cdaImages\image_2-02.zip`), and **[OPTIONS]** are the options listed in the following table.

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

Run the following command to update Oracle schema statistics:

```
EXEC DBMS_STATS.GATHER_SCHEMA_STATS (ownname = '&1',no_invalidate =
FALSE,options = 'GATHER');
```

This command does not require database administrator privileges and can be run by the schema owner (ownname).

Option	Description
--validate	Validates the image before it is imported.
--validate-only	Validates the image, but does not import it.
--components <c1>, ..., <cn>	A comma-separated list of the components to be imported. The following component options are available: <ul style="list-style-type: none"> <li>• <b>all:</b> Exports the configuration and data except for the license. This is the default if you omit --components.</li> <li>• <b>Configuration:</b> Exports the configuration data.</li> <li>• <b>Content:</b> Exports all data without the configuration and security data.</li> <li>• <b>Security:</b> Exports the security configuration, including user profiles, groups, and roles and the default ACLs for newly created resources.</li> </ul>
--quiet	Runs the script without needing a confirmation request.
--silent	Provides less verbose console output.
--reset	Deletes all existing artifacts and lifecycle processes before the import operation commences.
--passphrase <passphrase>	Specifies the master passphrase that was entered during HP CDA installation and configuration if HP CDA was installed with password encryption enabled.
--force	Overwrites imported resources if they are already in the database.

<code>--configuration-all</code>	Imports the complete configuration (not recommended).
<code>--configuration-application</code>	Imports the application configuration (default).
<code>--configuration-system</code>	Imports the system configuration.
<code>--configuration-license</code>	Imports license details.
<code>--configuration-passphrase</code> <code>&lt;passphrase&gt;</code>	Specifies the passphrase of the image to be imported if that image was encrypted with a passphrase that is different from the current passphrase.
<code>--userEvents-dontSkip</code>	Imports user event data, which is not imported by default. Existing user event data is deleted before the current user event data is imported if this option is used.

The import operation imports the image from the directory or Zip file specified by `<image>`, which contains the following, depending on the image:

- **image.properties:** A file containing the export execution properties and a list of the data sets exported.
- **configuration:** A directory containing the configuration properties files, including role-based UI customizations. The file also contains the license details if you use the `--configuration-license` option.
- **dist:** A directory containing properties specific to a particular distribution.
- **executions:** A directory containing the execution report results of asynchronous tasks such as publishing, discovery, and bulk operations.
- **lifecycle:** A directory containing the lifecycle data.
- **platform:** A directory containing the service catalog data.
- **policyManager:** A directory containing the policy data.
- **reporting:** A directory containing the reporting definitions.
- **security:** A directory containing the security configuration.
- **userEvents:** A directory containing user specific event information.

**Note:** If specific image components are specified via options with the import script, then other components are ignored. If a component that is specified via a script option is not present, the import operation fails.

## Running the reset Administration Script

Use the reset script to reset the HP CDA image and import the default image.

HP CDA must not be running when you run the reset script.

**Note:** HP CDA deletes all artifacts and lifecycle processes prior to the default image import. This resolves any conflicts with identical artifacts or process UUIDs between existing and imported data.

The syntax is:

```
reset [OPTIONS]
```

Where [OPTIONS] are the options listed in the following table.

Option	Description
<code>--components &lt;c1&gt;, ..., &lt;cn&gt;</code>	<p>A comma-separated list of the components to be reset. The following component options are available:</p> <ul style="list-style-type: none"> <li>• <b>all:</b> Exports the configuration and data except for the license. This is the default if you omit <code>--components</code>.</li> <li>• <b>Configuration:</b> Exports the configuration data.</li> <li>• <b>Content:</b> Exports all data without the configuration and security data.</li> <li>• <b>Security:</b> Exports the security configuration, including user profiles, groups, and roles and the default ACLs for newly created resources.</li> </ul>
<code>--quiet</code>	Runs the script without needing a confirmation request.
<code>--silent</code>	Provides less verbose console output.
<code>--passphrase &lt;passphrase&gt;</code>	Specifies the master passphrase that was entered during HP CDA installation and configuration if HP CDA was installed with password encryption enabled.

## Running the serverstart Administration Script

Use the serverstart script to start the HP CDA server.

The syntax is:

```
serverstart
```

**Note:** The `serverstart` script contains no options.

When you use the OS command prompt to run the `serverstart` script, various messages scroll in the command prompt window as the server starts. Startup can take several minutes. When a message similar to “JBoss AS 7.1.0.Final “Thunder” started in xxxxxxms” appears and messages stop scrolling, the HP CDA server has started.

## Running the `serverstop` Administration Script

Use the `serverstop` script to stop the HP CDA server.

The syntax is:

```
serverstop
```

**Note:** The `serverstop` script contains no options.

When you use the OS command prompt to run the `serverstop` script, the following message indicates that the HP CDA server has been successfully stopped:

```
{"outcome" => "success"}
```

Press any key to continue . . .

You can press any key to close the OS command prompt window.

## Running the `servicestart` Administration Script

Use the `servicestart` script to start the HP CDA server as an operating system (OS) service.

The syntax is:

```
servicestart
```

**Note:** The `servicestart` script contains no options.

When you use the OS command prompt to run the `servicestart` script, the message “Attempting to start service <service\_name>” will appear followed by one of the following messages:

- The <service\_name> service has been started but it may take few minutes before the application fully initializes.
- The <service\_name> service is starting but it may take few minutes before the application fully initializes.

- The attempt to start <service\_name> service has likely failed. Check the service status by using 'sc.exe query [service name]' command.

If you see this message in response to running the `servicestart` script, it might indicate that the service is already running.

## Running the servicestop Administration Script

Use the `servicestop` script to stop the HP CDA server operating system (OS) service.

The syntax is:

```
servicestop
```

**Note:** The `servicestop` script contains no options.

When you use the OS command prompt to run the `servicestop` script, the message “Attempting to stop service <service\_name> which may take a while” will appear followed by one of the following messages:

- The <service\_name> service has been stopped
- The <service\_name> service is stopping
- The attempt to stop <service\_name> service has likely failed. Check the service status by using 'sc.exe query [service name]' command

If you see this message in response to running the `servicestop` script, it might indicate that the service is not running.

## Running the setup Administration Script

Use the `setup` administration script to perform tasks (setup scenarios) listed in the following table.

You can use the `setup` script two ways:

- Use the script to start a GUI `setup` tool wizard. The wizard offers setup scenarios that you can perform and the steps to complete the scenarios.
- Use the script as a command line tool that you run from the OS command prompt.

**Note:** To protect data consistency, the `setup` script by default does not allow you to import data while there is a server running. In some environments (for example, behind a load balancer proxy) something is always running at the server endpoint. To enable the `setup` script in these

```
environments, run the setup script with the following Java option: -  
Dinstall.ignore.running.platform=true.
```

The following topics provide information about running the setup administration script to perform setup scenarios.

Topic	Information Included
<a href="#">"How to Start the Setup Tool Wizard " below</a>	How to start the setup tool wizard.
<a href="#">"Running the Setup Tool Wizard to Change the HP License Key " on the facing page</a>	How to use the setup tool wizard to change the HP CDA license key.
<a href="#">"Running the Setup Tool Wizard to Install Updates " on the facing page</a>	How to use the setup tool wizard to install available updates (also referred to as patches).
<a href="#">"Running the Setup Tool Wizard to Configure Server Endpoints " on page 269</a>	How to use the setup tool wizard to configure the HP CDA server endpoint IP address and port.
<a href="#">"Running the Setup Tool Wizard to Configure an SMTP Server " on page 270</a>	How to use the setup tool wizard to configure the SMTP server so that HP CDA can send notification emails.
<a href="#">"Running the Setup Tool Wizard to Change Lightweight SSO Settings " on page 271</a>	How to use the setup tool wizard to change lightweight SSO settings.
<a href="#">"Running the Setup Tool Wizard to Un-install HP CDA " on page 272</a>	How to use the setup script to un-install HP CDA.
<a href="#">"Running the setup Script as a Command Line Tool " on page 272</a>	How to run the setup script as a command line tool from the OS command prompt.

## ***How to Start the Setup Tool Wizard***

To start the setup tool wizard, complete the following steps:

1. Type **setup** without any options to open the wizard.
2. Read the Welcome window information and then click **Next**.
3. The Scenario Selection window opens. Select a scenario and click **Next**.
4. Complete the scenario steps.

## ***Running the Setup Tool Wizard to Change the HP License Key***

To change the license key, complete the following steps:

1. Store the HP CDA license key file that you received from your HP sales representative in a directory on the HP CDA server machine.
2. Run the setup script without any options to start the Setup Tool wizard.
3. Click **Next** in the Welcome window that appears.
4. In the Scenario Selection window, select the Change License Key option and then click **Next**.
5. With the Enter license key file option selected, browse to and select the license key file that you stored on the HP CDA server machine, and then click **Next**.
6. Click **Next** in the Complete window that appears. The Setup Progress bar on the next window shows the progress of the key file import operation.
7. When the key file import operation completes, click **Next**, and then click **Finish** to close the Setup Tool wizard.

**Note:** The Administration tab in the HP CDA GUI also provides support for changing the license key. For details, see ["Changing the License System " on page 254](#).

## ***Running the Setup Tool Wizard to Install Updates***

Contact your HP support representative to determine whether HP recommends any updates (patches) for your deployment.

To run the setup tool wizard to install updates, complete the following steps:

1. If HP CDA is currently running, run the `serverstop` script to stop it.  
  
For details, see ["Running the serverstop Administration Script " on page 265](#).
2. Store any HP CDA update JAR files that you received from your HP support representative in a directory on the HP CDA server machine.
3. Run the setup script without any options to start the Setup Tool wizard.
4. Click **Next** in the Welcome window that opens.

5. In the Scenario Selection window, select the **Update** option and then click **Next**.
6. Use the **Add...** button to select the updates that need to be applied to HP CDA, and then click **Next**.

The updates are the JAR files that you stored on the HP CDA server machine in Step 2.

7. Click **Next** in the Complete window that opens. The Setup Progress bar on the next window show the progress of the update.
8. When the update completes, click **Next**, and then click **Finish** to close the Setup Tool wizard.

## ***Running the Setup Tool Wizard to Configure Server Endpoints***

To configure server endpoints, complete the following steps:

1. If HP CDA is currently running, run the `serverstop` script to stop it.  
  
For details, see ["Running the serverstop Administration Script " on page 265](#)
2. Run the `setup` script without any options to start the Setup Tool wizard. Click **Next** in the Welcome window that opens.
3. In the Scenario Selection window, select the **Endpoint Settings** option and then click **Next**.
4. Modify the endpoint settings as required:
  - a. **Hostname:** The FQDN (recommended) or IP address of the server where the HP CDA web-interface will be visible to the user. Make sure that the server can be reached using the addressing scheme you type in this text box.
  - b. **Port Numbers**
    - o **HTTP:** If necessary, select the checkbox and type the port number on which the application server will listen for incoming HTTP connections. For example, JBoss listens for incoming HTTP connections on port 8080 by default.
    - o **HTTPS:** If necessary, select the checkbox and type the port number on which the application server will listen for incoming HTTP secure (HTTPS) connections. For example, JBoss listens for incoming HTTPS connections on port 8443 by default.
  - c. **Enforce HTTPS:** Select the checkbox, if necessary, to enforce secured URLs.
  - d. **Verify Certificates:** Select the checkbox, if necessary, to validate HTTPS communications against SSL certificates.

- e. **Web Context:** This entry, along with the hostname and port, will constitute the Web browser URL for the HP CDA login window; for example:  
`https://server1.dept.company.com:8443/cda`, where `cda` is the web context. The web context cannot contain spaces.
5. Click **Next**.
6. If you changed the Web Context from the default (`cda`), click **OK** in the warning box that follows to dismiss it.
7. Click **Next** in the Complete window that opens. The Setup Progress bar on the next window indicates the progress of the application of the endpoint settings.
8. When the endpoint settings application operation completes, click **Next**, and then click **Finish** to close the Setup Tool wizard.

## ***Running the Setup Tool Wizard to Configure an SMTP Server***

To configure an SMTP server, complete the following steps:

1. If HP CDA is currently running, run the `serverstop` script to stop it.  
  
For details, see ["Running the serverstop Administration Script " on page 265](#)
2. Run the `setup` script without any options to start the Setup Tool wizard.
3. Click **Next** in the Welcome window that opens.
4. In the Scenario Selection window, select the **SMTP Settings** option, and then click **Next**.
5. Modify the SMTP server settings as required. SMTP server setting descriptions are as follows:
  - a. **Server:** The hostname of the SMTP server that will act as the outgoing mail server for HP CDA.
  - b. **Authenticate:** Select this checkbox if the connection between HP CDA and the SMTP server will be secured using SSL.
  - c. **Username:** The username to be used by HP CDA for SSL authentication, if the Authenticate option is selected.
  - d. **Password:** The password to be used by HP CDA for SSL authentication, if the Authenticate option is selected.
  - e. Click **Next**.

6. Click **Next** in the Complete window that opens. The Setup Progress bar on the next window indicates the progress of the application of the SMTP settings.
7. When the SMTP settings application operation completes, click **Next**, and then click **Finish** to close the Setup Tool wizard.

## ***Running the Setup Tool Wizard to Change Lightweight SSO Settings***

To change lightweight SSO settings, complete the following steps:

1. If HP CDA is currently running, run the `serverstop` script to stop it.  
  
For details, see ["Running the serverstop Administration Script " on page 265](#)
2. Run the `setup` script without any options to start the Setup Tool wizard.
3. Click **Next** in the Welcome window that opens.
4. In the Scenario Selection window, select the **Lightweight SSO Settings** option, and then click **Next**.
5. Select one of the following options for the type of HP Lightweight SSO configuration to use:
  - **Basic**: If you want to use basic HP Lightweight SSO configuration information.
  - **Advanced**: If you want to use advanced HP Lightweight SSO configuration information.
6. Click **Next**.
7. If you selected the **Basic** option, modify the basic LWSSO configuration information as required:
  - a. **Domain**: The domain that includes the systems participating in SSO. You cannot use localhost or a domain name including an IP address.
  - b. **Pass-phrase**: The pass-phrase being used by all HP products using LWSSO. A pass-phrase can be one or more UTF-8 characters, but HP recommends that standard password best practices be implemented when choosing a pass-phrase. The pass-phrase is case-sensitive.
  - c. **Confirm Pass-phrase**: Enter the LWSSO pass-phrase again.
  - d. Click **Next**.
8. If you selected the Advanced option, modify the advanced LWSSO configuration information

as required:

- a. **Lightweight SSO Configuration File:** Browse to and select the lightweight SSO configuration file. You noted the directory and name of this file earlier in this procedure.
  - b. **Keystore with Lightweight SSO Keys:** Browse to and select the keystore file containing the lightweight SSO keys. You noted the directory and name of this file earlier in this procedure.
  - c. Click **Next**.
9. Click **Next** in the Complete window that opens. The Setup Progress bar on the next window indicates the progress of the application of the lightweight SSO settings.
  10. When the lightweight SSO settings application operation completes, click **Next**, and then click **Finish** to close the Setup Tool wizard.

## ***Running the Setup Tool Wizard to Un-install HP CDA***

To uninstall HP CDA, complete the following steps:

1. If HP CDA is currently running, run the `serverstop` script to stop it.  
  
For details, see ["Running the serverstop Administration Script " on page 265](#)
2. Run the `setup` script without any options to start the Setup Tool wizard.
3. Click **Next** in the Welcome window that opens.
4. In the Scenario Selection window, select the **Un-install** option, and then click **Next**.
5. Click **Next** in the Complete window that opens. The Setup Progress bar on the next window indicates the progress of the un-install operation.
6. When the un-install operation completes, click **Next**, and then click **Finish** to close the Setup Tool wizard.

## ***Running the setup Script as a Command Line Tool***

The `setup` script can be run as a command line tool.

The syntax is:

```
setup [OPTIONS]
```

Where [OPTIONS] are the options listed in the following table.

Option	Description
-h, --help [scenarios steps]	<ul style="list-style-type: none"> <li>• --help Displays help for the available command options.</li> <li>• --help scenarios Displays help for the various setup scenarios.</li> <li>• --help steps Displays the steps taken by default by the setup script when each setup scenario is run.</li> </ul>
-c, --console	Runs the setup script in console mode.
-a, --dbadmin-mode	Turns on the database administrator mode. The setup stops to allow the database administrator to run SQL scripts to modify the database. Continue setup by running setup -c.
-n, --scenario <scenario_name>	Specifies a scenario to run. Use --help scenarios to list the available scenarios.
-p, --steps <s1>, ..., <sn>	Runs a custom scenario consisting of only the comma-separated list of specified steps. Use --help steps to list the available steps.
-u, --use-config <custom.properties_ file>	Specifies the path to a custom property file. Properties from the specified file override the default or current configuration properties.
--passphrase <passphrase>	Specifies the master passphrase that was entered during HP CDA installation and configuration if HP CDA was installed with password encryption enabled.
-d, --debug	Enables debug mode. In debug mode, all properties are visible in the log file as well as SQL statements and other details.

## Running the ssltool Administration Script

Use the `ssltool` script to set up client-side SSL for HP CDA. It also can be used to print SSL server certificates and download the SSL server certificate chain.

The syntax is:

```
ssltool <action> [OPTIONS]
```

Where **<action>** is one of the actions listed under Actions, and **[OPTIONS]** are the options explained in the tables that follow.

Actions:

- **serverInfo**: This action prints the SSL requirements for the specified HTTPS URL, and saves the server certificate to a file.

- **keystoreEI**: This action exports or imports SSL certificates to the HP CDA database keystore or truststore.
- **customize**: This action changes the effective SSL customization.

The following **[OPTIONS]** are available for the `serverInfo` action:

Option	Description
<code>-u, --url &lt;url&gt;</code>	Specifies the https URL for which to print SSL server requirements.
<code>-h, --hostname &lt;hostname&gt;</code>	The hostname (not required when <code>--url</code> is specified).
<code>-p, --port &lt;port&gt;</code>	The port (not required when <code>--url</code> is specified).
<code>-f, --certFile &lt;certfile_name&gt;</code>	Specifies the file to which to save the server certificates (optional).
<code>--certIndex &lt;index&gt;</code>	Specifies an index of a certificate to save: 0 - server, 1 server certificate issuer; (optional).
<code>--test</code>	Tests the URL connection.
<code>-d, --debug</code>	Enables debug mode to print debug messages.
<code>-s, ---silent</code>	Disables printing of server information. This option makes sense only when the <code>--certFile</code> option is specified.
<code>--help</code>	Displays help for the available command options.
<code>--passphrase &lt;passphrase&gt;</code>	Specifies the master passphrase that was entered during HP CDA installation and configuration if HP CDA was installed with password encryption enabled.

The following **[OPTIONS]** are available for the `keystoreEIaction`:

Option	Description
<code>-i, --import &lt;string&gt;</code>	Imports the keystore or truststore to the platform database ( <code>&lt;string&gt;</code> must be either keystore or truststore).
<code>-e, --export &lt;string&gt;</code>	Exports the keystore or truststore from the platform database ( <code>&lt;string&gt;</code> must be either keystore or truststore).
<code>--keystore &lt;keystore&gt;</code>	The keystore file to import or export.

--storepass <password> - -storepass file:<file>	The keystore password (<password>). The password can be stored in a file (<file>) if file: is used in the option.
--storetype <type>	The type of java keystore, either JKS or PKCS12.
--alias <alias>	The alias to export or import (optional). The <alias> is a user-specific name given to the entry (key, certificate, etc.) stored in the keystore.
--keypass <key-password>	The key password of the alias to import (optional).
--overwrite	Overwrites the target keystore.
--add	Adds to the target keystore.
--certfile <file>	Imports the trusted certificate from the specified file.
--withpassword <file>	Exports the database password to the specified file.  Use the --withpassword option with no argument to export the database password to a file named "<keystore>.pwd."
-d, --debug	Enables debug mode to print debug messages.
--help	Displays help for the available command options.
--passphrase <passphrase>	Specifies the master passphrase that was entered during HP CDA installation and configuration if HP CDA was installed with password encryption enabled.

The following [**OPTIONS**] are available for the `customize` action:

Option	Description
-i, --info, --list	Lists the SSL customization that is currently in effect under the heading "Effective customization," followed by a list of available SSL customizations under the heading "Available customizations."
-c, --change <customization>	Changes the effective SSL customization. the <customization> argument must be one of those listed by the -i, --info, or --list option.
-d, --debug	Enables debug mode to print debug messages.
--help	Displays help for the available command options.
--passphrase <passphrase>	Specifies the master passphrase that was entered during HP CDA installation and configuration if HP CDA was installed with password encryption enabled.

## Shared Content

The following topics contain material for HP CDA help that is shared by two or more topics. For example, you can add parameters in several HP CDA locations.

Topic	Information Included
<a href="#">"HP CDA Command Line Interface (CLI)" below</a>	Information about Command Line Interface commands, including name, alias, synopsis, description and options.
<a href="#">"HP Diagnostics Probe Deployment" on page 341</a>	How to import platform software for installing Diagnostics probes, configure platform software to point to the Diagnostics distribution, and configure a platform to attach the platform software.
<a href="#">"Configuring Monitoring Tools to Send Events to HP CDA" on page 343</a>	How to configure various monitoring tools to send events to HP CDA.
<a href="#">"Managing Parameters" on page 350</a>	How to add parameters, use stored parameter sets, use substitution variables and reference parameters.
<a href="#">"Canceling Provisioning or Deployment Operations" on page 368</a>	How to cancel, back out and clean up from failed deployment or provisioning.
<a href="#">"HP CDA Workflows" on page 372</a>	How to use workflows such as Start, Stop, Deploy and so on.
<a href="#">"HP CDA Programming Operations" on page 372</a>	How to use HP CDA programming operations such as Placed File and Executed Script.
<a href="#">"For More Information" on page 392</a>	Links and information about information resources available from the HP Software Product Manuals Web site.

## HP CDA Command Line Interface (CLI)

### HP CDA Command Line Interface Commands Summary

#### Bundle Commands

bundle delete <Bundleid>
bundle get <Bundleid>
bundle get bundledefinition <Bundledefinitionid>
bundle promote <Bundleid>
<b>Configuration Commands</b>
configuration encryptproperties
<b>Model Commands</b>
model list application [-d domain]
model list applicationversion -a application
model list applicationmodel [-a application] [-n name] [-av applicationversion] [-d domain]
model list bundle [-bd bundledefinition] [-a application] [-n name] [-av applicationversion] [-d domain] [-rt realizedtopology]
model list bundledefinition [-a application] [-n name] [-av applicationversion] [-d domain]
model list domain [-cwd current]
model list logicaltopology [-am applicationmodel] [-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 realizedtopologymonitorparameters [-cp] -rt realizedtopology [-mt monitoringtool]
model list realizedtopology [-am applicationmodel] [-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]
model list pluginconfig
<b>Job Commands</b>
job provision -n name -lp logicalplatform -lp sp logicalplatformstoredparameters [-rb rollbackOnError]

job deploy -n <i>name</i> -lt <i>logicaltopology</i> -ltsp <i>logicaltopologystoredparameters</i> [-lpsp <i>logicalplatformstoredparameters</i> ] [-rp <i>realizedplatform</i> ] [-ds <i>deploymentset</i> ] [-rb <i>rollbackOnError</i> ]
job undeploy -rt <i>realizedtopology</i>
job deprovision -rp <i>realizedplatform</i>
job checkprogress -jr <i>jobreport</i>
job start -rt <i>realizedtopology</i>
job stop -rt <i>realizedtopology</i>
job customoperation -rt <i>realizedtopology</i> -co <i>customoperation</i>
job bundle -av <i>applicationversion</i> -bd <i>bundledefinition</i> [-bn <i>bundlename</i> ] [-fp <i>filepath</i> ]
<b>Existing Infrastructure Commands</b>
existinginfrastructure add server -hn <i>hostname</i> -ip <i>ip</i> [-down] [-d <i>domainId</i> ]
existinginfrastructure set server < <i>Hostname</i> > [-hn <i>hostname</i> ] [-ip <i>ip</i> ] [-d <i>domainId</i> ]
existinginfrastructure list server [-d <i>domainId</i> ]
existinginfrastructure delete server < <i>Hostname</i> > [-d <i>domainId</i> ]
existinginfrastructure get server < <i>Hostname</i> > [-d <i>domainId</i> ]
existinginfrastructure add servergroup -nm <i>name</i> [-a <i>arch</i> ] -os <i>ostype</i> [-min <i>min</i> ] [-max <i>max</i> ] [-cpu <i>cpucount</i> ] [-maxcpu <i>maxcpucount</i> ] [-mem <i>memsize</i> ] [-maxmem <i>maxmemsize</i> ] [-memType <i>memsizeType</i> ] [-svr <i>attachserver</i> ] [-d <i>domainId</i> ]
existinginfrastructure set servergroup < <i>Name</i> > [-nm <i>name</i> ] [-a <i>arch</i> ] [-os <i>ostype</i> ][-min <i>min</i> ] [-max <i>max</i> ] [-cpu <i>cpucount</i> ] [-maxcpu <i>maxcpucount</i> ] [-mem <i>memsize</i> ] [-maxmem <i>maxmemsize</i> ] [-memType <i>memsizeType</i> ] [-svr <i>attachserver</i> ] [-detachAllsvrs] [-d <i>domainId</i> ]
existinginfrastructure list servergroup [-d <i>domainId</i> ]
existinginfrastructure delete servergroup < <i>Name</i> > [-force <i>force</i> ] [-d <i>domainId</i> ]
existinginfrastructure get servergroup < <i>Name</i> > [-d <i>domainId</i> ]
existinginfrastructure list service [-d <i>domainId</i> ]
existinginfrastructure delete service <ID> [-d <i>domainId</i> ]
existinginfrastructure add template -nm <i>name</i> [-desc <i>description</i> ] [-svrgrp <i>attachservergroup</i> ] [-d <i>domainId</i> ]

<code>existinginfrastructure set template &lt;Name&gt; [-nm name] [-desc description] [-srvgrp attachservergroup] [-d domainId]</code>
<code>existinginfrastructure list template [-d domainId]</code>
<code>existinginfrastructure delete template &lt;Name&gt;[-force force] [-d domainId]</code>
<code>existinginfrastructure get template &lt;Name&gt; [-d domainId]</code>
<b>Realized Infrastructure Commands</b>
<code>realizedinfrastructure get ipaddressid -ipaddr ipaddress</code>
<code>realizedinfrastructure get hostname -id ipaddressid</code>
<code>realizedinfrastructure get capacitypoolid -id ipaddressid</code>
<code>realizedinfrastructure get identifierid -id ipaddressid</code>
<code>realizedinfrastructure get licensefeatureid -id ipaddressid</code>
<code>realizedinfrastructure get servergroup -id ipaddressid</code>
<code>realizedinfrastructure list ipaddresses -srvgrpId servergroupid</code>
<code>realizedinfrastructure list softwareinstances -id ipaddressid</code>
<code>realizedinfrastructure list applayers -id ipaddressid</code>
<code>realizedinfrastructure list networkinterfaces -id ipaddressid</code>
<b>Provisioner Plugin Service Commands</b>
<code>provisionerpluginservice assign elasticip -ipaddrId ipaddressid [-nicid networkinterfacecardid] -pgcfigId pluginConfigId</code>
<code>provisionerpluginservice move elasticip -ipaddrId ipaddressid -nicid networkinterfacecardid -eip elasticipaddress -force forceownerchange -pgcfigId pluginConfigId</code>
<code>provisionerpluginservice remove elasticip -ipaddrId ipaddressid -eip elasticipaddress -pgcfigId pluginConfigId</code>
<code>provisionerpluginservice release elasticip -ccpid cloudcapacitypoolid -eip elasticipaddress -pgcfigId pluginConfigId</code>
<code>provisionerpluginservice list availableelasticip -ccpid cloudcapacitypoolid -pgcfigId pluginConfigId</code>
<code>provisionerpluginservice list assignedelasticip -ipaddrId ipaddressid [-nicid networkinterfacecardid] -pgcfigId pluginConfigId</code>
<code>provisionerpluginservice get publicip -ipaddrId ipaddressid [-nicid networkinterfacecardid] -pgcfigId pluginConfigId</code>
<b>External Infrastructure Commands</b>

externalinfrastructure add samanagedserver <pluginConfigId> -t <i>template</i> -sg <i>servergroup</i> [-f <i>filter</i> ] [-d <i>domainId</i> ]
externalinfrastructure list samanagedserver <pluginConfigId> [-f <i>filter</i> ]

Bundle Command		
<b>NAME</b>	bundle delete	
<b>ALIAS</b>	b delete	
<b>SYNOPSIS</b>	bundle delete <BundleId>	
<b>DESCRIPTION</b>	Deletes a specified bundle. Specify bundle UUID.	
<b>OPTIONS</b>	<BundleId>	Bundle UUID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Bundle Command		
<b>NAME</b>	bundle get	

<b>ALIAS</b>	b get	
<b>SYNOPSIS</b>	bundle get <Bundleid>	
<b>DESCRIPTION</b>	Fetches bundle information. Specify bundle UUID.	
<b>OPTIONS</b>	. <Bundleid>	Bundle UUID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Bundle Command</b>		
<b>NAME</b>	bundle get bundledefinition	
<b>ALIAS</b>	b get bdldf	
<b>SYNOPSIS</b>	bundle get bundledefinition <Bundledefinitionid>	
<b>DESCRIPTION</b>	Fetches bundle definition information. Specify bundle definition UUID.	

<b>OPTIONS</b>	< <i>Bundledefinitionid</i> >	A bundle definition UUID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Bundle Command</b>		
<b>NAME</b>	bundle promote	
<b>ALIAS</b>	b promote	
<b>SYNOPSIS</b>	bundle promote < <i>Bundleid</i> >	
<b>DESCRIPTION</b>	Requests promotion of a bundle to next life cycle stage. Specify bundle UUID.	
<b>OPTIONS</b>	< <i>Bundleid</i> >	A bundle UUID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Configuration Command</b>		
<b>NAME</b>	configuration encryptproperties	
<b>ALIAS</b>	config encryptproperties	
<b>SYNOPSIS</b>	configuration encryptproperties	
<b>DESCRIPTION</b>	Encrypts the password in the properties file.	
<b>OPTIONS</b>	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Model Command		
<b>NAME</b>	model list application	
<b>ALIAS</b>	m list app	
<b>SYNOPSIS</b>	model list application [-d <i>domain</i> ]	
<b>DESCRIPTION</b>	Lists application(s). Applications can be listed by domain. If domain is not specified, all applications are returned.	
<b>OPTIONS</b>	-d, --domain	A domain UUID. If provided, all applications related to the specified to domain will be listed.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Model Command		
<b>NAME</b>	model list applicationversion	
<b>ALIAS</b>	m list av	
<b>SYNOPSIS</b>	model list applicationversion -a <i>application</i>	

<b>DESCRIPTION</b>	Lists application version(s) by application UUID.	
<b>OPTIONS</b>	-a, --application	An application UUID
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Model Command</b>		
<b>NAME</b>	model list applicationmodel	
<b>ALIAS</b>	m list am	
<b>SYNOPSIS</b>	model list applicationmodel [-a <i>application</i> ] [-n <i>name</i> ] [-av <i>applicationversion</i> ] [-d <i>domain</i> ]	
<b>DESCRIPTION</b>	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.	
<b>OPTIONS</b>	-a, --application	An application UUID

	-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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Model Command		
<b>NAME</b>	model list bundle	
<b>ALIAS</b>	m list bdl	
<b>SYNOPSIS</b>	model list bundle [-bd <i>bundledefinition</i> ] [-a <i>application</i> ] [-n <i>name</i> ] [-av <i>applicationversion</i> ] [-d <i>domain</i> ] [-rt <i>realizedtopology</i> ]	
<b>DESCRIPTION</b>	Lists bundle(s). Bundles can be listed by either name, bundle definition id, application id, application version id or domain. If no argument is specified, all bundles are returned. No more than one argument can be supplied to this operation.	

<b>OPTIONS</b>	-bd, --bundledefinition	A bundle definition UUID. If provided, only the bundles for this bundle definition will be listed.
	-a, --application	An application UUID. If provided, only the bundles for this application will be listed.
	-n, --name	A bundle name. If more than one bundle shares the same name then the UUID for each bundle of that name will be returned.
	-av, --applicationversion	An application version UUID. If provided only the bundles for this application version will be listed.
	-d, --domain	A domain UUID. If provided, all bundles related to the specified domain will be listed.
	-rt, --realizedtopology	A realized topology UUID. If provided, the bundle related to the specified realized topology will be listed.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>&lt;name&gt;=&lt;value&gt;</code> ", where <code>&lt;name&gt;</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.
<p>Must be pre-defined in a configuration file or defined in the command line. <code>--server(-s)</code> will default to localhost value(s) if not pre-defined in a configuration file.</p>		

<b>Model Command</b>	
<b>NAME</b>	model list bundledefinition

<b>ALIAS</b>	m list bdldef	
<b>SYNOPSIS</b>	model list bundledefiniton [-a <i>application</i> ] [-n <i>name</i> ] [-av <i>applicationversion</i> ] [-d <i>domain</i> ]	
<b>DESCRIPTION</b>	Lists bundle definition(s). Bundle Definitions can be listed by either name, application id, application version id or domain. If no argument is specified, all bundle definitions are returned. No more than one argument can be supplied to this operation.	
<b>OPTIONS</b>	-a, --application	An application UUID. If provided, only the bundle definitions for this application will be listed.
	-n, --name	A bundle definition name. If more than one bundle definition shares the same name then the UUID for each bundle definition of that name will be returned.
	-av, --applicationversion	An application version UUID. If provided only the bundles for this application version will be listed.
	-d, --domain	A domain UUID. If provided, all bundle definitions related to the specified domain will be listed.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Model Command		
<b>NAME</b>	model list domain	
<b>ALIAS</b>	m list domain	
<b>SYNOPSIS</b>	model list domain [-cwd <i>current</i> ]	
<b>DESCRIPTION</b>	Lists domains. If "--current" is specified will only list the current working domain for the user defined in the "--user" general parameter.	
<b>OPTIONS</b>	-cwd, --current	If specified, return only the working domain of the current user.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Model Command		
<b>NAME</b>	model list logicaltopology	
<b>ALIAS</b>	m list lt	
<b>SYNOPSIS</b>	model list logicaltopology [-am <i>applicationmodel</i> ] [-d <i>domain</i> ] [-n <i>name</i> ]	

<b>DESCRIPTION</b>	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.	
<b>OPTIONS</b>	-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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Model Command		
<b>NAME</b>	model list logicalplatform	
<b>ALIAS</b>	m list lp	
<b>SYNOPSIS</b>	model list logicalplatform [-d <i>domain</i> ] [-n <i>name</i> ] [-lt <i>logicaltopology</i> ]	
<b>DESCRIPTION</b>	Lists all logical platforms. Select only one option from: -d, -n, or -lt.	

<b>OPTIONS</b>	--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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Model Command</b>		
<b>NAME</b>	model list monitoringtool	
<b>ALIAS</b>	m list mt	
<b>SYNOPSIS</b>	model list monitoringtool	
<b>DESCRIPTION</b>	Lists available monitoring tools.	
<b>OPTIONS</b>	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Model Command</b>		
<b>NAME</b>	model list realizedtopology	
<b>ALIAS</b>	m list rt	
<b>SYNOPSIS</b>	model list realizedtopology [-am <i>applicationmodel</i> ] [-n <i>name</i> ] [-lt <i>logicaltopology</i> ] [-d <i>domain</i> ]	
<b>DESCRIPTION</b>	Lists all realized topologies. Select only one option from: -am, -n, -lt, or -d.	
<b>OPTIONS</b>	-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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Model Command</b>		
<b>NAME</b>	model list realizedtopologymonitors	
<b>ALIAS</b>	m list rtmn	
<b>SYNOPSIS</b>	model list realizedtopologymonitors -rt <i>realizedtopology</i> [-mt <i>monitoringtool</i> ]	
<b>DESCRIPTION</b>	Lists monitors deployed in the given realized topology. If monitoringtool is specified, results are filtered to only include monitors deployed for given monitoring tool.	
<b>OPTIONS</b>	-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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Model Command		
<b>NAME</b>	model list realizedtopologymonitorparameters	
<b>ALIAS</b>	m list rtmpn	
<b>SYNOPSIS</b>	model list realizedtopologymonitorparameters [-cp] -rt <i>realizedtopology</i> [-mt <i>monitoringtool</i> ]	
<b>DESCRIPTION</b>	In the given realized topology, lists deployed monitors and their parameters. If monitoringtool is specified, results are filtered to only include monitors deployed for given monitoring tool.	
<b>OPTIONS</b>	-cp, --commonparms	A realizedtopology UUID. Identifies the realizedtopology to search.
	-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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Model Command		
<b>NAME</b>	model list realizedplatform	
<b>ALIAS</b>	m list rp	
<b>SYNOPSIS</b>	model list realizedplatform [-d <i>domain</i> ] [-lt <i>logicaltopology</i> ] [-lp <i>logicalplatform</i> ] [-n <i>name</i> ]	
<b>DESCRIPTION</b>	Lists all realized platforms. Select only one option from: -d, -lt, -lp, or -n.	
<b>OPTIONS</b>	-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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Model Command		
<b>NAME</b>	model list storedparameters	
<b>ALIAS</b>	m list sp	
<b>SYNOPSIS</b>	model list storedparameters [-lt <i>logicaltopology</i> ] [-lp <i>logicalplatform</i> ] [-rt <i>realizedtopology</i> ] [-rp <i>realizedplatform</i> ]	
<b>DESCRIPTION</b>	Lists the stored parameters for a given model object specified by its UUID. Only one model object UUID may be specified.	
<b>OPTIONS</b>	-lt, --logicaltopology	A logical topology UUID.
	-lp, --logicalplatform	A logical platform UUID.
	-rt, --realizedtopology	A realized topology UUID.
	-rp, --realizedplatform	A realized platform UUID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Model Command													
<b>NAME</b>	model list deploymentset												
<b>ALIAS</b>	m list ds												
<b>SYNOPSIS</b>	model list deploymentset [-lt <i>logicaltopology</i> ] [-av <i>applicationversion</i> ]												
<b>DESCRIPTION</b>	Lists deployment sets for a given logical topology or application version.												
<b>OPTIONS</b>	<table border="0"> <tr> <td>-lt, --logicaltopology</td> <td>A logical topology UUID.</td> </tr> <tr> <td>-av, --applicationversion</td> <td>An application version UUID.</td> </tr> <tr> <td>-s, --server</td> <td>The URL of the CDA instance (e.g. 'http://localhost:8080/cda').</td> </tr> <tr> <td>-u, --user</td> <td>The CDA username.</td> </tr> <tr> <td>-p, --password</td> <td>The password for the CDA user.</td> </tr> <tr> <td>-c, --config</td> <td>A file containing default general option values in the format "&lt;name&gt;=&lt;value&gt;", where &lt;name&gt; is "server", "user", or "password". If not specified, the cliforcda.properties file in the current user's home directory is used if it exists.</td> </tr> </table>	-lt, --logicaltopology	A logical topology UUID.	-av, --applicationversion	An application version UUID.	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').	-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.
-lt, --logicaltopology	A logical topology UUID.												
-av, --applicationversion	An application version UUID.												
-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').												
-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.												
<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.</p>													

Model Command	
<b>NAME</b>	model list pluginconfig
<b>ALIAS</b>	m list plgconf

<b>SYNOPSIS</b>	model list pluginconfig	
<b>DESCRIPTION</b>	Lists Plugin Configuration information.	
<b>OPTIONS</b>	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Job Command</b>		
<b>NAME</b>	job bundle	
<b>ALIAS</b>	j bundle	
<b>SYNOPSIS</b>	job bundle -av <i>applicationversion</i> -bd <i>bundledefinition</i> [-bn <i>bundlename</i> ] [-fp <i>filepath</i> ]	
<b>DESCRIPTION</b>	<p>Creates a bundle.</p> <p>User can specify either an existing bundle definition or a new bundledefinition on which the bundle will be created.</p> <p>Filepaths can be specified to update the existing artifacts of the specified bundle definition. If not specified, the existing bundle definition is packaged to create a new bundle.</p> <p>When a new bundle definition name is provided, the Default bundle definition is cloned to create a bundle definition and this is packaged to create a new bundle.</p>	

<b>OPTIONS</b>	-av, -- applicationversion	An application version UUID.
	-bd, -- bundledescription	A bundle definition name. This can be either an existing bundle definition or a new bundle definition.
	-bn, --bundlename	A bundle instance name. If not specified, a unique bundle instance name is generated.
	-fp, --filepath	File paths.. Specify a file path in the format -fp <name of the artifact to be updated>=<artifactpath to be uploaded> .  The name of the artifact specified in the file path should be an existing artifact for an existing bundle definition. User will not be allowed to add new artifacts. A file path can either point to a file or directory. For directories, optionally include or exclude filter can be mentioned <DirectoryArtifactPath>[[[inc exc]=filterpattern].  Multiple paths can be provided.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Job Command</b>	
<b>NAME</b>	job provision
<b>ALIAS</b>	j provision

<b>SYNOPSIS</b>	<code>job provision -n name -lp <i>logicalplatform</i> -lpsp <i>logicalplatformstoredparameters</i> [-rb <i>rollbackOnError</i>]</code>	
<b>DESCRIPTION</b>	<p>Provisions a realized platform from a logical platform and logical platform stored parameters.</p> <p>If the 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' command.</p>	
<b>OPTIONS</b>	<code>-n, --name</code>	A name from which the name of the realized platform will be derived.
	<code>-lp, --logicalplatform</code>	A logical platform UUID from which the realized platform will be provisioned.
	<code>-lpsp, --logicalplatformstoredparameters</code>	A logical platform stored parameters UUID.
	<code>-rb, --rollbackOnError</code>	Rollback the operation when an error occurs. Defaults to true.
	<code>-s, --server</code>	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	<code>-u, --user</code>	The CDA username.
	<code>-p, --password</code>	The password for the CDA user.
	<code>-c, --config</code>	<p>A file containing default general option values in the format "<code>&lt;name&gt;=&lt;value&gt;</code>", where <code>&lt;name&gt;</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>
<p>Must be pre-defined in a configuration file or defined in the command line. <code>--server(-s)</code> will default to localhost value(s) if not pre-defined in a configuration file.</p>		

<b>Job Command</b>		
<b>NAME</b>	job deploy	

<b>ALIAS</b>	j deploy	
<b>SYNOPSIS</b>	job deploy -n <i>name</i> -lt <i>logicaltopology</i> -ltsp <i>logicaltopologystoredparameters</i> [-lpsp <i>logicalplatformstoredparameters</i> ] [-rp <i>realizedplatform</i> ] [-ds <i>deploymentset</i> ] [-rb <i>rollbackOnError</i> ]	
<b>DESCRIPTION</b>	<p>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.</p> <p>If the 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' command.</p>	
<b>OPTIONS</b>	-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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Job Command		
<b>NAME</b>	job undeploy	
<b>ALIAS</b>	j undeploy	
<b>SYNOPSIS</b>	job undeploy -rt <i>realizedtopology</i>	
<b>DESCRIPTION</b>	<p>Undeploy a realized topology.</p> <p>If the 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' command.</p>	
<b>OPTIONS</b>	-rt, --realizedtopology	A realized topology UUID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Job Command		
<b>NAME</b>	job deprovision	
<b>ALIAS</b>	j deprovision	
<b>SYNOPSIS</b>	job deprovision -rp <i>realizedplatform</i>	
<b>DESCRIPTION</b>	<p>Deprovisions a specified realized platform.</p> <p>If the 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' command.</p>	
<b>OPTIONS</b>	-rp, --realizedplatform	A realized platform UUID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Job Command		
<b>NAME</b>	job checkprogress	
<b>ALIAS</b>	j checkprogress	

<b>SYNOPSIS</b>	<code>job checkprogress -jr <i>jobreport</i></code>	
<b>DESCRIPTION</b>	Displays the progress information for a specified CDA job report.	
<b>OPTIONS</b>	<code>-jr, --jobreport</code>	A job report UUID.
	<code>-s, --server</code>	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	<code>-u, --user</code>	The CDA username.
	<code>-p, --password</code>	The password for the CDA user.
	<code>-c, --config</code>	A file containing default general option values in the format " <code>&lt;name&gt;=&lt;value&gt;</code> ", where <code>&lt;name&gt;</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.
<p>Must be pre-defined in a configuration file or defined in the command line. <code>--server(-s)</code> will default to localhost value(s) if not pre-defined in a configuration file.</p>		

<b>Job Command</b>		
<b>NAME</b>	<code>job start</code>	
<b>ALIAS</b>	<code>j start</code>	
<b>SYNOPSIS</b>	<code>job start -rt <i>realizedtopology</i></code>	
<b>DESCRIPTION</b>	<p>Executes the start operation on the specified realized topology.</p> <p>If the 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" command..</p>	
<b>OPTIONS</b>	<code>-rt, --realizedtopology</code>	A realized topology UUID.

	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Job Command</b>		
<b>NAME</b>	job stop	
<b>ALIAS</b>	j stop	
<b>SYNOPSIS</b>	job stop -rt <i>realizedtopology</i>	
<b>DESCRIPTION</b>	<p>Executes the stop operation on the specified realized topology.</p> <p>If the 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' command.</p>	
<b>OPTIONS</b>	-rt, -- realizedtopology	A realized topology UUID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Job Command		
<b>NAME</b>	job customoperation	
<b>ALIAS</b>	j customoperation	
<b>SYNOPSIS</b>	job customoperation -rt <i>realizedtopology</i> -co <i>customoperation</i>	
<b>DESCRIPTION</b>	<p>Executes the named custom workflow that has been defined as part of an application model.</p> <p>If the 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 command.</p>	
<b>OPTIONS</b>	-rt, --realizedtopology	A realized topology UUID.
	-co, --customoperation	The name of the custom workflow that has been defined for an application model.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.

Existing Infrastructure Command		
<b>NAME</b>	existinginfrastructure add server	
<b>ALIAS</b>	xinfra add svr	
	xi add svr	
<b>SYNOPSIS</b>	existinginfrastructure add server -hn <i>hostname</i> -ip <i>ip</i> [-down] [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	<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>	
<b>OPTIONS</b>	-hn, --hostname	A Server hostname.
	-ip, --ip	A Server IP address.
	-down, --inActive	To indicate that server is temporarily down and cannot be used.
	-d, --domainID	Domain Identifier
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>Existing Infrastructure Command</b>		
<b>NAME</b>	existinginfrastructure set server	
<b>ALIAS</b>	xinfra set svr	
	xi set svr	
<b>SYNOPSIS</b>	existinginfrastructure set server <Hostname> [-hn <i>hostname</i> ] [-ip <i>ip</i> ] [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	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.	
<b>OPTIONS</b>	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 URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Existing Infrastructure Command		
<b>NAME</b>	existinginfrastructure list server	
<b>ALIAS</b>	xinfra list svr	
	xi list svr	
<b>SYNOPSIS</b>	existinginfrastructure list server [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	Lists all available servers.	
<b>OPTIONS</b>	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>&lt;name&gt;=&lt;value&gt;</i> ", where <i>&lt;name&gt;</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.
<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.</p>		

Existing Infrastructure Command		
<b>NAME</b>	existinginfrastructure delete server	
<b>ALIAS</b>	xinfra delete svr	
	xi delete svr	
<b>SYNOPSIS</b>	existinginfrastructure delete server <i>&lt;Hostname&gt;</i> [-d <i>domainId</i> ]	

<b>DESCRIPTION</b>	Deletes a server. Specify server hostname.	
<b>OPTIONS</b>	Hostname	A Server hostname.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Existing Infrastructure Command</b>		
<b>NAME</b>	existinginfrastructure get server	
<b>ALIAS</b>	xinfra get svr	
	xi get svr	
<b>SYNOPSIS</b>	existinginfrastructure get server <Hostname> [-d domainId]	
<b>DESCRIPTION</b>	Fetches server information.	
<b>OPTIONS</b>	Hostname	A Server hostname.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Existing Infrastructure Command</b>	
<b>NAME</b>	existinginfrastructure add servergroup
<b>ALIAS</b>	xinfra add svrgrp
	xi add svrgrp
<b>SYNOPSIS</b>	existinginfrastructure add servergroup -nm <i>name</i> [-a <i>arch</i> ] -os <i>ostype</i> [-min <i>min</i> ] [-max <i>max</i> ] [-cpu <i>cpucount</i> ] [-maxcpu <i>maxcpucount</i> ] [-mem <i>memsize</i> ] [-maxmem <i>maxmemsize</i> ] [-memType <i>memsizeType</i> ] [-svr <i>attachserver</i> ] [-d <i>domainId</i> ]
<b>DESCRIPTION</b>	<p>Adds a server group.</p> <p>Specify type of operating system (os) of servers to be allocated to the server group. Specify "Windows", "Linux" or "Unknown", if the os is undetermined at this stage.</p> <p>Remember to set operating system using the set servergroup command before using this server group for provisioning. Expect failures if this is not done!</p> <p>To attach more than one server to server group, specify "--attacheserver" multiple times.</p> <p>Specify "--domainId" to indicate which domain the specified server group should be added to.</p>

<b>OPTIONS</b>	-nm, --name	A server group name.
	-a, --arch	Common CPU architecture of servers in server group.
	-os, --ostype	Common operating system of servers in server group.
	-min, --min	Minimum server count.
	-max, --max	Maximum server count.
	-cpu, --cpucount	Common Processor Count of Servers in Server Group.
	-maxcpu, --maxcpucount	Common Maximum Processor Count of Servers in Server Group.
	-mem, --memsize	Memory of Servers in Server Group.
	-maxmem, --maxmemsize	Maximum Memory of Servers in Server Group.
	-memType, --memsizeType	Unit of Memory of Servers in Server Group.
	-svr, --attachserver	A server hostname.
	-d, --domainId	Domain Identifier
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Existing Infrastructure Command</b>	
<b>NAME</b>	existinginfrastructure set servergroup

<b>ALIAS</b>	xinfra set svrgrp	
	xi set svrgrp	
<b>SYNOPSIS</b>	existinginfrastructure set servergroup <Name> [-nm <i>name</i> ] [-a <i>arch</i> ] [-os <i>ostype</i> ] [-min <i>min</i> ] [-max <i>max</i> [-cpu <i>cpucount</i> ] [-maxcpu <i>maxcpucount</i> ] [-mem <i>memsize</i> ] [-maxmem <i>maxmemsize</i> ] [-memType <i>memsizeType</i> ] [-svr <i>attachserver</i> ] [-detachAllsvrs] [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	<p>Edits server group information. Specify server group name.</p> <p>Specify "--name" to change the name of server group.</p> <p>Specify "--ostype" to change the operating system type of the server group. Specify "Windows", "Linux" or "Unknown", if the operating system is undetermined at this stage.</p> <p>Remember to set operating system using the set servergroup command before using this server group for provisioning. Expect failures if this is not done!</p> <p>Specify "--attachserver" to attach servers to servergroup. To attach more than one server to server group, specify "--attachserver" multiple times.</p> <p>Specify "--domainId" to indicate which domain the specified server group should be added to.</p>	
<b>OPTIONS</b>	Name	A Server Group name.
	-nm, --name	A Server Group new name.
	-a, --arch	Common CPU architecture of servers in server group..
	-os, --ostype	Common Operating System of Servers in Server Group.
	-min, --min	Minimum server count.
	-max, --max	Maximum server count.
	-cpu, --cpucount	Common Processor Count of Servers in Server Group.

	-maxcpu, --maxcpucount	Common Maximum Processor Count of Servers in Server Group.
	-mem, --memsize	Memory of Servers in Server Group.
	-maxmem, --maxmemsize	Maximum Memory of Servers in Server Group.
	-memType, --memsizeType	Unit of Memory of Servers in Server Group.
	-svr, --attachserver	A server hostname.
	-detachAllsvrs, --detachAllservers	Detaches all available servers from this Server Group.
	-d, --domainId	Domain Identifier
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.

Existing Infrastructure Command	
<b>NAME</b>	existinginfrastructure list servergroup
<b>ALIAS</b>	xinfra list svrgrp xi list svrgrp
<b>SYNOPSIS</b>	existinginfrastructure list servergroup [-d <i>domainId</i> ]

<b>DESCRIPTION</b>	<p>Lists all available server groups.</p> <p>Specify "--domainId" to indicate the domain whose server groups need to be listed.</p>	
<b>OPTIONS</b>	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-u, --user	The CDA username.
	-p, --password	The password for the CDA user.
	-c, --config	<p>A file containing default general option values in the format "&lt;name&gt;=&lt;value&gt;", where &lt;name&gt; 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>
<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.</p>		

<b>Existing Infrastructure Command</b>		
<b>NAME</b>	existinginfrastructure delete servergroup	
<b>ALIAS</b>	xinfra delete svrgrp xi delete svrgrp	
<b>SYNOPSIS</b>	existinginfrastructure delete servergroup <Name> [-force <i>force</i> ] [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	<p>Deletes a server group.</p> <p>Specify server group name.</p> <p>Specify "--domainId" to indicate which domain the specified server group belongs to.</p>	
<b>OPTIONS</b>	Name	A Server Group name.

	-force, --force	Force delete a server group in use.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Existing Infrastructure Command		
<b>NAME</b>	existinginfrastructure get servergroup	
<b>ALIAS</b>	xinfras get svrgrp	
	xi get svrgrp	
<b>SYNOPSIS</b>	existinginfrastructure get servergroup <Name> [-d domainId]	
<b>DESCRIPTION</b>	<p>Fetches server group information.</p> <p>Specify server group name.</p> <p>Specify "--domainId" to indicate which domain the specified server group belongs to.</p>	
<b>OPTIONS</b>	Name	A Server Group name.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').

	-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.
<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.</p>		

<b>Existing Infrastructure Command</b>		
<b>NAME</b>	existinginfrastructure list service	
<b>ALIAS</b>	xinfra list svc	
	xi svc	
<b>SYNOPSIS</b>	existinginfrastructure list service [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	Lists provisioned instances of templates, also known as services.	
<b>OPTIONS</b>	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Existing Infrastructure Command</b>		
<b>NAME</b>	existinginfrastructure delete service	
<b>ALIAS</b>	xinfra delete svc	
	xi delete svc	
<b>SYNOPSIS</b>	existinginfrastructure delete service <ID> [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	<p>Deletes a service. Specify service id.</p> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p><b>Caution:</b> Use the Deprovision command when you want to delete a service and return its servers to the available servers list. Only use <code>delete service</code> to manually override standard HP CDA and clean up an Existing Infrastructure environment in extreme cases.</p> </div>	
<b>OPTIONS</b>	<ID>	A Service ID.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>&lt;name&gt;=&lt;value&gt;</code> ", where <code>&lt;name&gt;</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.
<p>Must be pre-defined in a configuration file or defined in the command line. <code>--server(-s)</code> will default to localhost value(s) if not pre-defined in a configuration file.</p>		

**Existing Infrastructure Command**

<b>NAME</b>	existinginfrastructure add template	
<b>ALIAS</b>	xinfra add tmplt	
	xi add tmplt	
<b>SYNOPSIS</b>	existinginfrastructure add template -nm <i>name</i> [-desc <i>description</i> ] [-svrgrp <i>attachservergroup</i> ] [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	<p>Adds a template.</p> <p>To attach more than one server group to template, specify "--attachservergroup" multiple times.</p>	
<b>OPTIONS</b>	-nm, --name	A template name.
	-desc, --description	A template description.
	-svrgrp, --attachservergroup	A Server Group name.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Existing Infrastructure Command</b>	
<b>NAME</b>	existinginfrastructure set template

<b>ALIAS</b>	xinfra set tmplt	
	xi set tmplt	
<b>SYNOPSIS</b>	existinginfrastructure set template <Name> [-nm name] [-desc description] [-svrgrp attachservergroup] [-d domainId]	
<b>DESCRIPTION</b>	<p>Edits template information. Specify template name.</p> <p>Specify "--name" to change the name of template.</p> <p>Specify "--attachservergroup" to attach server groups to template. To attach more than one server group to template, specify "--attachservergroup" multiple times."</p>	
<b>OPTIONS</b>	Name	A Template name.
	-nm, --name	A Template new name.
	-desc, --description	A Template description.
	-svrgrp, --attachservergroup	A Server Group name.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

**Existing Infrastructure Command**

<b>NAME</b>	existinginfrastructure list template	
<b>ALIAS</b>	xinfra list tmplt	
	xi list tmplt	
<b>SYNOPSIS</b>	existinginfrastructure list template [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	Lists all available templates.	
<b>OPTIONS</b>	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>Existing Infrastructure Command</b>		
<b>NAME</b>	existinginfrastructure delete template	
<b>ALIAS</b>	xinfra delete tmplt	
	xi delete tmplt	
<b>SYNOPSIS</b>	existinginfrastructure delete template <Name> [-force <i>force</i> ] [-d <i>domainId</i> ]	
<b>DESCRIPTION</b>	Deletes a template. Specify template name.	

<b>OPTIONS</b>	Name	A template name.
	-force, --force	Force delete a template in use (used by a service).
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Existing Infrastructure Command</b>		
<b>NAME</b>	existinginfrastructure get template	
<b>ALIAS</b>	xinfra get tmplt	
	xi get   tmplt	
<b>SYNOPSIS</b>	existinginfrastructure get template <Name> [-d domainId]	
<b>DESCRIPTION</b>	Fetches template information. Specify template name.	
<b>OPTIONS</b>	Name	A Template name.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').

	-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.
<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.</p>		

<b>Realized Infrastructure Command</b>		
<b>NAME</b>	realizedinfrastructure get ipaddressid	
<b>ALIAS</b>	rinfra get ipid	
	ri get ipid	
<b>SYNOPSIS</b>	realizedinfrastructure get ipaddressid -ipaddr <i>ipaddress</i>	
<b>DESCRIPTION</b>	Gets the ID of the given ipaddress.	
<b>OPTIONS</b>	-ipaddr, --ipaddress	The IP address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Realized Infrastructure Command		
<b>NAME</b>	realizedinfrastructure get hostname	
<b>ALIAS</b>	rinfra get hn	
	ri get hn	
<b>SYNOPSIS</b>	realizedinfrastructure get hostname -id <i>ipaddressid</i>	
<b>DESCRIPTION</b>	Gets the hostname of the given ipaddress id.	
<b>OPTIONS</b>	-id, --ipaddressid	The IP Address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Realized Infrastructure Command		
<b>NAME</b>	realizedinfrastructure get capacitypoolid	
<b>ALIAS</b>	rinfra get ccpid	
	ri get ccpid	
<b>SYNOPSIS</b>	realizedinfrastructure get capacitypoolid -id <i>ipaddressid</i>	

<b>DESCRIPTION</b>	Gets the capacity pool id of the given ipaddress id.	
<b>OPTIONS</b>	-id, --ipaddressid	The IP Address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Realized Infrastructure Command</b>		
<b>NAME</b>	realizedinfrastructure get identifierid	
<b>ALIAS</b>	rinfra get idfid	
	ri get idfid	
<b>SYNOPSIS</b>	realizedinfrastructure get identifierid -id <i>ipaddressid</i>	
<b>DESCRIPTION</b>	Gets the identifier id of the given ipaddress id.	
<b>OPTIONS</b>	-id, --ipaddressid	The IP Address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Realized Infrastructure Command</b>		
<b>NAME</b>	realizedinfrastructure get licensefeatureid	
<b>ALIAS</b>	rinfra get licid	
	ri get licid	
<b>SYNOPSIS</b>	realizedinfrastructure get licensefeatureid -id <i>ipaddressid</i>	
<b>DESCRIPTION</b>	Gets the license feature id of the given ipaddress id	
<b>OPTIONS</b>	-id, --ipaddressid	The IP Address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Realized Infrastructure Command		
<b>NAME</b>	realizedinfrastructure get servergroup	
<b>ALIAS</b>	rinfra get svrgrp	
	ri get svrgrp	
<b>SYNOPSIS</b>	realizedinfrastructure get servergroup -id <i>ipaddressid</i>	
<b>DESCRIPTION</b>	Gets the server group UUID and name of the given realized server.	
<b>OPTIONS</b>	-id, --ipaddressid	The IP Address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.		

Realized Infrastructure Command		
<b>NAME</b>	realizedinfrastructure list ipaddresses	
<b>ALIAS</b>	rinfra list ips	
	ri list ips	
<b>SYNOPSIS</b>	realizedinfrastructure list ipaddresses -svrgrp <i>servergroupid</i>	

<b>DESCRIPTION</b>	Lists all of the realized IP addresses associated with the server group id.	
<b>OPTIONS</b>	-srvrgrpId, --servergroupid	The server group ID.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Realized Infrastructure Command</b>		
<b>NAME</b>	realizedinfrastructure list softwareinstances	
<b>ALIAS</b>	rinfra list swinsts	
	ri list swinsts	
<b>SYNOPSIS</b>	realizedinfrastructure list softwareinstances -id <i>ipaddressid</i>	
<b>DESCRIPTION</b>	Lists all software instances deployed in the given realized ipaddress id.	
<b>OPTIONS</b>	-id, --ipaddressid	The IP Address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Realized Infrastructure Command</b>		
<b>NAME</b>	realizedinfrastructure list applayers	
<b>ALIAS</b>	rinfra list aplys	
	ri list aplys	
<b>SYNOPSIS</b>	realizedinfrastructure list applayers -id <i>ipaddressid</i>	
<b>DESCRIPTION</b>	Lists all application layers deployed in a given realized ipaddress id.	
<b>OPTIONS</b>	-id, --ipaddressid	The IP Address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Realized Infrastructure Command</b>		
<b>NAME</b>	realizedinfrastructure list networkinterfaces	
<b>ALIAS</b>	rinfra list nwints	
	ri list nwints	
<b>SYNOPSIS</b>	realizedinfrastructure list networkinterfaces -id <i>ipaddressid</i>	
<b>DESCRIPTION</b>	Lists all network interfaces with the name and address associated with a given realized ipaddress id.	
<b>OPTIONS</b>	-id, --ipaddressid	The IP Address ID of the realized server.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>Provisioner Plugin Service Command</b>		
<b>NAME</b>	provisionerpluginservice assign elasticip	
<b>ALIAS</b>	prplgnsvc assign eip	
	ppns assign eip	

<b>SYNOPSIS</b>	provisionerpluginservice assign elasticip -ipaddrid <i>ipaddressid</i> [-nicid <i>networkinterfacecardid</i> ] -pgcfgid <i>pluginConfigId</i>	
<b>DESCRIPTION</b>	Assigns an Elastic IP Address to a realized/provisioned public cloud instance.	
<b>OPTIONS</b>	-ipaddrid, --ipaddressid	The Server IP Address Id
	-nicid, --networkinterfacecardid	The Network Interface Card Id
	-pgcfgid, --pluginConfigId	Provisioner Plugin Configuration Id
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>Provisioner Plugin Service Command</b>		
<b>NAME</b>	provisionerpluginservice move elasticip	
<b>ALIAS</b>	prplgnsvc move eip	
	ppns move eip	
<b>SYNOPSIS</b>	provisionerpluginservice move elasticip -ipaddrid <i>ipaddressid</i> -nicid <i>networkinterfacecardid</i> -eip <i>elasticipaddress</i> -force <i>forceownerchange</i> -pgcfgid <i>pluginConfigId</i>	

<b>DESCRIPTION</b>	Moves an Elastic IP address from the realized/provisioned public cloud instance to another realized/provisioned public cloud instance, or allocates an Elastic IP address from the pool to a realized/provisioned public cloud instance.	
<b>OPTIONS</b>	-ipaddrid, --ipaddressid	The Server IP Address Id
	-nicid, --networkinterfacecardid	The Network Interface Card Id
	-eip, --elasticipaddress	The Elastic IP Address
	-force, --forceownerchange	Set true/false to indicate whether the elastic ip should be moved forcibly. If set false, it will ask for the provisioner admin approval. Ex:HP Cloud System Matrix Admin approval. Please use it carefully.
	-pgcfigid, --pluginConfigId	Provisioner Plugin Configuration Id
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>Provisioner Plugin Service Command</b>		
<b>NAME</b>	provisionerpluginservice remove elasticip	
<b>ALIAS</b>	prplgnsvc remove eip	
	ppns remove eip	

<b>SYNOPSIS</b>	<code>provisionerpluginservice remove elasticip -ipaddrid <i>ipaddressid</i> -eip <i>elasticipaddress</i> -pgcfigid <i>pluginConfigId</i></code>	
<b>DESCRIPTION</b>	Removes an Elastic IP address from a realized/provisioned public cloud instance permanently.	
<b>OPTIONS</b>	-ipaddrid, -- ipaddressid	The Server IP Address Id
	-eip, -- elasticipaddress	The Elastic IP Address
	-pgcfigid, -- pluginConfigId	Provisioner Plugin Configuration Id
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>&lt;name&gt;=&lt;value&gt;</code> ", where <code>&lt;name&gt;</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. <code>--server(-s)</code> will default to localhost value(s) if not pre-defined in a configuration file.		

<b>Provisioner Plugin Service Command</b>		
<b>NAME</b>	<code>provisionerpluginservice release elasticip</code>	
<b>ALIAS</b>	<code>prplgnsvc release eip</code>	
	<code>ppns release eip</code>	
<b>SYNOPSIS</b>	<code>provisionerpluginservice release elasticip -ccpid <i>cloudcapacitypoolid</i> -eip <i>elasticipaddress</i> -pgcfigid <i>pluginConfigId</i></code>	

<b>DESCRIPTION</b>	Releases an Elastic IP address from the cloud capacity pool permanently.	
<b>OPTIONS</b>	-ccpid, -- cloudcapacitypoolid	Cloud Capacity Pool Name
	-eip, -- elasticipaddress	The Elastic IP Address
	-pgcfgid, -- pluginConfigId	Provisioner Plugin Configuration Id
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>Provisioner Plugin Service Command</b>		
<b>NAME</b>	provisionerpluginservice list availableelasticip	
<b>ALIAS</b>	prplgnsvc list availableelasticip	
	ppns list availableelasticip	
<b>SYNOPSIS</b>	provisionerpluginservice list availableelasticip -ccpid <i>cloudcapacitypoolid</i> -pgcfgid <i>pluginConfigId</i>	
<b>DESCRIPTION</b>	Lists the available Elastic IP(s) from the given cloud capacity pool, configured with provisioner plugin.	

<b>OPTIONS</b>	-ccpid, -- cloudcapacitypoolid	Cloud Capacity Pool Name
	-pgcfgid, -- pluginConfigId	Provisioner Plugin Configuration Id
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

<b>Provisioner Plugin Service Command</b>		
<b>NAME</b>	provisionerpluginservice list assignedelasticip	
<b>ALIAS</b>	prplgnsvc list assignedelasticip	
	ppns list assignedelasticip	
<b>SYNOPSIS</b>	provisionerpluginservice list assignedelasticip -ipaddrId <i>ipaddressid</i> [-nicid <i>networkinterfacecardid</i> ] -pgcfgid <i>pluginConfigId</i>	
<b>DESCRIPTION</b>	Lists the Elastic IP(s) assigned to a realized/provisioned public cloud instance.	
<b>OPTIONS</b>	-ipaddrId, --ipaddressid	The Server IP Address Id
	-nicid, -- networkinterfacecardid	The Network Interface Card Id

	-pgcfigid, -- pluginConfigId	Provisioner Plugin Configuration Id
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

Provisioner Plugin Service Command		
<b>NAME</b>	provisionerpluginservice get publicip	
<b>ALIAS</b>	prplgnsvc get pubip	
	ppns get pubip	
<b>SYNOPSIS</b>	provisionerpluginservice get publicip -ipaddrId <i>ipaddressid</i> [-nicid <i>networkinterfacecardid</i> ] -pgcfigid <i>pluginConfigId</i>	
<b>DESCRIPTION</b>	Fetches the Public IP address of a realized/provisioned public cloud instance.	
<b>OPTIONS</b>	-ipaddrId, --ipaddressid	The Server IP Address Id
	-nicid, -- networkinterfacecardid	The Network Interface Card Id
	-pgcfigid, -- pluginConfigId	Provisioner Plugin Configuration Id

	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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.
<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.</p>		

External Infrastructure Command	
<b>NAME</b>	externalinfrastructure add samanagedserver
<b>ALIAS</b>	exinfra add sasvr ei add sasvr
<b>SYNOPSIS</b>	externalinfrastructure add samanagedserver <pluginConfigId> -t <i>template</i> -sg <i>servergroup</i> [-f <i>filter</i> ] [-d <i>domainId</i> ]
<b>DESCRIPTION</b>	<p>Adds SA managed servers into existing infrastructure server group.</p> <p>Specify an existing infrastructure template and server group name to associate the SA servers, optionally specify the SA server filter attributes.</p> <p>Optionally specify "--filter" for filter conditions. Use delimiter ":". Ex: -f OSFAMILYTYPE:X64. To apply more than one filter, specify "--filter" multiple times. Allowed filter attributes are OSFAMILYTYPE, CPUCOUNT, MEMORY, FACILITYNAME, CUSTOMERNAME, AGENTVERSION, BAY, ENCLOSURE, HOSTNAME, MANUFACTURERNAME, RACKNAME, REALM, SERVERID, IPADDRESS, OSVERSION.</p> <p>Operating system for the server group is used as default filter, hence it needs to be updated correctly. Specify "--domainId" to indicate which domain the specified server group belongs to.</p>

<b>OPTIONS</b>	<pluginConfigId>	A Plugin Configuration UUID
	-t, --template	An Existing Infrastructure Template name.
	-sg, --servergroup	A Server Group name that belongs to the specified template.
	-f, --filter	SA managed server filter condition.
	-d, --domainId	Domain Identifier.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-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>&lt;name&gt;=&lt;value&gt;</code> ", where <code>&lt;name&gt;</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.
<p>Must be pre-defined in a configuration file or defined in the command line. <code>--server(-s)</code> will default to localhost value(s) if not pre-defined in a configuration file.</p>		

<b>External Infrastructure Command</b>	
<b>NAME</b>	externalinfrastructure list samanagedserver
<b>ALIAS</b>	exinfra list sasrvr ei list sasrvr
<b>SYNOPSIS</b>	externalinfrastructure list samanagedserver <pluginConfigId> [-f filter]

<b>DESCRIPTION</b>	<p>Fetches List of SA Managed Servers.</p> <p>Optionally specify "--filter" for filter conditions. Use delimiter ":". Ex: -f OSTYPE:Linux. To apply more than one filter, specify "--filter" multiple times. Allowed filter attributes are OSTYPE, OSFAMILYTYPE, CPUCOUNT, MEMORY, FACILITYNAME, CUSTOMERNAME, AGENTVERSION, BAY, ENCLOSURE, HOSTNAME, MANUFACTURERNAME, RACKNAME, REALM, SERVERID, IPADDRESS, OSVERSION.</p>	
<b>OPTIONS</b>	<pluginConfigId>	A Plugin Configuration UUID
	-f, --filter	SA filter.
	-s, --server	The URL of the CDA instance (e.g. 'http://localhost:8080/cda').
	-u, --user	The CDA username.
	-p, --password	The password for the CDA user.
	-c, --config	<p>A file containing default general option values in the format "&lt;name&gt;=&lt;value&gt;", where &lt;name&gt; 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>
<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.</p>		

## Floating or Elastic IP Addresses

Floating or elastic IP management is made available in HP CDA in order to provide extended support for cloud infrastructure through the HP Matrix Operating Environment [HP MOE). Floating

or elastic IP is used to access cloud instances in a public cloud through the Internet. This IP is not bound to an instance but by the cloud vendor. There is a life cycle associated with the floating IP.

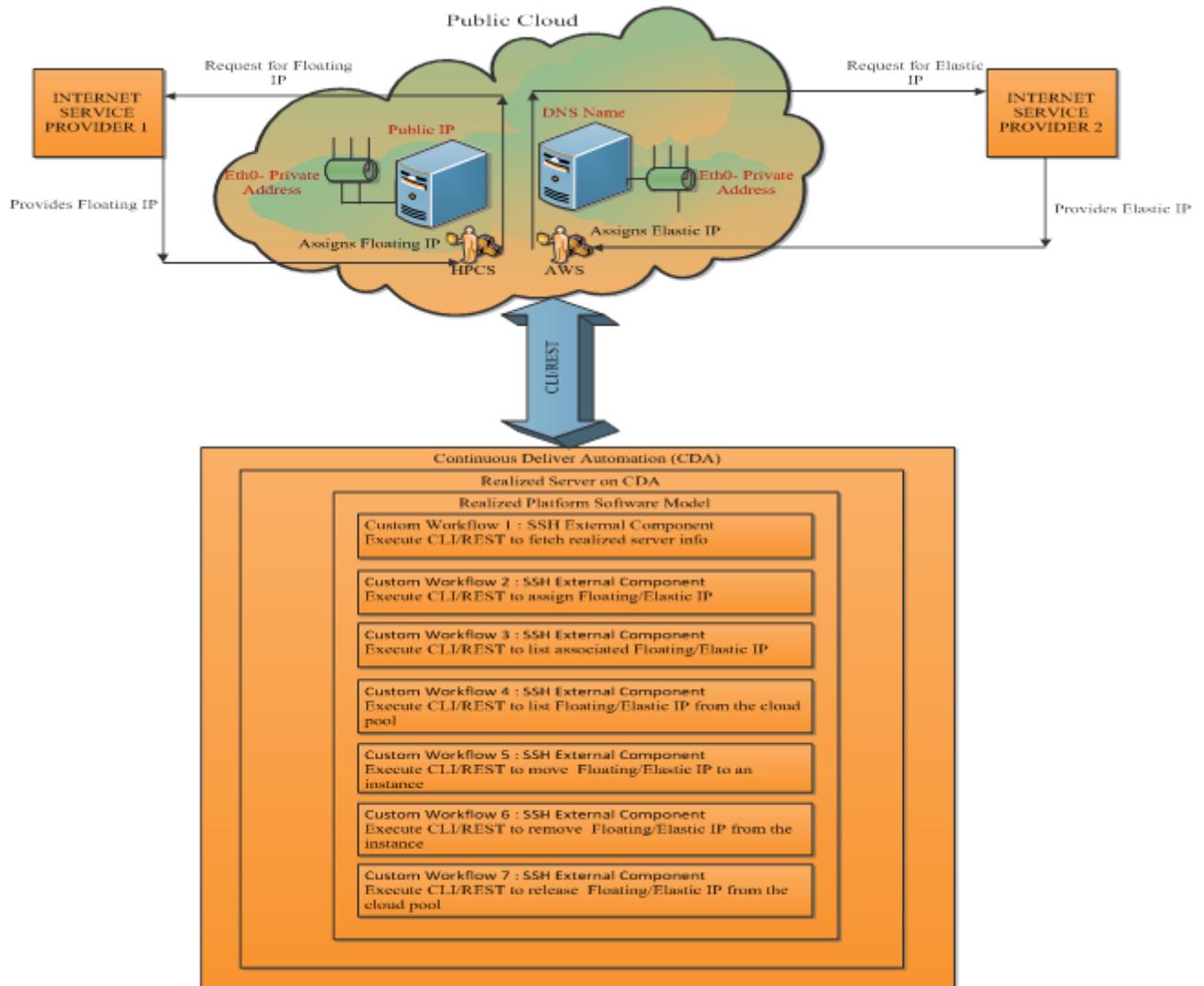
A floating IP can be:

- fetched from the cloud vendor and added to the pool
- assigned to a running cloud instance
- moved to different running cloud instance
- removed from a cloud instance and made available again in the pool
- released from an cloud instance or pool and given back to the cloud vendor

All of the above operations can be performed using the HP CDA Command Line Interface (CLI) or REST-based API interface. Refer to the Provisioner Plugin Service CLI commands for more information.

A floating IP can be fetched as part of the platform software model and can be assigned to a particular instance at the time of provisioning itself, as part of a post-provisioning task, or it can be added as custom action to assign a floating IP. A platform software model can be created to manage the Floating IP. The assigned floating IP can be fetched again and used for endpoints and connections manually. The IP can be used again for configuring the application DB or App Servers to connect to each other if they require a public interface to connect each other.

The following figure depicts a sample realized platform model present in the realized server which is provisioned out of a public cloud using HP CDA. This can manage the floating and elastic IP address using various custom workflows and components. The custom workflow in the realized platform model can be executed once the machine is provisioned. Hence the floating and elastic IP address can be associated, un-associated, or moved from the instance on demand and in an automated way by a single click once the machine is up and running.



## 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 **Platforms** from the highest level tabs.
2. Click the **Model Archive** link that is in the navigation pane at the left side of the screen.
3. Click **File**.
4. Click **Browse...** and select the model Diagnostics probe platform software archive file.
5. Click **Import** to start the archive import process.

### **Configure platform software to point to the Diagnostics distribution**

1. Click **Platforms** from the highest level tabs.
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.

### **Configure the Logical Platform to Attach the Platform Software**

1. Click the **Platforms** tab.
2. Click the **My Platforms** link that on the left.
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.

## Configuring Monitoring Tools to Send Events to HP CDA

This appendix contains the steps for configuring for the following tools to send events to HP CDA:

- HP Diagnostics
- HP SiteScope
- HP Operations Manager (HP OM)
- Nagios

## Configuring HP Diagnostics to Send Events to HP CDA

To configure HP Diagnostics to send events to HP CDA:

1. In the HP Diagnostics configuration user interface (which is separate from its performance analysis user interface), click the **Configuration** link.
2. In the HP Diagnostics Configuration menu, click the **Alert Properties** link.
3. In the Alert Properties definition, set “Enable Script Execution on Alerts” to true.
4. Click **Show Advanced Options**.

5. In Advanced Options for Alert Properties, specify the following:

Requirement	Description
Enable Script Execution alerts	Change the value of “Minimum time between events for a single rule” from the default value of 1800 to 1 milliseconds. This value represents the minimum number of milliseconds between the firing of any two event alert events for a given alert rule. The value cannot be 0.
Specify a URL for posting events to the CDA host	<code>http://&lt;cda host&gt;:&lt;CDA port&gt;/mon-sis-wer/sisreceiver</code>  You can test this URL from a browser in the HP Diagnostics host to get a timestamp from HP CDA, thus verifying connectivity.
Script status mapping	Available status definitions are: <ul style="list-style-type: none"> <li>■ Unavailable</li> <li>■ Warning</li> <li>■ Good</li> <li>■ Error</li> </ul> <p><b>Note:</b> The status mapping script cannot contain spaces or tabs.</p>

6. Click **Submit**.

**Note:** HP Diagnostics is agent-based, so when you deploy it you also deploy all of the agents. Also, the HP Diagnostics agent for Windows, Linux and .NET is already installed with HP CDA, and you can use it during software provisioning.

For additional details about alert properties in HP Diagnostics, see the HP Diagnostics product documentation.

## Configuring HP SiteScope to Send Events to HP CDA

When you configure HP SiteScope to send events to CDA, be prepared to specify the following items:

Requirement	Description
-------------	-------------

Search/filter tag	Used to tag the templates that you want to send events to CDA. SiteScope doesn't send all the events; rather, it sends only events from deployed templates that use this tag.
URL for posting events to CDA host	An HTTP Recipient containing the URL for posting events with the search/filter tag to the HP CDA host. For example: http://<cda host>:<CDA port>/mon-sis-wer/sisreceiver.
An event integration preference	Binds the search/filter tag with a connector. In this case, a connector is an HTTP preference where a URL is configured for event reception in CDA.
Templates	Templates must: <ul style="list-style-type: none"><li>• Reference the HP CDA URL via a tag</li><li>• Use HP CDA Event Mapping</li></ul>

To configure HP SiteScope to send events to HP CDA:

1. In the HP SiteScope user interface, click **Preference** in the left panel to make the Search/Filter Tags visible.
2. Click the **Search/Filter Tags** tab.
3. Click **New Tag**.
4. In the New Tag dialog, enter "cda receiver" in the Name field for the new tag and also specify the Tag Values. Optionally, provide a description.

**Note:** You can place as many tags as you want in your template. In other words, you can send events to as many URLs as you want.

5. Click **OK**.
6. Click **HTTP Preferences** in the left panel.
7. Click the New HTTP Preferences icon.
8. In the New HTTP Recipient dialog, enter an HTTP name, an optional description, HTTP preferences, and a connector. The HTTP preference is simply the location to which events are sent.
9. Click **Integration Preferences** in the left panel.
10. Click \* to add an integration preference, and then select Generic Event Integration as the Integration Preference type that you are adding.
11. In the Generic Event Integration Preferences dialog, enter the following:

- A Name for your event integration preference
- The HTTP connector preference (specified in step 8)
- The "cda receiver" tag (specified in step 4)

**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.

12. Click **OK**.
13. Click the CDA Template container in the left panel. Here you can apply the search filter tag to the search containers that you want to send events to HP CDA. When you navigate to the Search/Filter tags, select those you want to use. These tags are applied to HP SiteScope objects to enable filtering and search using your own key words.
14. Click the checkbox next to the My CDA event recipient tag to add the tag.
15. Click **Save**.

## Configuring HP Operations Manager to Send Events to HP CDA

Before configuring HP Operations Manager (HP OM) to send events to HP CDA, ensure that the following prerequisites are satisfied:

- HP OM (OMU) for Linux is installed.

**Note:** While HP CDA does not support OM Server on Windows, the OM Server for Linux is capable of monitoring Windows nodes because the HP OM agent works with either HP OM Server platform (Windows or Linux).

- Ensure that all required SPIs (smart plugins) are installed.

To configure HP OM to send events to HP CDA:

1. In HP OM, copy the HP CDA integration archive from your HP CDA system to a temporary location on your HP OM Server. This archive is located on your CDA server at:

```
<install dir>/integration/monitoring/om/om-svr-cda-integration-1.1.0-install.tar.gz.
```

2. Extract the archive contents:

```
gunzip om-svr-cda-integration-1.1.0-install.tar.gz
```

```
untar om-svr-cda-integration-1.1.0-install.tar
```

3. Modify the permissions on the installation scripts:

```
cd om-svr-cda-integration-1.1.0-install
```

```
chmod 774 *.sh
```

4. Place the following dependent jar files in the `/opt/lib` folder:

```
-commons-beanutils-1.6.1.jar  
-commons-logging-1.1.jar  
-commons-codec-1.3.jar  
-groovy-all-1.8.0.jar  
-httpclient-4.1.1.jar  
-commons-collections-3.2.jar  
-httpcore-4.1.jar  
-xercesImpl-2.7.1.jar  
-commons-lang-2.3.jar  
-http-builder-0.5.1.jar  
-json-lib-2.3-jdk15.jar  
-xml-resolver-1.2.jar
```

5. Run the installation script:

```
./install.sh addcda <CDA HOSTNAME> <CDA HTTP PORT>
```

For example, `./install.sh addcda mycda.server.domain.name 8080`.

**Note:** HP OM is agent-based, so when you deploy it you also deploy your agents. HP OM integration supports the following management packs (not to be confused with HP CDA plugins):

- Infrastructure SPI
- System Infrastructure SPI
- Oracle SPI
- Microsoft SPI
- Microsoft SQL Server SPI
- WebSphere SPI
- JBoss SPI
- WebLogic SPI

## ***Revising the HP Operations Manager Configuration***

To change the CDA host name or CDA port after it has already been registered:

1. If it is already running, stop the HP CDA process:

```
ovc -stop cda
```

2. Enter the new CDA HOSTNAME in the file `url.properties`, which is located in the directory `/opt/OV/cda/bin`.

3. Re-start the CDA process:

```
ovc -start cda
```

This will register CDA with OVC and run as a process named 'cda'.

## Uninstalling the HP CDA Integration

If you want to remove the HP CDA integration with HP OM, you must uninstall it.

To uninstall the HP CDA integration:

1. Run the uninstall script to remove the 'cda' process from 'ovc':

```
cd /opt/OV/cda/uninstall
./install.sh deletecda
```

2. Delete the integration files:

```
rm -r /opt/OV/cda
```

## Configuring Nagios to Integrate with HP CDA

Before you configure Nagios to integrate with HP CDA, be aware of the following requirements:

Requirement	Description
Nagios Server is installed on Linux OS	The Nagios server must be installed, configured and running before you can configure it to send events to HP CDA.
HTTP server is running	The HTTP server should be configured to run as user <i>nagios</i> and group as <i>nagios</i> .
Install plugins on Nagios server	Ensure that the <code>check_nrpe</code> and <code>check_nt</code> plugins have been installed on the Nagios server.

After the Nagios server has been installed, you must configure it to integrate with HP CDA:

1. Copy the files from CDA installed directory to the `tmp` folder on the Nagios server. Files are available in:

– (Linux users): `/opt/hp/cda/1.10/integration/monitoring/nagios`.

– (Windows users): `C:\CDA\1.10\integration\monitoring\nagios`.

From here, copy the `nagios-svr-cda-integration-<cda-version>-install.tar.gz` file to a `/tmp` directory:

```
tar -xzvf nagios-svr-cda-integration-<cda-version>-install.tar.gz -C /tmp
```

where `<cda-version>` is the HP CDA version that you have installed.

2. Edit the file `/tmp/install.cfg` by specifying values for the following parameters:

Parameter Name	Description
<code>cda_token</code>	Authentication for the HP CDA-Nagios integration.
<code>nagios_cmd</code>	The location of the <code>nagios.cmd</code> file. You can find this information in the <code>nagios.cfg</code> file under "command file".
<code>nag_checkresults</code>	The Nagios check results value is defined in the <code>nagios.cfg</code> file as <code>check_result_path</code> .
<code>nagios_tmp</code>	A Nagios temporary directory whose value is defined in the <code>nagios.cfg</code> file as <code>temp_path</code> .
<code>nagios_bin</code>	The location of the Nagios executable file.
<code>cda_server</code>	The HP CDA server fully qualified domain name (FQDN).
<code>nagios_config_file</code>	The location of the primary Nagios configuration file ( <code>nagios.cfg</code> ).
<code>plugins_dir</code>	The location of the Nagios default plugin directory.

3. As a root user, run the script `/tmp/install.sh` to configure the Nagios integration with HP CDA.
4. Add the following configuration definitions to the `nagios.cfg` file:

```
cfg_dir=/opt/cda/etc/config/
```

```
cfg_dir=/opt/cda/etc/hosts/
```

```
cfg_dir=/opt/cda/etc/services/
```

5. Restart Nagios and the HTTP servers.

**Note:** Monitoring with Nagios requires that an agent is installed on the provisioned servers. Ensure that the Nagios agents have been installed as part of the platform software.

**Note:** Ensure the Nagios HTTP server is configured with the alias `/nagios`.

**Note:** The Nagios NRPE agent contains some out-of-box plugins. Add additional plugins in the `plugins` directory with the appropriate configuration on the provisioned server.

## Managing Parameters

An important aspect of modeling platforms and applications using HP CDA is the ability to provision and deploy models into different environments; this is accomplished 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.

*Parameters* are named objects, created by a model designer/developer, to hold a value that can then be substituted into various points in a model via substitution variables. Parameters and parameterization allow developers to design flexibility into their models. A parameter's value can come from any of the following:

- A default value that is assigned to the parameter when it is 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 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.
- The value of another parameter.

After model developers determine the information that needs to be customizable for different environments, they can add parameters to the models and place substitution variables 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 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 at the beginning or during the provisioning or deployment operation, depending on the parameter type used.

You can customize parameter values either manually, or through the use of previously-saved stored parameter sets.

The following topics provide information about parameters.

Topic	Information Contained
<a href="#">"Adding Parameters " below</a>	How to add parameters to various models or portions of models (application model, application layer, platform models, monitors, etc.).
<a href="#">"Using Stored Parameter Sets " on page 354</a>	How to create, edit and use stored parameters for application deployment and platform provisioning.
<a href="#">"Using Substitution Variables " on page 356</a>	How to use substitution variables in work flows, fields, and inline strings.
<a href="#">"Adding Reference Variable Parameters" on page 359</a>	How to use multiple levels of parameter referencing, such that a parameter derives its value from another parameter.
<a href="#">"Using Previous Command Output Variables" on page 363</a>	How to use previous command output variables to temporarily hold results of components that are participating in software workflow or application layers.

## Adding Parameters

Model designers and application developers can add parameters to customize model and application components at several levels. You can add parameters at the domain or tenant level, to applications, to platform software, and so on.

As a best practice, you define a parameter at the lowest level where it is visible to all the children that must view it. For example:

- If a parameter must be visible to two different application layers in the same model, it should apply to the model.
- If a parameter must be visible to two different application layers in two different application models, it should be created in the application layer.

To add a parameter, complete the following steps:

1. Navigate to where you want to add a parameter.
2. Under the Parameter Definitions header, click **+ Add** to open the Add dialog box.

3. Make entry:

[Click here to see detailed input rules.](#)

Field	Description
Label	This label identifies the parameter—it is not the actual name of the parameter used in references and substitution variables. Your entry here automatically populates Name.
Name	<p>This field becomes the character-for-character string that is used in references and substitution variables. This field is automatically populated by Label entry, but you can change it before you click Save. After you save, you cannot change Name.</p> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p><b>Note:</b> You can use a workaround to change the name by deleting your original parameter and recreating it with a new name. If you rename a parameter in this way, you must change the parameter name in places where it is referenced, such as in substitution variables.</p> </div> <p>Parameters within the same parameter group cannot use the same name. However, parameter names are case-sensitive. You can create a parameter named <code>my.parameter</code> and another named <code>my.Parameter</code> in the same group.</p> <p>The duplicate name restriction does not apply to parameters that are created in different parameter groups.</p>
Description	An optional description for the parameter.
Required	If you want a parameter value to be required entry when the parameter value is called. 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	To mask the parameter value where it appears or is entered on interface windows.
Read Only	If you want the parameter value to be un-editable on interface windows. (The Add and Edit parameter dialog box will allow edits.)
Private Scope	To limit the parameter scope to within the area where it was created. For example, a parameter with a private scope might be listed on the Parameters tab of the Application Model window, but not on the Deploy Application wizard.

Parameter Type	<p>Following are the available parameter types:</p> <ul style="list-style-type: none"><li>■ <b>String:</b><ul style="list-style-type: none"><li>- <i>Minimal Length</i>: The string's minimum character length.</li><li>- <i>Maximum Length</i>: The string's maximum character length.</li><li>- <i>Allowed text</i>: Enter a text string here to establish a regular expression for the string. User input must match the regular expression.</li></ul><p>HP CDA supports a common subset of the Java and JavaScript regular expression syntax.</p><p>For information on Java regular expression syntax, search <a href="http://docs.oracle.com">http://docs.oracle.com</a>.</p><p>For information on JavaScript regular expressions, search <a href="http://www.ecma-international.org/publications">http://www.ecma-international.org/publications</a> .</p><p>If you leave this text box empty, it is the same as establishing the default “.” regular expression. In that case, users are allowed to enter any string.</p></li><li>■ <b>Number::</b><ul style="list-style-type: none"><li>- <i>Minimal Value</i>: The number's minimal value.</li><li>- <i>Maximum Value</i>: The number's maximum value.</li></ul></li><li>■ <b>List</b>: Click <b>Add Value</b> to add additional values that appear in or populate the List.  Choose the <b>Enable MultiSelect</b> option to enter more than one parameter value, separated by delimiters.  Accept a comma (,) as the <b>Delimiter Character</b> or type a different character in the text box.</li><li>■ <b>Email</b>: No additional input required.</li><li>■ <b>Boolean</b>: Use entry here to define the values for <i>True</i> and <i>False</i>.</li><li>■ <b>Key</b>: No additional input required.</li><li>■ <b>Directory</b>: Choose Directory to enforce that files are copied to</li></ul>
----------------	---

	predefined directories (using the <code>BASE_PATH</code> parameter). If an administrator has defined a base path for the Directory, users cannot edit the base path.
Default Value	<p>The default value of the parameter, which can be one of the following:</p> <ul style="list-style-type: none"> <li>■ <b>Blank:</b> When no default value is desired.</li> <li>■ <b>A textual value:</b> Type a textual value in the text box.</li> <li>■ <b>A reference to another parameter:</b> Click the ellipsis icon (...) that appears below the field, and select the appropriate reference variable from the Reference Variables dialog box.</li> </ul> <p>If you want to reference a parameter that has not yet been created, you can complete creating the parameter and edit it to include the reference later.</p>

4. Click **Add** to save the parameter and close the dialog box.

## Using Stored Parameter Sets

You can specify the values of user-exposed parameters either manually or through the use of a *stored parameter set*. A stored parameter set enables you to customize a platform or application for different environments, and provides a mechanism whereby you can save parameter values so that they can be recalled at a subsequent time; for example, when deploying to the same environment.

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).

When a provision or deploy operation is initiated via HP CDA, either the "Provision Platform" dialog box or the "Deploy Application" wizard will open. Within those interface elements, you can view user-exposed parameters that have been designed into the model.

The following topics provide information about using stored parameters.

Topic	Information Included
<a href="#">"Creating or Editing a Stored Parameter Set " on next page</a>	How to use the Stored Parameters tab to create or edit stored parameters for platform or application models.
<a href="#">"Saving Parameter Values to a Stored Parameter Set " on next page</a>	How to save parameters during provisioning or deployment, so they become stored parameters.
<a href="#">"Using a Stored Parameter Set During Provisioning or Deployment " on next page</a>	How to use stored parameters.

## ***Creating or Editing a Stored Parameter Set***

To create or edit a stored parameter set:

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 **New** above the list of existing parameter sets. 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 necessary, and then click **OK** to save the values and close the dialog box.

## ***Saving Parameter Values to a Stored Parameter Set***

To save parameter values to a stored parameter set:

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.
3. Click the **Save Parameters** button (the last screen of the “Deploy Application” wizard) in the dialog box or wizard.
4. In the **Enter stored parameters entry name** field, enter a name for the stored parameter set and then click **Save**.

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

5. You can either follow through with the provisioning or deployment operation by clicking **OK** or **Finish** as is applicable, or click **Cancel** to cancel the provisioning or deployment operation. Regardless, the stored parameter set has been saved.

## ***Using a Stored Parameter Set During Provisioning or Deployment***

To use a stored parameter set during provisioning or deployment:

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

## Using 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.

The following topics provide information about using substitution variables.

Topic	Information Included
<a href="#">"Using Substitution Variables in Workflow Actions " below</a>	How to use substitution variables in Executed Script and Placed File workflow actions.
<a href="#">"Using Substitution Variables in Fields " on next page</a>	List of interface screen fields where you can use substitution variables.
<a href="#">"Using Substitution Variables in Inline Strings " on next page</a>	How to use substitution variables in inline strings.
<a href="#">"Using Previous Command Output Variables" on page 363</a>	How to use transient variables in a software workflow or an application layer.

## Using Substitution Variables in Workflow Actions

You can use substitution variables in an Executed Script workflow action. When you create substitution variables, they are displayed in the Input Component Parameters portion of the

window. When you run the Executed Script workflow action, the substitution variables are replaced by parameter values.

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.

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

## ***Using Substitution Variables in Fields***

Substitution variables can also be used in certain fields of some of the interface screens. The following fields 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

## ***Using Substitution Variables in Inline Strings***

HP CDA currently supports the use of substitution variables in inline strings for application and platform software *workflow* components. You can include inline strings in executed scripts, placed files, and in fields that support the use of substitution variables. Inline strings can also be used as a default value in the **Value** field of parameters.

You cannot use substitution variables in inline strings at the following levels:

- domain
- application layer or model
- application or software level parameters

HP CDA Placed File workflow actions contain a **Content Parameterized** checkbox. If you select the checkbox, HP CDA searches for and replaces the substitution variables as needed.

If you do not select **Content Parameterized**, HP CDA will not perform the search-and-substitute operation. The checkbox is necessary because the search-and-substitute operation can take considerable processing time. Use the checkbox to prevent executing the operation when a placed file does not include substitution variables.

At the appropriate time, HP CDA automatically resolves substitution into a concatenated string. For example, in the following inline string, HP CDA replaces the substitution variables with the values of the named parameters:

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

In the following 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:

**jdbc:mysql://db\_srv-017:3306/CDA**

You can only add inline substitution variables for parameters that are local to that component. For example, if a deploy workflow has two Placed File components:

**Placed File 1 = baseConfigFile.dat** with parameters **P1**, **P2** and **P3**

**Placed File 2 = installScript.sh** with parameters **P1**, **P88** and **P59**

**Placed File 1** can use inline, parameters **@{P1}**, **@{P2}** and **@{P3}**

**Placed File 2** can use inline, parameters **@{P1}**, **@{P88}** and **@{P59}**

**Placed File 1** cannot use **@{P88}** or **@{P59}** inline.

**Placed File 2** cannot use **@{P2}** or **@{P3}** inline.

Note that **Placed File 1** parameter **@{P1}** is completely different from **Placed File 2** parameter **@{P1}**.

## Adding Reference Variable Parameters

In HP CDA, a parameter value can be derived by referencing other parameters to get their values via 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 references **parameter.three**.

HP CDA protects against circular referencing and does not allow it.

Parameter references occur when a parameter value is defined by specified reference variable. In HP CDA reference variables are identified by the following pattern or format:

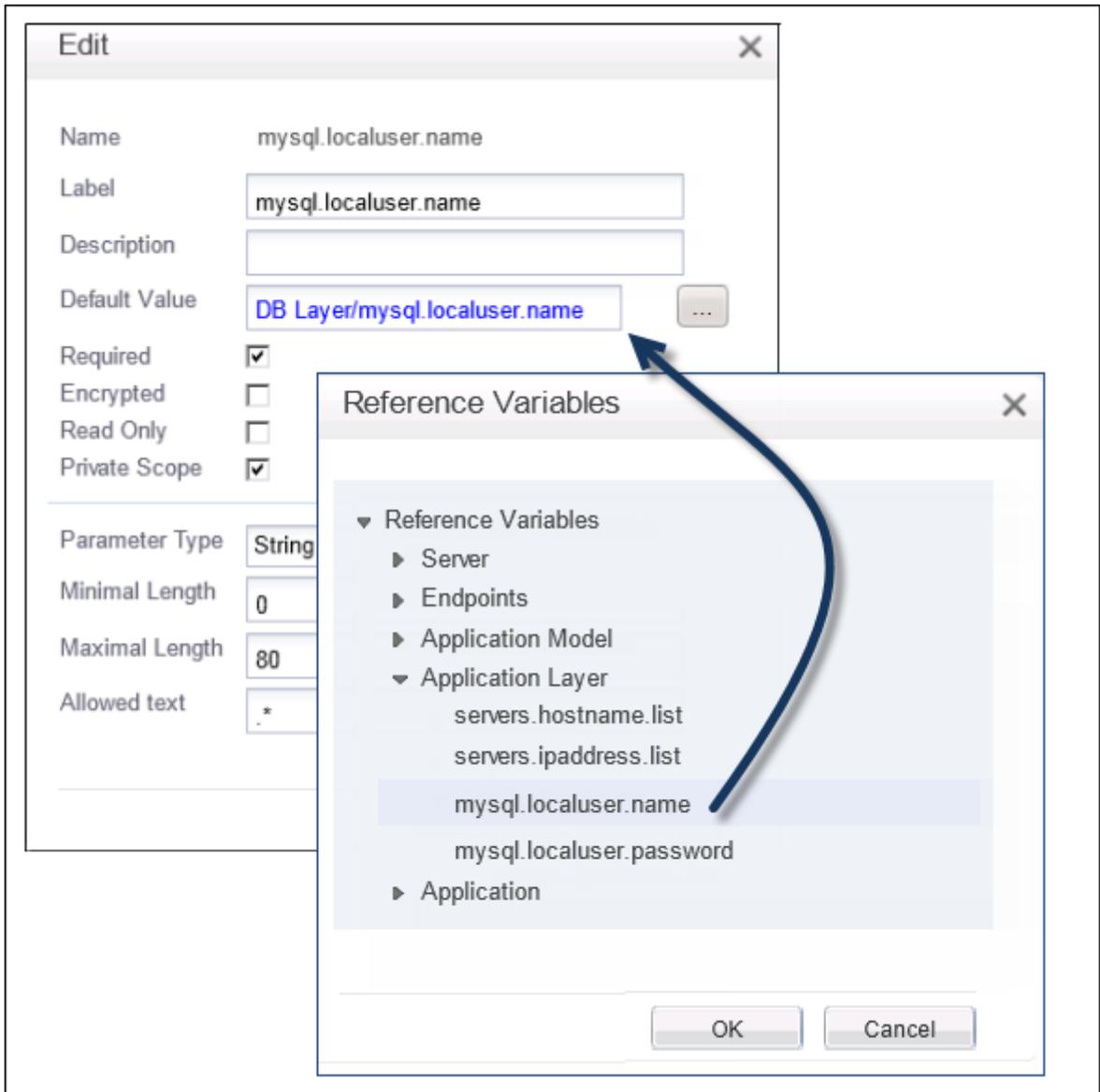
```
${reference-details}
```

Reference-*details* is syntax that is readable by HP CDA and used to locate the parameter that has been referenced.

The reference variable pattern and reference-*details* syntax combination can be complex and difficult to enter manually. However, you do not need to this data manually. Instead, you can select the required reference variable on the Reference Variable dialog box. Advantages of using a reference variable include:

- You do not need to know the reference-*details* syntax.
- You choose from parameters that *can* be referenced. You see choices are categorized, with easy-to-identify names.
- The Reference Variables dialog box only shows 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, Application Layer and Domain parameters. Only those reference variable types appear in the Reference Variables dialog box.

After you select a reference variable, you see the reference variable pattern or reference-*details* in the Default Value text box (as shown in blue below).

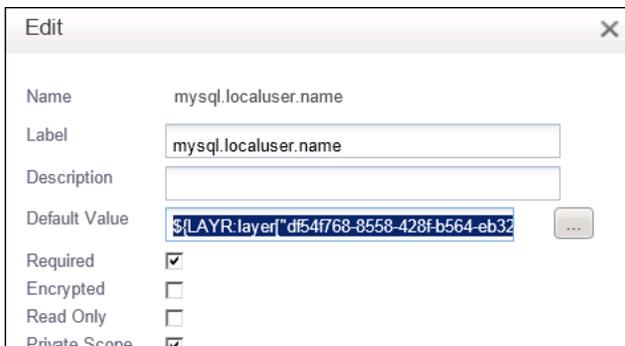


A reference variables default value is comprised of the names of objects in the model that "contain" the parameter or property being referred to and the name of the parameter or property itself. Each name is separated by the "/" character.

In this example, the default value appears as **DB Layer/mysql.localuser.name**. The first name, **DB Layer**, refers to the DB Layer application layer. The second name, **mysql.localuser.name**, refers to the **mysql.localuser.name** DB Layer parameter.



If you click in the Default Value text box when you are editing a reference parameter, the blue reference text converts to the internal reference syntax details that HP CDA uses to represent a reference variable.



If you click outside of the Default Value field, the text reverts back to the easier-to-read blue reference text.

**Caution:** If you modify the Default Value reference syntax details, the parameter is no longer a reference variable (and the text is no longer blue). The Default Value becomes a literal value. If you make a change inadvertently, you can revert back to a reference. Use the (...) button and reselect the appropriate item from the Reference Variables dialog box.

It is not critical that you remember the information included in the reference-details because it is used by HP CDA only. However, following `Ref->${}`, you see a four upper-case characters that are a reference type indicator. The characters tell you what kind of reference variable the parameter references:

- **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 is 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.

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

This reference variable type is for parameters that are shared across actions in a layer. When used, parameters are defined at the layer level and referenced by parameters at the layer workflow-action level using this reference variable type. If the default values of the referencing parameters subsequently must change, then that value only needs to be changed at the layer parameter level and all the referencing workflow action parameters will automatically use 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 to which a layer is applied. These reference parameters can also be used in Executed Script or Placed File workflow actions if the script or file needs that information.

If the list of hostnames or IP addresses is needed only by HP ALM, you can create a non-private, layer-level parameter that references either the `servers.hostname.list` or `servers.ipaddress.list` reference parameter. The HP ALM process can then 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.

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 can provide optional capabilities or have software requirements. If so, the platform software has capability or requirement software tags associated with it.

If a platform software model designer tags platform software this way, 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 you create the following tags:

— Tomcat 6, having a defined parameter “catalina\_home”

— Java 6, having the defined parameter “java\_home”

And if a designer creates a platform software model with the Tomcat 6 capability tag and Java 6 requirement tag, then platform software-level parameters and workflow parameters can 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.

The following reference variables types do not include a four-character reference type indicator:

- **Per Server**

Per Server 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 resolved by the system at runtime.

- **Monitoring**

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

## Using Previous Command Output Variables

*Previous command output variables* temporarily hold the result of a component that is participating in a software workflow or application layer. Similar to substitution variables, the values of previous command output variables are substituted before the next component is executed. The value of previous command output variables is set using the previous component result, which is command output, command error, and command exit code. Component results can only be passed to the *next immediate component*, and are *not* passed to all subsequent components. While the result (either output or error) of a command that is configured in a component can produce output beyond 200 characters in length, only the first 200 characters are used and stored in previous command output variables. Components that have been configured with executed scripts (which can execute a

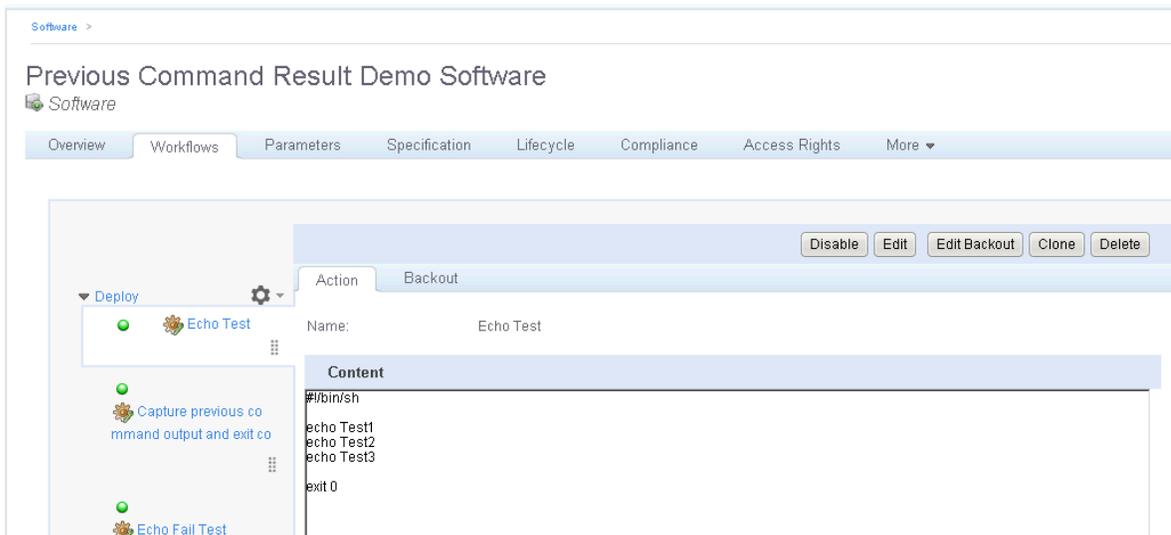
command) can pass the result to the next component if the next component is configured using previous command output variables and works with a deployer such as SSH, Chef, and SA. Also, external components configured with external deployers (such as Chef, SA, DMA, OO, and SSH External Component) that contain the actions or services or both to execute a command can pass the result to the next component that is configured using previous command output variables.

**Note:** When Chef is used to deploy platform software or applications to a Windows OS-based platform, the previous command output variable feature cannot be used, as it will not work properly.

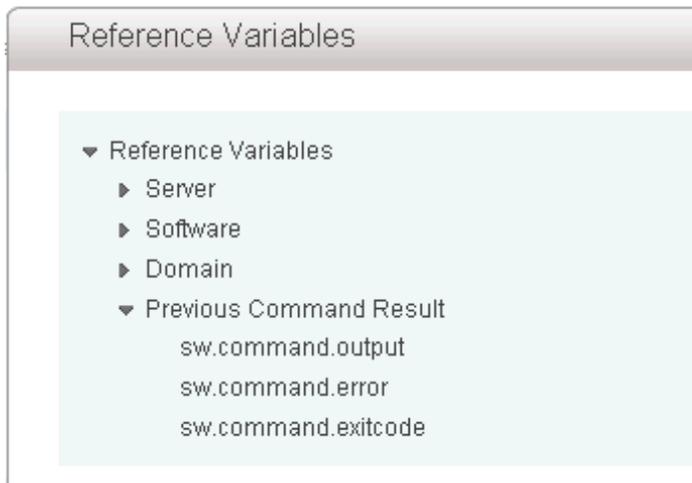
You do not use previous command output variables in the first component of platform software or an application layer workflow. Rather, you use them in the second component, because the first or previous component, which contains a command to be executed, stores the results into the variable that is used in the next component.

### Examples for Platform Software Containing a Workflow with Components Configured as Executed Scripts

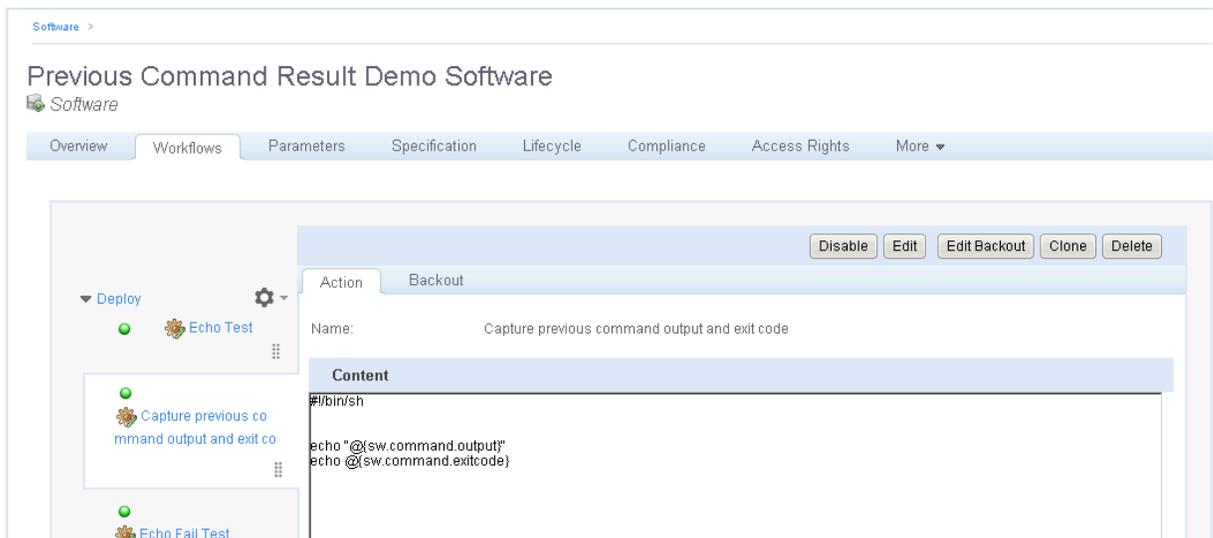
**Example 1, Capturing script output and return/exit code:** In this example, a platform software has a Deploy workflow containing two components named "Echo Test" and "Capture previous command output and exit code." The first component (Echo Test) contains a script that echoes three strings and outputs a return/exit code when run. This component is shown below:



The purpose of the second component (Capture previous command output and exit code) is, as the name implies, is to capture the output and return/exit code from the previous component, and echo them. In order to do this, it uses the `sw.command.output` and `sw.command.exitcode` Previous Command Result reference variables seen below:



These variables are shown included in the script in the following screen capture:



When the workflow runs and the second component executes, the following output will be shown:

Executing component Capture previous command output and exit code for server

Status: ✔ Successful  
 Start: July 30, 2013 10:45:38 AM  
 Entity: rheloamc.blr.adam.qa.hp.com  
 End: July 30, 2013 10:45:39 AM

**Parameters**

sw.command.output: `${sw.command.output}`  
 sw.command.exitcode: `${sw.command.exitcode}`

**Output**

```
Test1
Test2
Test3
0
```

**Example 2, Capturing script output, error and return/exit code:** The first component of this example contains a script with two commands. The first command is a valid `echo` command, and the second command is actually spelled incorrectly as `echoo` so it is an invalid command that will result in an error condition when the script runs. Thus, the component will fail if we don't use `exit 0` and if we don't use `exit 0`, we won't be able to capture the error and exit code. Use `exit 0`, therefore, if you need to capture the error. This component is shown below:

Software >

Previous Command Result Demo Software

Software

Overview Workflows Parameters Specification Lifecycle Compliance Access Rights More ▾

The screenshot shows a workflow editor with a component named "Echo Fail Test". The component's content is as follows:

```
#!/bin/sh
echo This will be stored in the previous command output variable
echoo This is a wrong command and should FAIL. This can be fetched from the previous command error variable
exit 0
```

The component is currently in a "Backout" state, as indicated by the "Backout" tab and the "Disable", "Edit", "Edit Backout", "Clone", and "Delete" buttons.

The second component captures the output, error, and return/exit code from the previous component, and echoes them. In order to do this, it uses the `sw.command.output`, `sw.command.error`, and `sw.command.exitcode` Previous Command Result reference variables as seen below:

The screenshot shows the configuration page for an action in the HP Continuous Delivery Automation interface. The action is named "Capture Previous command error and exit code". The content of the action is a shell script that echoes the previous command's output, error, and exit code. The input component parameters table is as follows:

Label	Default Value	Required	Encrypted	Read Only	Private Sc...
sw.command.error	Ref->\${sw.command.error}	No	No	No	Yes
sw.command.exitcode	Ref->\${sw.command.exitc...}	No	No	No	Yes

When the workflow runs and the second component executes, the following output will be shown:

Executing component Capture Previous command error and exit code for server

Status: ✔ Successful  
 Start: July 30, 2013 10:45:42 AM  
 Entity: rheloamc.blr.adam.qa.hp.com  
 End: July 30, 2013 10:45:43 AM

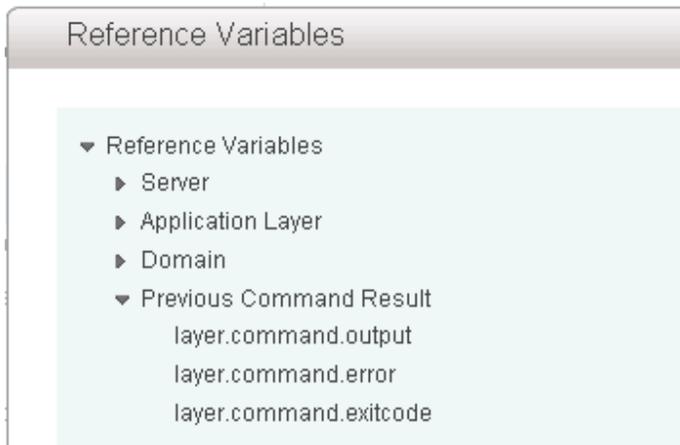
### Parameters

sw.command.error: \${sw.command.error}  
 sw.command.output: \${sw.command.output}  
 sw.command.exitcode: \${sw.command.exitcode}

### Output

```
The below variables captures the previous component result individually
This will be stored in the previous command output variable
./bl164cala_07e6_460d_bd4a_fc32e435f95d: line 4: echoo: command not found
0
```

Note that previous command output variables are available for platform software as described above, as well as for application layers (below):



## Canceling Provisioning or Deployment Operations

During a platform provisioning or application deployment, the need might arise for the operation to be canceled. 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.

The following topics provide information about canceling, backing out and cleanup during provisioning and deployment.

Topic	Information Included
<a href="#">"Canceling During Provisioning or Deployment " below</a>	How to cancel an in-process provisioning or deployment operation.
<a href="#">"Backing Out of Provisioning or Deployment Operations " on next page</a>	How to reverse and undo operations of a failed platform provisioning or application deployment sequence.
<a href="#">"Forced Cleanup of Failed Operations " on page 371</a>	Information about the HP CDA forced cleanup that you can use when backout or cancel operations fail to complete successfully.

## Canceling During Provisioning or Deployment

When a platform provisioning or application deployment operation commences, you can access a Report window that lists the steps underway. The Report window includes a Cancel button.

To cancel the provisioning or deployment operation, complete the following steps:

1. Click to access the report.
2. Click **Cancel**.
3. Click **Yes** in the warning that appears.

The cancellation begins, but it may be several minutes before you see related events in the Report screen.

## Backing Out of Provisioning or Deployment Operations

For a variety 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*).

Topic	Information Included
<a href="#">"System-Generated Backout Steps " below</a>	Information about automatic system-generated backout steps.
<a href="#">"User-Created Backout Steps " on the facing page</a>	Information about backout steps that are part of the Deploy workflow.
<a href="#">"Forced Backout " on the facing page</a>	Information about what happens when Forced Backout is enabled.
<a href="#">"Manual Backout " on page 371</a>	How to initiate a manual backout.

### 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.
- For application deployment – The **Deploy settings** screen of the **Deploy Application** wizard.

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 will appear in the report indicating that forced backout has automatically started.

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

**Note:** You can make additional parameter settings for the individual programming actions of a workflow to affect a forced backout. See the **Advanced Options** setting descriptions for the programming actions in "[HP CDA Programming Operations](#)" on page 372.

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

## 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.

If a manual backout operation fails, HP CDA includes an additional feature called *forced cleanup* that can be used.

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

## HP CDA Workflows

A workflow is a sequence of actions that are performed on an application or platform software. Following are your workflow choices:

- **Deploy:** Programming operations for Placed File, Executed Script, Service Command, and External Component. See Application, Application Version, & Application Model for descriptions of these operations and their parameters.
- **Custom:** Programming operation that executes a custom workflow from within another workflow. This is useful in situations where a similar sequence of programming operations needs to be run in different workflows.
- **Undeploy:** Programming operation that deletes an application database, stops deployment scripts, and restores configuration scripts.

**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.

- **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 workflow:** Custom (user-defined) programming operations actions, such as Pause, Restart, and Backup.
- **Add New:** Click to add components to the selected workflow, such as custom (user-defined) programming operations for Placed File, Placed Directory, Executed Script, Service Command, External Component, and Invoke Workflow.

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, SA Policy, or SA Script. In the Action tab, click Browse to select an external component.

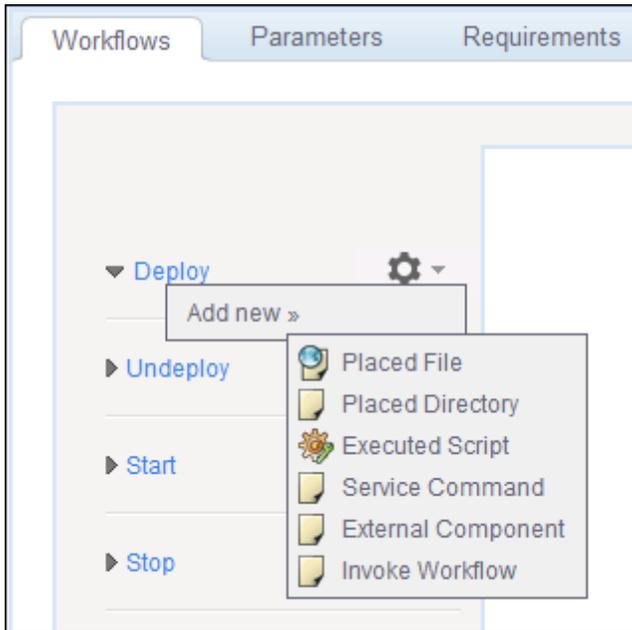
You can also create custom application workflows, such as Pause, Restart, and Backup. A custom application workflow is commonly known as a user-defined application workflow.

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

## HP CDA Programming Operations

HP CDA workflows can contain a sequence of one or more programming operations that run sequentially.

Programming operations can be added to workflows. The following graphic shows programming operations that you can add, in this case to the Deploy workflow.



The following topics provide information about programming operations.

Topic	Information Included
<a href="#">"Placed File Programming Operation " on the facing page</a>	How to copy a file to a target server.
<a href="#">"Placed Directory Programming Operation " on page 379</a>	How to include a complete directory structure on a target system.
<a href="#">"Executed Script Programming Operation " on page 380</a>	How to run a script on a target server.
<a href="#">"Service Command Programming Operation " on page 383</a>	How to run service operations as part of an application flow.
<a href="#">" External Component Programming Operation " on page 387</a>	How to run an automation sequence from an integrated service provider.
<a href="#">"Invoke Workflow Programming Operation " on page 388</a>	How to run another workflow from within a workflow.
<a href="#">"Enabling and Disabling Programming Operations " on page 389</a>	Information about the programming operation enable/disable feature. Use this feature to select the created programming operations that will run during workflow execution.

<p><a href="#">"Base Path Parameter " on page 391</a></p>	<p>Information about the parameter. Use Base Path to enforce that placed files are installed only within whitelisted directories in a specific domain.</p>
---	--

## Placed File Programming Operation

**Note:** If a placed file operation fails because the file is too large, you can use the executed script operation to copy the file directly from the source to the target machine. The executed script operation bypasses Chef, which causes the failure.

**Note:** Once a placed file operation has been created, you can choose to disable it so that it will not run when the workflow in which it is contained executes. This is helpful, for example, when troubleshooting workflows. Refer to ["Enabling and Disabling Programming Operations " on page 389](#) for more information.

Use the Placed File programming operation to copy a file to a target server.

To add a Placed File programming operation, complete the following steps:

1. Navigate to the appropriate workflow.
2. Click the gear symbol (  ), click **Add new**, and then choose **Placed File** on the menu. Make entry in the window on the right.
3. **Name:** used to identify the Placed File operation.
4. **Source:** select the appropriate source option. Your choices depend upon whether you are creating an application or platform software workflow.

Application workflow options	Platform Software workflow options
Use build content	
Download from external URL	Download from external URL
Use image from DSL: <ul style="list-style-type: none"> <li>■ Upload a file to DSL</li> <li>■ Use existing image</li> </ul>	Use image from DSL: <ul style="list-style-type: none"> <li>■ Upload a file to DSL</li> <li>■ Use existing image</li> </ul>

- **Use build content.**

Choose a file on the menu.

- **Download from external URL.**

In the text box, type the URL that references the file to download. For example:  
`ftp://1.2.3.4/HP/Diagnostics/Diagnostics-installer/instFile.zip`.

- **Use image from DSL:**

- Click **Upload** to upload a new image file. Make entry in the Attach Data Content dialog box:

- i. **Source:** Select a source option. If you select **File**, type or browse to enter the file path. If you select **URL**, type or paste in a URL.
- ii. **Server Folder:** Enter the HP CDA server location to place the uploaded file. You can click **Select** and drill down to select the location. The default is `/defaultDomain`.
- iii. Click **Upload**.

- Click **Change** to use an existing artifact. Make entry in the Select Artifacts dialog box, and click **Add**.

5. **Base Path:** If the base path has not been set by an Administrator, type or paste it in the text box. To use a reference variable, click the ellipses (...), and select the appropriate reference variable from the Reference Variables dialog box. Click **OK**.
6. **Destination:** Enter the path and the name of the file as it will be placed in the HP CDA server file system, for example `- /tmp/Diagnostics/installer/instFile.zip`.
7. **Full Install Path:** HP CDA joins Base Path and Destination to create a Full Install Path.
8. **Content Parameterized:** Select this option 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.
9. **Input Component Parameters:**

[Click here to see detailed input rules.](#)

Field	Description
Label	This label identifies the parameter—it is not the actual name of the parameter used in references and substitution variables. Your entry here automatically populates Name.
Name	<p>This field becomes the character-for-character string that is used in references and substitution variables. This field is automatically populated by Label entry, but you can change it before you click Save. After you save, you cannot change Name.</p> <div data-bbox="570 657 1369 869" style="background-color: #f0f0f0; padding: 10px;"> <p><b>Note:</b> You can use a workaround to change the name by deleting your original parameter and recreating it with a new name. If you rename a parameter in this way, you must change the parameter name in places where it is referenced, such as in substitution variables.</p> </div> <p>Parameters within the same parameter group cannot use the same name. However, parameter names are case-sensitive. You can create a parameter named <code>my.parameter</code> and another named <code>my.Parameter</code> in the same group.</p> <p>The duplicate name restriction does not apply to parameters that are created in different parameter groups.</p>
Description	An optional description for the parameter.
Required	If you want a parameter value to be required entry when the parameter value is called. 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	To mask the parameter value where it appears or is entered on interface windows.
Read Only	If you want the parameter value to be un-editable on interface windows. (The Add and Edit parameter dialog box will allow edits.)
Private Scope	To limit the parameter scope to within the area where it was created. For example, a parameter with a private scope might be listed on the Parameters tab of the Application Model window, but not on the Deploy Application wizard.

Parameter Type	<p>Following are the available parameter types:</p> <ul style="list-style-type: none"><li>■ <b>String:</b><ul style="list-style-type: none"><li>- <i>Minimal Length</i>: The string's minimum character length.</li><li>- <i>Maximum Length</i>: The string's maximum character length.</li><li>- <i>Allowed text</i>: Enter a text string here to establish a regular expression for the string. User input must match the regular expression.</li></ul><p>HP CDA supports a common subset of the Java and JavaScript regular expression syntax.</p><p>For information on Java regular expression syntax, search <a href="http://docs.oracle.com">http://docs.oracle.com</a>.</p><p>For information on JavaScript regular expressions, search <a href="http://www.ecma-international.org/publications">http://www.ecma-international.org/publications</a> .</p><p>If you leave this text box empty, it is the same as establishing the default “.” regular expression. In that case, users are allowed to enter any string.</p></li><li>■ <b>Number::</b><ul style="list-style-type: none"><li>- <i>Minimal Value</i>: The number's minimal value.</li><li>- <i>Maximum Value</i>: The number's maximum value.</li></ul></li><li>■ <b>List</b>: Click <b>Add Value</b> to add additional values that appear in or populate the List.  Choose the <b>Enable MultiSelect</b> option to enter more than one parameter value, separated by delimiters.  Accept a comma (,) as the <b>Delimiter Character</b> or type a different character in the text box.</li><li>■ <b>Email</b>: No additional input required.</li><li>■ <b>Boolean</b>: Use entry here to define the values for <i>True</i> and <i>False</i>.</li><li>■ <b>Key</b>: No additional input required.</li><li>■ <b>Directory</b>: Choose Directory to enforce that files are copied to</li></ul>
----------------	---

	predefined directories (using the <code>BASE_PATH</code> parameter). If an administrator has defined a base path for the Directory, users cannot edit the base path.
Default Value	<p>The default value of the parameter, which can be one of the following:</p> <ul style="list-style-type: none"> <li>■ <b>Blank:</b> When no default value is desired.</li> <li>■ <b>A textual value:</b> Type a textual value in the text box.</li> <li>■ <b>A reference to another parameter:</b> Click the ellipsis icon (...) that appears below the field, and select the appropriate reference variable from the Reference Variables dialog box.</li> </ul> <p>If you want to reference a parameter that has not yet been created, you can complete creating the parameter and edit it to include the reference later.</p>

10. **Details:** Use **Details** to configure specific permissions for the placed file. The following table provides entry guidelines:

Deployer	Target Platform	Details Section
Opscode Chef	Microsoft Windows	<b>Details</b> does not apply and should not be used.
Opscode Chef	Linux	Use <b>Details</b> to set Linux-style permissions
HP Server Automation	Microsoft Windows or Linux	Use <b>Details</b> to set permissions that are appropriate to the target server.

Enter **Details**:

**Group:** To which the placed file belongs.

**Mode:** User, group, and other permissions you want applied to the placed file. The input format is the same as traditional UNIX permission format using three character octal notation.

**Owner:** Placed file owner name.

**Replace:** Choose this option if you want the placed file to overwrite a file with the same name (if one exists) at the placed file destination. If not selected, an existing file will not be replaced.

**Content Encoding:** Character encoding of the content. Use this parameter to prevent HP CDA from corrupting the placed file content during parameter replacement.

For example, if you choose this option, 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.

11. **Advanced Options:** Select one of the following options:

[Click here to see detailed input rules.](#)

- **Ignore Failure:** To continue if the operation fails.

If you do not select this option and a failure occurs, the operation will be left in its failed state. (You can start the Backout action from the job report at a later time.)

- **Retry on Failure:** Select this option if you want the operation to retry if it fails.

Enter:

**Retry Count:** Number of times you want HP CDA to retry the operation.

**Time Between Retries:** Length of time and time units between retries.

**Note:** If all retry attempts fail, the **Force Automatic Backout and cleanup on any failure** option determines subsequent behavior. If selected, the Backout action begins automatically when all retries have failed.

## Placed Directory Programming Operation

A *placed directory* enables you to include a complete directory structure within the server groups in a deployment topology (as opposed to the "[Placed File Programming Operation](#)" on page 374, which only places a single file). To use the placed directory component, you must first define a software artifact that contains a directory structure from a software artifact provider (for example, the SVN Artifact Provider). Then, at deployment time, the file set defined by the software artifact (the entire directory structure) is deployed into the configured full install path. Essentially, placed directory components are used to simply copy directories from a variety of sources to a target provisioned instance during application deployment.

Additionally, you can create filters for the placed directory component to either include or exclude files that match the provided filter regular expression from the file set contained in the software artifact at deployment time. For example, if you apply the Everything filter, then all contents of the software artifact will be included at deploy time (in other words, no filter is applied).

You can add a placed directory component within the application layer workflow, or from a platform software workflow.

**Note:** Once a placed directory operation has been created, you can choose to disable it so that it will not run when the workflow in which it is contained executes. This is helpful, for example, when troubleshooting workflows. Refer to "[Enabling and Disabling Programming Operations](#) " on page 389 for more information.

## Executed Script Programming Operation

Use the Executed Script programming operation to run a script on a target server.

**Note:** Once an executed script programming operation has been created, you can choose to disable it so that it will not run when the workflow in which it is contained executes. This is helpful, for example, when troubleshooting workflows. Refer to "[Enabling and Disabling Programming Operations](#) " on page 389 for more information.

To add an Executed Script programming operation, complete the following steps:

1. Navigate to the appropriate workflow.
2. Click the gear symbol (  ), click **Add new**, and then choose **Executed Script** on the menu. Make entry in the window on the right.
3. **Name:** used to identify the Executed Script operation.
4. **Path:** path on the target system to use when searching for commands in scripts
5. **Precondition:** a command that will test that some important condition is met. For example, if you need to ensure that `/tmp/myfile` exists before executing the script, you can use:
  - `test -f /tmp/myfile` on Unix
  - `Test-Path C:\someDir` on Windows
6. **User:** name of the user who runs the script as on the target system
7. **Working Directory:** the current working directory to run the command from on the target system

**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.

8. **Content:** Type or paste the script to be executed.

## 9. Input Component Parameters:

[Click here to see detailed input rules.](#)

Field	Description
Label	This label identifies the parameter—it is not the actual name of the parameter used in references and substitution variables. Your entry here automatically populates Name.
Name	<p>This field becomes the character-for-character string that is used in references and substitution variables. This field is automatically populated by Label entry, but you can change it before you click Save. After you save, you cannot change Name.</p> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p><b>Note:</b> You can use a workaround to change the name by deleting your original parameter and recreating it with a new name. If you rename a parameter in this way, you must change the parameter name in places where it is referenced, such as in substitution variables.</p> </div> <p>Parameters within the same parameter group cannot use the same name. However, parameter names are case-sensitive. You can create a parameter named <code>my.parameter</code> and another named <code>my.Parameter</code> in the same group.</p> <p>The duplicate name restriction does not apply to parameters that are created in different parameter groups.</p>
Description	An optional description for the parameter.
Required	If you want a parameter value to be required entry when the parameter value is called. 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	To mask the parameter value where it appears or is entered on interface windows.
Read Only	If you want the parameter value to be un-editable on interface windows. (The Add and Edit parameter dialog box will allow edits.)
Private Scope	To limit the parameter scope to within the area where it was created. For example, a parameter with a private scope might be listed on the Parameters tab of the Application Model window, but not on the Deploy Application wizard.

Parameter Type	<p>Following are the available parameter types:</p> <ul style="list-style-type: none"><li>■ <b>String:</b><ul style="list-style-type: none"><li>- <i>Minimal Length</i>: The string's minimum character length.</li><li>- <i>Maximum Length</i>: The string's maximum character length.</li><li>- <i>Allowed text</i>: Enter a text string here to establish a regular expression for the string. User input must match the regular expression.</li></ul><p>HP CDA supports a common subset of the Java and JavaScript regular expression syntax.</p><p>For information on Java regular expression syntax, search <a href="http://docs.oracle.com">http://docs.oracle.com</a>.</p><p>For information on JavaScript regular expressions, search <a href="http://www.ecma-international.org/publications">http://www.ecma-international.org/publications</a> .</p><p>If you leave this text box empty, it is the same as establishing the default “.*” regular expression. In that case, users are allowed to enter any string.</p></li><li>■ <b>Number::</b><ul style="list-style-type: none"><li>- <i>Minimal Value</i>: The number's minimal value.</li><li>- <i>Maximum Value</i>: The number's maximum value.</li></ul></li><li>■ <b>List</b>: Click <b>Add Value</b> to add additional values that appear in or populate the List.  Choose the <b>Enable MultiSelect</b> option to enter more than one parameter value, separated by delimiters.  Accept a comma (,) as the <b>Delimiter Character</b> or type a different character in the text box.</li><li>■ <b>Email</b>: No additional input required.</li><li>■ <b>Boolean</b>: Use entry here to define the values for <i>True</i> and <i>False</i>.</li><li>■ <b>Key</b>: No additional input required.</li><li>■ <b>Directory</b>: Choose Directory to enforce that files are copied to</li></ul>
----------------	--

	predefined directories (using the <code>BASE_PATH</code> parameter). If an administrator has defined a base path for the Directory, users cannot edit the base path.
Default Value	<p>The default value of the parameter, which can be one of the following:</p> <ul style="list-style-type: none"><li>■ <b>Blank:</b> When no default value is desired.</li><li>■ <b>A textual value:</b> Type a textual value in the text box.</li><li>■ <b>A reference to another parameter:</b> Click the ellipsis icon (...) that appears below the field, and select the appropriate reference variable from the Reference Variables dialog box.</li></ul> <p>If you want to reference a parameter that has not yet been created, you can complete creating the parameter and edit it to include the reference later.</p>

10. **Environment Variables:**

Enter the name of an environment variable in the left text box, and enter the value of the environment variable in the right text box, and click **Add**. Click **Add** again to enter another Environment Variable.

11. **Advanced Options:** Select one of the following:

- **Ignore Failure:** Click here for details.

Select this checkbox to continue the deployment if the operation fails to deploy correctly. If this checkbox is not selected and a failure occurs, the deployment will be left in its failed state, and you can start the Backout operation from the job report at a later time.

- **Retry on Failure:** Click here for details.

Select this checkbox if you want the operation to retry if it fails.

Enter:

**Retry Count:** number of times that the operation is to be retried

**Time Between Retries:** length of time between retries and units of time

If **Retry on Failure** is selected and all retry attempts fail, subsequent deployment behavior depends on whether **Force Automatic Backout and cleanup on any failure** checkbox is selected. If selected, the Backout operation begins automatically when all retries have failed.

## Service Command Programming Operation

Use the Service Command programming operation to run service operations as part of a workflow.

For example, if an application requires configuring a database that is installed as a service on the target system, you can use Service Command components to stop the database prior to configuration and then restart it when configuration is done. This provides an expedient alternative to writing executed script components to accomplish the same stop and start operations.

**Note:** Once a service command programming operation has been created, you can choose to disable it so that it will not run when the workflow in which it is contained executes. This is helpful, for example, when troubleshooting workflows. Refer to "[Enabling and Disabling Programming Operations](#) " on page 389 for more information.

To add a Service Command programming operation, complete the following steps:

1. Navigate to the appropriate workflow.
2. Click the gear symbol (  ), click **Add new**, and then choose **Service Command** on the menu. Make entry in the window on the right.
3. **Name:** The name of the HP CDA workflow component.
4. **Service Name:** The name of the service that runs on the client.
5. Choose an **Action** on the menu:
  - **Restart:** Stop and then start the service.
  - **Start:** Start the service.
  - **Stop:** Stop the service.
6. Choose a **Startup Type** on the menu:
  - **Disable at Startup:** Configure the service to *not* start when the client operating system starts.
  - **Don't change:** Leave the current service startup configuration as-is.
  - **Enable at Startup:** Configure the service to automatically start when the client operating system starts.
7. **Path:** The full path to the service executable to be used if **Service Name** does not match the service that is configured on the client.
8. **Pattern.** A regular expression that certain deployers can use to find running instances of a service.
9. Enter **Input Component Parameters:**

[Click here to see detailed input rules.](#)

Field	Description
Label	This label identifies the parameter—it is not the actual name of the parameter used in references and substitution variables. Your entry here automatically populates Name.
Name	<p>This field becomes the character-for-character string that is used in references and substitution variables. This field is automatically populated by Label entry, but you can change it before you click Save. After you save, you cannot change Name.</p> <div data-bbox="570 657 1370 869" style="background-color: #f0f0f0; padding: 10px;"> <p><b>Note:</b> You can use a workaround to change the name by deleting your original parameter and recreating it with a new name. If you rename a parameter in this way, you must change the parameter name in places where it is referenced, such as in substitution variables.</p> </div> <p>Parameters within the same parameter group cannot use the same name. However, parameter names are case-sensitive. You can create a parameter named <code>my.parameter</code> and another named <code>my.Parameter</code> in the same group.</p> <p>The duplicate name restriction does not apply to parameters that are created in different parameter groups.</p>
Description	An optional description for the parameter.
Required	If you want a parameter value to be required entry when the parameter value is called. 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	To mask the parameter value where it appears or is entered on interface windows.
Read Only	If you want the parameter value to be un-editable on interface windows. (The Add and Edit parameter dialog box will allow edits.)
Private Scope	To limit the parameter scope to within the area where it was created. For example, a parameter with a private scope might be listed on the Parameters tab of the Application Model window, but not on the Deploy Application wizard.

Parameter Type	<p>Following are the available parameter types:</p> <ul style="list-style-type: none"><li>■ <b>String:</b><ul style="list-style-type: none"><li>- <i>Minimal Length</i>: The string's minimum character length.</li><li>- <i>Maximum Length</i>: The string's maximum character length.</li><li>- <i>Allowed text</i>: Enter a text string here to establish a regular expression for the string. User input must match the regular expression.</li></ul><p>HP CDA supports a common subset of the Java and JavaScript regular expression syntax.</p><p>For information on Java regular expression syntax, search <a href="http://docs.oracle.com">http://docs.oracle.com</a>.</p><p>For information on JavaScript regular expressions, search <a href="http://www.ecma-international.org/publications">http://www.ecma-international.org/publications</a> .</p><p>If you leave this text box empty, it is the same as establishing the default “.*” regular expression. In that case, users are allowed to enter any string.</p></li><li>■ <b>Number::</b><ul style="list-style-type: none"><li>- <i>Minimal Value</i>: The number's minimal value.</li><li>- <i>Maximum Value</i>: The number's maximum value.</li></ul></li><li>■ <b>List</b>: Click <b>Add Value</b> to add additional values that appear in or populate the List.  Choose the <b>Enable MultiSelect</b> option to enter more than one parameter value, separated by delimiters.  Accept a comma (,) as the <b>Delimiter Character</b> or type a different character in the text box.</li><li>■ <b>Email</b>: No additional input required.</li><li>■ <b>Boolean</b>: Use entry here to define the values for <i>True</i> and <i>False</i>.</li><li>■ <b>Key</b>: No additional input required.</li><li>■ <b>Directory</b>: Choose Directory to enforce that files are copied to</li></ul>
----------------	--

	predefined directories (using the <code>BASE_PATH</code> parameter). If an administrator has defined a base path for the Directory, users cannot edit the base path.
Default Value	<p>The default value of the parameter, which can be one of the following:</p> <ul style="list-style-type: none"><li>■ <b>Blank:</b> When no default value is desired.</li><li>■ <b>A textual value:</b> Type a textual value in the text box.</li><li>■ <b>A reference to another parameter:</b> Click the ellipsis icon (...) that appears below the field, and select the appropriate reference variable from the Reference Variables dialog box.</li></ul> <p>If you want to reference a parameter that has not yet been created, you can complete creating the parameter and edit it to include the reference later.</p>

10. *Optional:* Enter **Advanced Options:**

Click here to see detailed input rules.

- **Ignore Failure:** To continue if the operation fails.

If you do not select this option and a failure occurs, the operation will be left in its failed state. (You can start the Backout action from the job report at a later time.)

- **Retry on Failure:** Select this option if you want the operation to retry if it fails.

Enter:

**Retry Count:** Number of times you want HP CDA to retry the operation.

**Time Between Retries:** Length of time and time units between retries.

**Note:** If all retry attempts fail, the **Force Automatic Backout and cleanup on any failure** option determines subsequent behavior. If selected, the Backout action begins automatically when all retries have failed.

11. Click **Save**.

## External Component Programming Operation

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

- HP Operations Orchestration (HP OO) flows
- HP Server Automation (HP SA) packages, policies and scripts
- HP Database and Middleware Automation (HP DMA) flows
- Chef recipes

Note: If you create a workflow that contains an HP DMA or HP OO External Component programming operation, you will not see input parameters listed in the dialog box in which the external component is created even if the external component being referenced contains such parameters. When you deploy the application model that contains the programming operation, however, the input parameters will appear in the deployment wizard.

Note: When you create an External Component programming operation which references an HP SA script, a "Script Parameters" entry field will appear in the dialog box. If the script contains more than one input parameter, then enter the values into this field in the order in which they are accepted into the script separated by spaces. If any parameters contain spaces, then enclose them in quotes. This example shows how the parameters "one," "123," and "echo 123" would be entered into the "Script Parameters" entry field: `one 123 "echo 123"`.

Note: Once an external component programming operation has been created, you can choose to disable it so that it will not run when the workflow in which it is contained executes. This is helpful, for example, when troubleshooting workflows. Refer to "[Enabling and Disabling Programming Operations](#)" on next page for more information.

## Invoke Workflow Programming Operation

Use the Invoke Workflow programming operation to execute a workflow (referred to here as a *reusable workflow*) from within another workflow (referred to as a *parent workflow*).

This action is useful in situations where, for example, a similar sequence of programming operations needs to be run in different workflows. Rather than include the same sequence of programming operations in each workflow, you can do the following:

- Create a custom (reusable) workflow that contains the programming operations to be run from each parent workflow.
- Add an Invoke Workflow programming operation to each parent workflow at the point in each workflow where the programming operations in the reusable workflow need to run.

During execution, the parent workflow will branch to the reusable workflow when the Invoke Workflow programming operation runs, and then control will return to the parent workflow when the reusable workflow finishes running.

**Note:** Once an invoke workflow programming operation has been created, you can choose to disable it so that it will not run when the parent workflow in which it is contained executes. This is helpful, for example, when troubleshooting workflows. Refer to "[Enabling and Disabling Programming Operations](#)" below for more information.

To add an Invoke Workflow programming operation:

1. Navigate to the appropriate parent workflow.
2. Click the gear symbol (  ), click **Add new**, and then choose **Invoke Workflow** on the menu.
3. Enter a name for the workflow into the "Name" field.
4. Click the ellipsis icon (...) that is adjacent to the "Workflow Reference" field, select the name of the reusable workflow in the Reference Variables dialog box that appears, and then click **OK**.

If the reusable workflow has parameters defined within it, a Component Parameters section will appear in the dialog box when you click **OK**. That section will list the parameters that are currently defined for the reusable workflow, along with their current values.

5. Edit the component parameter values if necessary.

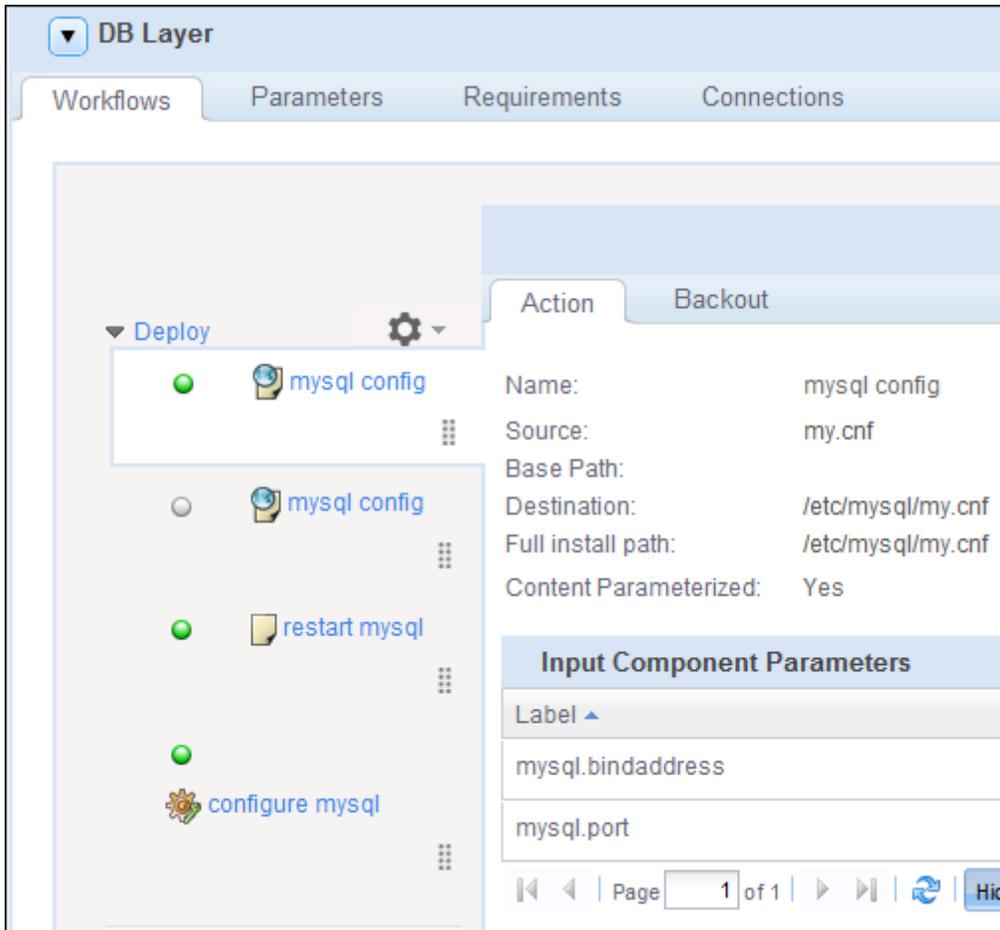
Any modifications to parameters that you make in this step will be passed to the reusable workflow when the workflow runs during deployment.

6. Click **OK** to save the Invoke Workflow programming operation.

## Enabling and Disabling Programming Operations

You can use the enable/disable programming operation feature to select programming operations that will run when the workflow is executed. This is useful when debugging a workflow. For example, certain programming operations can be temporarily disabled to check their effect on workflow execution, and then subsequently be re-enabled.

Consider a deployment workflow for the database layer of an application that configures the MySQL database, like shown in the following graphic.



Note that the "Deploy" workflow in the graphic contains two "mysql config" programming operations:

- The first "mysql config" programming operation in the flow is preceded by a green ball icon, indicating that it is enabled
- The second "mysql config" programming operation in the flow is preceded by a gray ball icon, indicating that it is disabled.

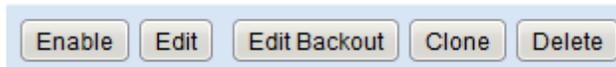
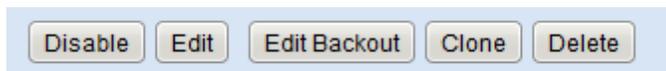
The user might use such a workflow setup to deploy an application with two different MySQL database configurations, to test which results in a more robust database.

To enable or disable a programming operation, complete the following steps:

1. If not already displayed, access the application workflow containing the programming operation to be enabled or disabled. See ["Accessing an Application Workflow "](#) on page 28
2. Click on the desired programming operation within the workflow to open it.

**Note:** Two methods can be used to enable or disable a programming operation - either by clicking the ball icon, or by clicking the **Enable/Disable** button. The programming operation does not need to be open to enable or disable it using the ball icon, but it must be open in order to enable or disable it using the **Enable/Disable** button.

3. If the programming operation is currently enabled, the ball icon will be green and the leftmost button in the row of buttons near the top of the screen will be a **Disable** button, like the first graphic that follows. If the programming operation is currently disabled, the ball icon will be gray and the leftmost button in the row of buttons will be an **Enable** button, like the second graphic that follows.



4. Either click the ball icon or the button to enable or disable the programming operation, as is applicable.

**Note:** Programming operations that have been disabled will be shown as skipped in the deployment report when the application containing them is deployed..

## Base Path Parameter

You can use the *Base Path* parameter to enforce that the installation destination of placed files occurs only within whitelisted directories in a specific domain. The domain-level parameter `Enforce Base Path` determines whether or not users must select from predefined directory parameters, or simply type in a directory path in the base path field. If an administrator has enabled the `Enforce Base Path` option, then the base path field cannot be manually entered and users must select the base path from a predefined set of directory parameter variables.

The specific parameters to which the Base Path parameter can refer depends upon whether or not the placed file component is part of the Application layer or Platform Software workflow.

Platform software component base path references include software and domain directory parameters (apart from tag parameters, which are part of the platform software requirements and capabilities).

The application layer workflow Base Path can reference the Domain/Application/Model/Layer apart from the Tag parameters, which are part of the layer's requirements.

**Note:** The parameters should be of type Directory; this must be enforced throughout the reference chain.

## 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.

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

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 receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

