



Hewlett Packard
Enterprise

Codar

Software version: 1.80

For Microsoft Windows® and Linux operating systems

Capsule Guide

Document release date: January 2017

Software release date: January 2017

Legal notices

Warranty

The only warranties for Hewlett Packard Enterprise 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. Hewlett Packard Enterprise 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 Hewlett Packard Enterprise 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 2017 Hewlett Packard Enterprise Development LP

Trademark notices

Adobe® is a trademark of Adobe Systems Incorporated.

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

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

UNIX® is a registered trademark of The Open Group.

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

The OpenStack word mark and the Square O Design, together or apart, are trademarks or registered trademarks of OpenStack Foundation in the United States and other countries, and are used with the OpenStack Foundation's permission.

Documentation updates

To check for recent updates or to verify that you are using the most recent edition of a document, go to the following URL and sign-in or register:
<https://softwaresupport.hpe.com>.

Select Manuals from the Dashboard menu to view all available documentation. Use the search and filter functions to find documentation, whitepapers, and other information sources.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your Hewlett Packard Enterprise sales representative for details.

Support

Visit the Hewlett Packard Enterprise Software Support Online web site at <https://softwaresupport.hpe.com>.

Contents

Legal notices	2
Introduction	4
PetClinic Application on AWS.....	4
PetClinic Application with a load balancer	7
PetClinic Application on existing servers	10
PetClinic Application on vCenter	12
Redeploy content creation.....	16
PetClinic Application.....	17
Existing-Infra	21
Auto-Generate.....	22
Script-Action	22
Send documentation feedback	23

Introduction

This section describes the application designs and the components that **are** shipped with HPE Codar 1.80. The intended audience of this document are administrators of Codar who create and configure application designs and components.

PetClinic Application on AWS

The CODAR_BP_AWS_PETCLINIC_APPLICATION_v1.80.00 application design is installed if you select the Install Sample Content option while installing Codar.

Application design name: CODAR_BP_AWS_PETCLINIC_APPLICATION_v1.80.00.zip

Prerequisites

To use this design, you have to host the following files on a Web Server (Apache Server or a Maven repository), that can be accessed using HTTP.

To host the application design:

1. `install_mysql.sh` - download this from wherever it is hosted internally.
2. `mysqldb_conf.sh` - download this from wherever it is hosted internally.
3. `install_tomcat.sh` - download this from wherever it is hosted internally.
4. `petclinic_jdbc_conf.sh` - download this from wherever it is hosted internally.

Note: The files above are located at `<CSA_HOME>\CSAKit-4.8\ContentArchives\topology\vmware_vcenter\petclinic\scripts`

5. `petclinic.war` - download this from Jenkins or wherever it is hosted internally.
6. `mysql-server_5.6.21-1ubuntu12.04_amd64.deb-bundle.tar` - download this from <http://ftp.kaist.ac.kr/mysql/Downloads/MySQL-5.6/> or wherever it is hosted internally.
7. `apache-tomcat-7.0.56.tar.gz` - download this from <http://tomcat.apache.org/download-70.cgi> or wherever it is hosted internally.

You must have an AWS template, which has the following tools installed:

- Ubuntu 12.04 or later
- JDK 1.7
- libaio1
- unzip
- zip
- Port 8080 should be available (if it is in use, try disabling IPv6)

Application design components

The imported application design has 6 components: Amazon Server (number of servers: 2), MySQL Database, Tomcat Application Server, PetClinicDB Conf, and PetClinic Application.

There is a dbDetails relationship created between 'PetClinicDB' and 'PetClinic Application'.

The PetClinic application that you deploy through this design performs the following steps:

1. Creates Amazon servers.
2. Deploys MySQL and Tomcat components on the Amazon server.
3. Configures PetClinicDB Conf on MySQL.
4. Deploys the PetClinic application on the Tomcat application server.

Application design customization

Amazon Server

Property	Description
availabilityZone	Enter the availabilityZone
Amid	Enter the amid (to be obtained from admin)
instanceNamePrefix	Enter the instanceNamePrefix
keyName	Enter the keyName (for example, Codar)
subnetId	Enter the subnetId (to be obtained from admin)
instanceType	Enter the instanceType (for example: small, medium)
securityGroupIds	Enter the securityGroupIds (to be obtained from admin)
username	Enter the username
password	Enter the password
privateKey	Enter the privateKey

MySQL

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where MySQL database. That is where <code>mysql-server_5.6.21-1ubuntu12.04_amd64.deb-bundle.tar</code> is located
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>install_mysql.sh</code> is located
installPath	Enter installPath (can be blank)
privatekeyPath	Enter privatekeyPath. For example, <code>C:\Users\Administrator\Downloads\CODAR.pem</code>
remoteFilePath	Enter the path on the server. For example it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh install_mysql.sh</code>
sshPort	SSH port. It can be 22.
privateKey	Enter the privateKey

Tomcat Application Server

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where apache-tomcat-7 is located. That is where <code>apache-tomcat-7.0.56.tar.gz</code> is located.
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>install_tomcat.sh</code> is located
installPath	Enter the tomcat install path including the tomcat home directory name. For example, <code>/opt/tomcat7</code>
privatekeyPath	Enter privatekeyPath. For example, <code>C:\Users\Administrator\Downloads\CODAR.pem</code>
remoteFilePath	Enter the path on the server. For example, it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh install_tomcat.sh</code>
sshPort	SSH port. It can be 22.

PetClinic DB Conf

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where apache-tomcat-7 is located. That is where <code>apache-tomcat-7.0.56.tar.gz</code> is located.
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>mysqladb_conf.sh</code> is located
mysqlpassword	Enter the password for the MySQL database
mysqlusername	Enter the user name for the MySQL database
remoteFilePath	Enter the path on the server. For example, it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh /tmp/mysqladb_conf.sh</code>
Port	It can be 22.
privatekeyPath	Enter privatekeyPath. For example, <code>C:\Users\Administrator\Downloads\CODAR.pem</code>

PetClinic Application

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where apache-tomcat-7 is located. That is where <code>apache-tomcat-7.0.56.tar.gz</code> is located.
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>mysqldb_conf.sh</code> is located
localfilepath	Enter the file name in local machine. It can be <code>petclinic.war</code>
remoteFilePath	Enter the path on the server. For example, it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh /tmp/mysqldb_conf.sh</code>
Port	It can be 22.
privatekeyPath	Enter privatekeyPath. For example, <code>C:\Users\Administrator\Downloads\CODAR.pem</code>

Note: The AWS template must not have MySQL and Tomcat installed. If they are already installed, then the design fails because the port will be in use. The content is validated only in IPv4.

PetClinic Application with a load balancer

The `CODAR_BP_PETCLINIC_APPLICATION_LOAD_BALANCER_v1.80.00` application design is installed if you select the Install Sample content option while installing Codar.

This design demonstrates the topology scaling that has a load balancer with the PetClinic application on vCenter servers. This application has a DB component installed on MySQL and an application component installed on the Tomcat server.

Service design name

`CODAR_BP_PETCLINIC_APPLICATION_LOAD_BALANCER_v1.80.00.zip`

Prerequisites

To use this design, you have to host the following files on a Web Server (Apache Server or a Maven repository), that can be accessed using HTTP.

To host the files:

1. `install_mysql.sh` - download this from wherever it is hosted internally.
2. `mysqldb_conf.sh` - download this from wherever it is hosted internally.
3. `install_tomcat.sh` - download this from wherever it is hosted internally.

4. `petclinic_jdbc_conf.sh` - download this from wherever it is hosted internally.
5. `loadbalancer_conf.sh` - download this from wherever it is hosted internally

Note: The files above are located at `<CSA_HOME>\CSAKit-4.8\ContentArchives\topology\vmware_vcenter\petclinic\scripts`

6. `petclinic.war` - download this from Jenkins or wherever it is hosted internally.
7. `mysql-server_5.6.21-1ubuntu12.04_amd64.deb-bundle.tar` - download this from <http://ftp.kaist.ac.kr/mysql/Downloads/MySQL-5.6/> or wherever it is hosted internally.
8. `apache-tomcat-7.0.56.tar.gz` - download this from <http://tomcat.apache.org/download-70.cgi> or wherever it is hosted internally.
9. `httpd-2.4.12.tar.gz`, `pcres-8.35.tar.gz` and `tomcat-connectors-1.2.40-src.tar.gz` must be saved in the `lb_artifact` folder. Bundle it as a `.tar.gz` file (for example, `lb_artifact.tar.gz`) and this bundled file is internally hosted.

`httpd-2.4.12.tar.gz` - This can be downloaded from <https://archive.apache.org/dist/httpd/>.

`pcres-8.35.tar.gz` - This can be downloaded from <http://sourceforge.net/projects/pcres/files/pcres/8.35/>.

`tomcat-connectors-1.2.40-src.tar.gz` - This can be downloaded from <https://archive.apache.org/dist/tomcat/tomcat-connectors/jk/>

Note: The downloaded `<httpd-2.4.12.tar.gz>` file must have the `apr` and `apr-util` folders under the `srclib` folder.

If these folders are missing, it can be downloaded from <https://apr.apache.org/> and must be bundled inside the `srclib` folder of `httpd-2.4.12.tar.gz`.

To use this partial design, you must fill in the following microservice designs:

- `CODAR_BP_MYSQL_MICROSERVICE_v1.80.00.zip`
- `CODAR_BP_TOMCAT_STACK_MICROSERVICE_v1.80.00.zip`

Create vCenter template which has the following tools installed:

- Ubuntu 12.04 or later
- JDK 1.7
- `libaio1`
- `unzip`
- `zip`
- Port 8080 should be available (if it is in use, try disabling IPv6)

Service design components

The imported service design has 6 components: Server, Application Server, Database Server which requires composition, PetClinic DB Conf, PetClinic Application, and Apache Load Balancer.

A 'dbDetails' relationship is created between PetClinic DB Conf and the PetClinic application.

A 'loadBalancer' relationship is created between Apache Load Balancer and the PetClinic application.

A scalable topology group called 'WebServerGroup' is created and contains the PetClinic application, Tomcat server, and its respective vCenter server. Instance Count can be set at WebServerGroup.

The PetClinic application that you deploy through this design performs the following steps:

1. Creates vCenter servers.
 2. Deploys MySQL and Tomcat components on vCenter server.
 3. Configures PetClinicDB on MySQL.
 4. Deploys PetClinic Application on the Tomcat application server.
 5. Deploys the Apache load balancer on the vCenter server and the load balancer manages the created
1. WebServerGroup.

Service design customization

The following components are part of the imported application design:
CODAR_BP_PETCLINIC_APPLICATION_LOAD_BALANCER_v1.80.00.

PetClinic DB Conf

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>mysqlldb_conf.sh</code> is located
mysqlpassword	Enter the password for the MySQL database
mysqlusername	Enter the user name for the MySQL database
serviceCommand	It must be <code>sh /tmp/mysqlldb_conf.sh</code>
Port	It can be 22.
privatekeyPath	Enter privatekeyPath (can be blank)

PetClinic Application

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location, where <code>petclinic.war</code> is located
configurationUrl	Enter the HTTP location, where <code>petclinic_jdbc_conf.sh</code> is located
remotefilepath	Enter the path on the server. For example, it can be <code>/tmp</code>
serviceCommand	It must be <code>sh /tmp/mysqlldb_conf.sh</code>

Port	It can be 22.
privatekeyPath	Enter privatekeyPath (can be blank)

Apache Load Balancer

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location, where <code>lb_artifact.tar.gz</code> is located
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>loadbalancer_conf.sh</code> is located
installPath	This is an optional property
remotefilepath	Enter the path on the server. For example, it can be <code>/tmp</code>
serviceCommand	It must be <code>bash loadbalancer_conf.sh</code>
Port	It can be 22.
privatekeyPath	Enter privatekeyPath (can be blank)

PetClinic Application on existing servers

The `CODAR_BP_PETCLINIC_APPLICATION_ON_AN_EXISTING_INFRA_v1.80.00.zip` application design is installed if you select the Install Sample content option while installing Codar.

This design demonstrates how to deploy a two-tier PetClinic application using existing infrastructure components. This two-tier application has a DB component installed on MySQL and an application component installed on Tomcat.

Service design name

`CODAR_BP_PETCLINIC_APPLICATION_ON_AN_EXISTING_INFRA.zip`

Prerequisites

To use this content archive, you have to host the following files on a Web Server (Apache Server or a Maven repository), that can be accessed using HTTP.

1. `install_mysql.sh` - download this from wherever it is hosted internally.
2. `mysqldb_conf.sh` - download this from wherever it is hosted internally.
3. `install_tomcat.sh` - download this from wherever it is hosted internally.
4. `petclinic_jdbc_conf.sh` - download this from wherever it is hosted internally.

Note: The files above are located at `<CSA_HOME>\CSAKit-4.8\ContentArchives\topology\vmware_vcenter\petclinic\scripts`

5. `petclinic.war` - download this from Jenkins or wherever it is hosted internally.

6. `mysql-server_5.6.21-1ubuntu12.04_amd64.deb-bundle.tar` - download this from <http://ftp.kaist.ac.kr/mysql/Downloads/MySQL-5.6/> or wherever it is hosted internally.
7. `apache-tomcat-7.0.56.tar.gz` - download this from <http://tomcat.apache.org/download-70.cgi> or wherever it is hosted internally.

You must have vCenter virtual machines that have the following tools installed:

- Ubuntu 12.04 or later
- JDK 1.7
- `libaio1`
- `unzip`
- `zip`

Service design components

The imported service design has 6 components: Existing infrastructure (2 items), MySQL database, Tomcat, PetClinic DB Conf, and the PetClinic application.

A 'databaseConnection' relationship is created between "PetClinic DB" and "PetClinic Application".

The PetClinic application that you deploy through this design performs the following steps:

1. Accesses the IP address of the existing servers.
2. Deploys MySQL and Tomcat components on the existing server.
3. Configures PetClinicDB on MySQL.
4. Deploys PetClinic Application on the Tomcat application server.

Service design customization

Existing Infrastructure

Property	Description
IPADDRESS	Enter the IP address of a machine that has JDK 1.7, <code>libaio1</code> , <code>unzip</code> , and <code>zip</code>
Password/privateKey	Either one can be used. Enter the password of the server or the private key for secure SSH access.
username	Enter the user name for SSH access to the server deployed

MySQL Database

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where <code>apache-tomcat-7</code> is located. That is where <code>apache-tomcat-7.0.56.tar.gz</code> is located.
artifactUsername	Enter the user details for accessing HTTP location

configurationUrl	Enter the HTTP location, where <code>install_tomcat.sh</code> is located
installPath	Enter installPath (can be blank)
privatekeyPath	Enter privatekeyPath. For example, <code>C:\Users\Administrator\Downloads\CODAR.pem</code>
remoteFilePath	Enter the path on the server. For example it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh install_tomcat.sh</code>
sshPort	SSH port. It can be 22.

PetClinic DB Conf

Property	Description
configurationUrl	Enter the HTTP location, where <code>mysqldb_conf.sh</code> is located
mysqlpassword	Enter the password for the MySQL database
mysqlusername	Enter the user name for the MySQL database
serviceCommand	It must be <code>sh /tmp/mysqldb_conf.sh</code>
Port	It can be 22.
remoteFilePath	It must be <code>sh /tmp/mysqldb_conf.sh</code>

PetClinic Application

Property	Description
artifactUrl	artifactUrl
configurationUrl	Enter the HTTP location, where <code>petclinic_jdbc_conf.sh</code> is located
localfilepath	Enter the path in the local machine. For example, it can be <code>petclinic.war</code>
Port	It can be 22.
privatekeyPath	Enter privatekeyPath (can be blank)

PetClinic Application on vCenter

The `CODAR_BP_VCENTER_PETCLINIC_APPLICATION_v1.80.00.zip` application design is installed if you select the Install Sample content option while installing Codar.

Service design name

Prerequisites

To use this design, you have to host the following files on a Web Server (Apache Server or a Maven repository), that can be accessed using HTTP.

1. `install_mysql.sh` - download this from wherever it is hosted internally.
2. `mysqldb_conf.sh` - download this from wherever it is hosted internally.
3. `install_tomcat.sh` - download this from wherever it is hosted internally.
4. `petclinic_jdbc_conf.sh` - download this from wherever it is hosted internally.

Note: The files above are located at `<CSA_HOME>\CSAKit-4.8\ContentArchives\topology\vmware_vcenter\petclinic\scripts`

5. `petclinic.war` - download this from Jenkins or wherever it is hosted internally.
6. `mysql-server_5.6.21-1ubuntu12.04_amd64.deb-bundle.tar` - download this from <http://ftp.kaist.ac.kr/mysql/Downloads/MySQL-5.6/> or wherever it is hosted internally.
7. `apache-tomcat-7.0.56.tar.gz` - download this from <http://tomcat.apache.org/download-70.cgi> or wherever it is hosted internally.

You must have vCenter template that has the following tools installed:

- Ubuntu 12.04 or later
- JDK 1.7
- libaio1
- unzip
- zip
- Port 8080 must be available (if it is in use, try disabling IPv6)

Service design components

The imported service design has 6 components: vCenter (2 items), MYSQL database, Tomcat application server, PetClinic DB Conf, and the PetClinic application.

The PetClinic application that you deploy through this design performs the following steps:

1. Creates vCenter servers.
2. Deploys MySQL and Tomcat components on the vCenter server.
3. Configures PetClinicDB on MySQL.
4. Deploys PetClinic Application on the Tomcat application server.

Service design customization

vCenter Server

Property	Description
CustomizationSpec	Enter the customization specification (to be obtained from admin)
memorySize	Enter the memory size.
username	Enter the user name (to be obtained from admin)

password	Enter the password (to be obtained from admin)
privateKey	Enter the private key (not required, can be blank)
vmFolder	Enter the vmFolder
vmNamePrefix	Enter the VM name prefix
vmTemplateReference	Enter vmTemplateReference (to be obtained from admin)
cpuCount	Enter the CPU count

MySQL Database

Property	Description
artifactPassword	Enter the password to access the HTTP location
artifactURL	Enter the HTTP location where MySQL database. That is where <code>mysql-server_5.6.21-1ubuntu12.04_amd64.deb-bundle.tar</code> is located.
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>install_mysql.sh</code> is located
installPath	Enter installPath (can be blank)
privatekeyPath	Enter privatekeyPath. (can be blank)
remoteFilePath	Enter the path on the server. For example it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh install_mysql.sh</code>
sshPort	SSH port. It can be 22.

Tomcat Application Server

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where apache-tomcat-7 is located. That is where <code>apache-tomcat-7.0.56.tar.gz</code> is located.
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>install_tomcat.sh</code> is located
installPath	Enter the tomcat install path including the tomcat home directory name. For example, <code>/opt/tomcat7</code>
privatekeyPath	Enter privatekeyPath (can be blank)

remoteFilePath	Enter the path on the server. For example, it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh install_tomcat.sh</code>
sshPort	SSH port. It can be 22.

PetClinic DB

Property	Description
artifactPassword	Enter the password for accessing the HTTP location
artifactUrl	Enter the HTTP location where <code>apache-tomcat-7.0.56.tar.gz</code> is located.
artifactUsername	Enter the user name for accessing the HTTP location
configurationUrl	Enter the HTTP location, where <code>mysqlldb_conf.sh</code> is located
mysqlpassword	Enter the password for the MySQL database
mysqlusername	Enter the user name for the MySQL database
serviceCommand	It must be <code>sh /tmp/mysqlldb_conf.sh</code>
Port	It can be 22.
remoteFilePath	Enter the path on the server. It can be <code>/tmp</code>
privatekeyPath	Enter privatekeyPath (can be blank)

PetClinic Application

Property	Description
artifactUsername	Enter the user details for accessing the HTTP location
artifactPassword	Enter the password for accessing the HTTP location
artifactUrl	Enter the HTTP location where <code>petclinic.war</code> is located
configurationUrl	Enter the HTTP location, where <code>petclinic_jdbc_conf.sh</code> is located
localfilepath	Enter the path in the local machine. For example, it can be <code>petclinic.war</code>
Port	It can be 22.
privatekeyPath	Enter privatekeyPath (can be blank)
remoteFilePath	Enter the path on the server. It can be <code>/tmp</code>
serviceCommand	It must be <code>sh /tmp/petclinic_jdbc_conf.sh</code>

The PetClinic Application component has a modify action for the redeploy feature.

Redeploy content creation

This section explains how to enable the Modify action for a component.

Flows

The individual flows are placed under the version folder. The name of a flow is used automatically as an operation name in the new component. The flow name is important for recognition of the lifecycle phase related to the operation.

Each OO flow should have an input property defined as a constant called LIFECYCLE_PHASE. The property does not relate to the flow logic. It is used as meta-information marking the flow's purpose related to the component lifecycle.

As you can see, there is a duplicate indicator of the operation lifecycle phase. Both the flow name (beginning of the name) and LIFECYCLE_PHASE input property need to be set appropriately.

This duplication will be removed in a future release. Currently, the flow name and the value of the constant property should be set as follows:

Flow purpose	Flow name prefix	LIFECYCLE_PHASE
Deployment	Deploy or Create	deploying
Undeployment	Undeploy or Delete	undeploying
Modification	Modify	modifying
Undo a successful modification	Unmodify	unmodifying
Handle a failure during deployment		Deploying_failure
Handle a failure during undeployment		Deploying_failure
Handle a failure during modification		Deploying_failure
Custom public action executable on a deployed instance		deployed

Input and output properties

The OO server automatically decorates all flows with a Result output property. However, it is suggested to define flow outputs explicitly as output properties:

- *response* – Mandatory property. Each OO flow used in a component operation should have a response output property. Codar relies on the response property to determine the state of the execution.
- For deployment, undeployment flows and flows related to public actions, the value should be either success or failure based on flow execution response.
- For Modify flow, the value should be success, noop or failure. Success indicates that the attempted modification was successful. Noop indicates that no action was taken that would affect the component state or properties. Failure indicates that the attempted modification unsuccessful.

- For Unmodify flow should overwrite the response property to failure if the unmodify failed and to noop if Unmodify succeeded to indicate that from an overall component point of view the component state and property values are unaffected due to modify transition. Modify Failure flow should set the response property to always failure indicate the failure to modify this component.
- If you include the deploy failure handler and/or undeploy failure handler, make sure that the response is being set correctly. The fact that these actions are called should result in the failure response.
- *result* – Optional property. Each OO flow used in a component operation should have a result output
- *property*. It can hold any contextual message, like detailed info about failure.

Codar 1.80 supports the modification of properties on an active subscription. Codar will pass previous property values to modification OO flows if the flows explicitly express interest in knowing these values - by defining additional flow inputs with a *prev_* prefix. As an example, if the modification OO flow defines an input of *memorySize* and is interested in knowing its previous value, it must also define an input of *prev_memorySize*.

A component's Modify flow may communicate the status of the modification, an error code or any other useful information to Modify Failure flow by placing such information in an output field called *modifyReturnValue*. This value may be received by the Modify Failure flow in an input field called *modifyFailureValue* and allows the Modify Failure flow to do cleanup the effect of failure in Modify flow intelligently, especially for a complex multi-step flows when failure might have occurred at any step which can be indicated by returning specific error code from the Modify flow via *ModifyReturnValue*. If these fields are set on the OO flows, the mapping between them is automatically handled within Codar and must not be modified from the Component Editor.

PetClinic Application

This design demonstrates how to deploy a two-tier partial PetClinic Application using vCenter. This two-tier application has Database Server and Application Server requirement. Another service design provides the infrastructure for the partial PetClinic application, this infrastructure design has MySQL component, which serves as Database Server and Tomcat component which serves as the Application Server both hosted on vCenter servers.

Folder contents

Service design content archive

- CSA_BP_PETCLINIC_TWO_TIER_INFRASTRUCTURE_v1.50.00.zip
- CSA_BP_PETCLINIC_APPLICATION_v1.50.00.zip

Scripts

- Scripts path: <CSA_HOME>\CSAKit-4.5\Content Archives\topology\vmware vcenter\petclinic\scripts
- *install_mysql.sh*
- *install_tomcat.sh*
- *mysqldb_conf.sh*
- *petclinic_jdbc_conf.sh*

Prerequisites

To use this content archive, you have to host the following files on a Web Server (Apache Server or a Maven repository), that can be accessed using HTTP.

1. *install_mysql.sh* - download this from wherever it is hosted internally.

2. `mysqldb_conf.sh` - download this from wherever it is hosted internally.
3. `install_tomcat.sh` - download this from wherever it is hosted internally.
4. `petclinic_jdbc_conf.sh` - download this from wherever it is hosted internally.

Note: The files above are located at `<CSA_HOME>\CSAKit-4.8\ContentArchives\topology\vmware_vcenter\petclinic\scripts`

5. `petclinic.war` - download this from Jenkins or wherever it is hosted internally.
6. `mysql-server_5.6.21-1ubuntu12.04_amd64.deb-bundle.tar` - download this from <http://ftp.kaist.ac.kr/mysql/Downloads/MySQL-5.6/> or wherever it is hosted internally.
7. `apache-tomcat-7.0.56.tar.gz` - download this from <http://tomcat.apache.org/download-70.cgi> or wherever it is hosted internally.

You must have vCenter virtual machines that have the following tools installed:

- Ubuntu 12.04 or later
- JDK 1.7
- `libaio1`
- `unzip`
- `zip`

You must configure the following:

- For users other than 'root' should have home directory. This is required for logging into the target system and expects home directory to be present.
- For sudo users, the password prompt should be disabled for running the shell commands. This is required because the system will try to perform silent installation of packages and does not expect any prompts. If there is any prompts then OO flow execution halts and fails.

Service design components

The imported design has 4 components: Database Server, Tomcat, PetClinic DB Conf, PetClinic Application, and the Application Server. Database Server is the capability components with MySQL 5.6 requirement, and Application Server is the capability component with Tomcat 7.

Two tier Infrastructure with MySQL and the imported Tomcat service design has 4 components: MySQL, Tomcat, and vCenter Server (number of servers: 2).

The PetClinic Application needs the infrastructure design for deployment. The test run wizard lists the matching infrastructure design and has the PetClinic two tier infrastructure design listed.

The PetClinic two tier infrastructure performs the following actions:

1. Deploys vCenter servers
2. Configures PetClinic Database on MySQL.
3. Deploys MySQL and Tomcat components on the vCenter server
4. Deploys the PetClinic Application on the Tomcat application server

Service design customization

PetClinic Application

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where <code>petclinic.war</code> is located.
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>petclinic_jdbc_conf.sh</code> is located
localfilePath	Enter the file name in the local machine. It can be <code>petclinic.war</code>
Port	SSH port. It can be 22.
privatekeyPath	Leave it blank
remoteFilePath	Enter the path on the server. For example it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh /tmp/petclinic_jdbc_conf.sh</code>

MySQL Database

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where MySQL is located. That is where <code>mysql-server_5.6.21-1ubuntu12.04_amd64.deb-bundle.tar</code> is located.
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>install_mysql.sh</code> is located
installPath	Leave it blank
privatekeyPath	Leave it blank
remoteFilePath	Enter the path on the server. For example it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh install_mysql.sh</code>
sshPort	SSH port. It can be 22.

PetClinic DB Conf

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>mysqlldb_conf.sh</code> is located
mysqlpassword	Enter the password for the MySQL database

mysqlusername	Enter the user name for the MySQL database
serviceCommand	It must be <code>sh /tmp/mysqlldb_conf.sh</code>
Port	It can be 22.
remoteFilePath	Enter the path on the server. It can be <code>/tmp</code>
privatekeyPath	Leave it blank

Tomcat Application Server

Property	Description
artifactPassword	Enter the password for accessing HTTP location
artifactUrl	Enter the HTTP location where apache-tomcat-7 is located. That is where <code>apache-tomcat-7.0.56.tar.gz</code> is located.
artifactUsername	Enter the user details for accessing HTTP location
configurationUrl	Enter the HTTP location, where <code>install_tomcat.sh</code> is located
installPath	Enter the tomcat install path including the tomcat home directory name. For example, <code>/opt/tomcat7</code>
privatekeyPath	Leave it blank
remoteFilePath	Enter the path on the server. For example, it can be <code>/tmp/</code>
serviceCommand	It must be <code>sh install_tomcat.sh</code>
sshPort	SSH port. It can be 22.

vCenter Server

Property	Description
CustomizationSpec	Enter the customization specification value for the vCenter template
username	Enter the user name for SSH access to the deployed server
Password/privateKey	Either one can be used. Enter the password of the server or the private key for secure SSH access
vmNamePrefix	Enter the VM name prefix
vmTemplateReference	Enter the Ubuntu VM template name

Notes: vCenter server must not have MySQL and Tomcat installed. If it is already installed then the design might fail because port is in use. Remove the values in privatekey and password (by default the user name and password is the same as in the template). The content is validated only with IPv4.

Existing-Infra

The Existing Infra capsule is installed if you select the Existing-Infra option while installing Codar.

This capsule allows using existing servers to deploy applications.

Prerequisites

To use the existing server component in the design, create a server configuration in Codar.

If the design is managed through release pipeline in Codar, then create a resource pool with this server configuration and associate it to the lifecycle stage.

Service design components

It has the server component 'Existing Server (1.80.0000)'.

Service design customization

Property	Description
instanceId	Service instance ID. This is hidden and populated automatically.
ipAddress	IP Address/ host name of existing server. This should always be represented by an existing server in Codar. Since the IP Address is not validated at the time of adding a server, you must add valid machines with valid credentials. The ipAddress is mandatory if the server is used for Test Run calls. The servers associated to this ipAddress should not be tagged to a non-shared pool. For Codar Pipeline use cases, if the ipAddress is specified for this component at the design level, then the server belonging to this ipAddress should be part of the associated pools for the package. If no pools are associated, then the server should not be associated to a non-shared pool.
isBlocked	This is a flag to govern multiple application deployment on a server. By default, this option is set to false. When the option is set to false, there is a possibility of multiple applications being deployed on the same server (if no eligible server with zero deployment exists). Setting it to true will enforce that a single application be deployed on the server. If there are no eligible machines with zero prior deployments, then the deployment fails. You can change this property setting by performing a 'Save As' of the original component. This component always tries to pick up the oldest least used server. The least used count refers to the number of times a server can be used for deployments. If the 'isBlocked' property is set to true at the component level, then always the oldest unused server will be picked up.
Capability	Not supported.
Password	Password of server configuration
Private Key	Private key of server configuration
SSH Port Configuration	SSH port of server configuration.
User Name	User name of server configuration

Test run for Design Scaling is not supported with this component.

Auto-Generate

The Auto_Generate capsule is installed if you select the Auto-Generate option while installing Codar.

This capsule enables generation of topology designs complete with all the components, OO flow, and package which is ready to deploy.

Prerequisites

Codar supports generation of topology designs on machines with the following flavors of Unix and Microsoft SQL.

Unix: Both single-tier and two-tier designs supported with the following flavors:

- RHEL (7.x)
- Ubuntu (12.04 and 14.04)

Microsoft SQL: Only single-tier designs supported.

- Windows 2008 (64-bit machine)
- Windows server 2012 (64-bit machine)

Service design components

This capsule has the following components:

- Tomcat Application Server. This component installs tomcat over the infrastructure provided.
- Web Application: This component deploys the web application on the tomcat server.
- MySQL Database Server: This component installs MySQL with default credentials root/2the#Moon

Script-Action

The Script-Action capsule is installed if you select Script-Action option while installing Codar.

This capsule enables creation of script release gate action at any stage in a lifecycle, to execute scripts (shell scripts for Linux and Powershell for Windows) in the target machines during the promotion of a package.

Prerequisites

Before using this content, you must create a resource provider with 'Provider Type' as GitHub. Refer to Codar Help for more information.

Send documentation feedback

If you have comments about this document, you can [contact the documentation team](#) by email.

If an email client is configured on this system, click the link above and an email window opens. Add subject and your feedback to the email, and click send. If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to clouddocs@hpe.com.

We appreciate your feedback!