Codar

Software version: 1.81

Existing Servers

Document release date: May 2017

Software release date: May 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

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

What is HPE Codar

HPE Codar is an application release management software that provides deployment and release management of complex multi-tier applications across the application lifecycle. HPE Codar facilitates Continuous Integration (CI) and Continuous Deployment (CD) by orchestration application deployment and testing across different lifecycle stages. It automates the deployment of applications by embracing existing content from HPE OO, Chef, HPE SA, and so on, and represents this content as components.

These components are used in a graphical topology designer to create an application model. These model describe how Applications are deployed. Typically models consist of various components like compute, network, storage, platform etc.

Applications are deployed on machines created by VCenter or AWS or on existing servers (machines pre-created). This document provides information about how Codar can be used to deploy applications on existing servers and related features.

What are Existing Servers

There are two approaches on how infrastructure can be requested and provisioned:

- JIT provisioning – Provision a server exactly when you need it.
- Pre-provisioning – Provision a bunch of servers in advance based on anticipated demand.

Both approaches have their own benefits:

- JIT provisioning is the most optimal way of consuming resources as the servers are provisioned and un-provisioned instantly just before and after their use respectively
- Pre-provisioning helps in increasing the speed of release automation process. But servers may still remain in the pool even if they are not required anymore.

Typical design may consist of VCenter or Amazon servers. When these designs are used for application deployment, Codar creates these servers. But in some cases the Servers might have been created through an external systems. Those servers are classified as Existing Servers. These servers can also be used in Release Automation by using Existing Server feature in HPE Codar

Existing Server management helps our customers who chose the pre-provisioning method over JIT provisioning.

Codar does not discover servers in a user environment. Lifecycle management of the externally created servers is not done by Codar. Codar uses the information provided

# Use case – Use Existing Severs to Deploy Applications

All the pre-created servers or existing servers must be added to Codar.

Add Servers to Codar
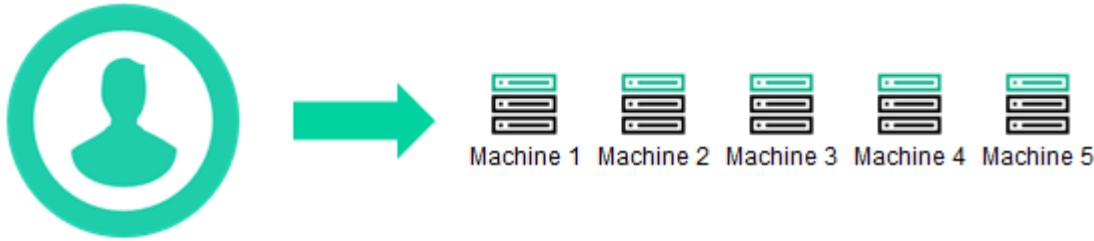
Architect adds machines or imports from CLI

The Architect must provide information such as IP Address, User ID, Password or Key to access the machine. At the time of adding the machine, no validation is done to check if the machine exists or is it running. The machine-related information is also not validated.
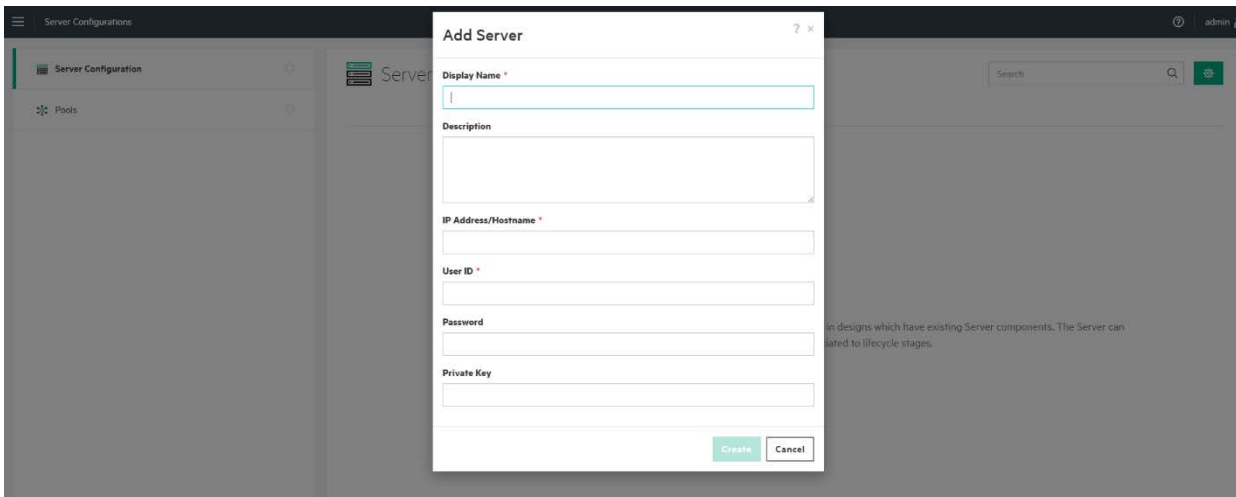
After the machines are added, user will be able to edit or delete them. There is no communication or synchronization between Codar and the servers.

Architect will also be able to import bulk servers from CLI. Download the latest hotfix for 1.80 which has updates to codar-clis.jar. (By default it is available in Codar 1.81).

Add servers with the following command:

```
codarexec.bat serverconfiguration addservers [-f filepath] [-s URL of the codar instance] [-u codar
username] [-p codar password]
```

Sample CSV File

Example Templates for a csv file:

name,description,IPOrHostName,username,password,privateKey

DevMachine1,For Development,10.2.12.1,admin,cloud,#atdjjsnsnsn$

DevMachine2,For Development,10.2.12.2,admin,cloud,#atdjjsnsnsn1$

DevMachine3,For Development,10.2.12.3,admin,cloud,#atdjjsnsnsn2$Note

Notes:

- File size limit is 300KB.

- CSV file should contain the headers in the order – name, description, IPOrHostName, username, password, privateKey

- The headers are case-insensitive

- If value is not given in any particular row for any corresponding headers then it should be comma separated.

  Example :(In example below "description" header in the first row has no actual description information)

  name,description,IPOrHostName,username,password,privateKey

  DevMachine,,10.2.12.1,admin,cloud,#atdjjsnsnsn$

  QaMachine,Its a qa machine,10.2.12.1,admin,cloud,#atdjjsnsnsn$

- name,IPOrHostName and username are mandatory header fields. Non mandatory header fields should be comma separated.
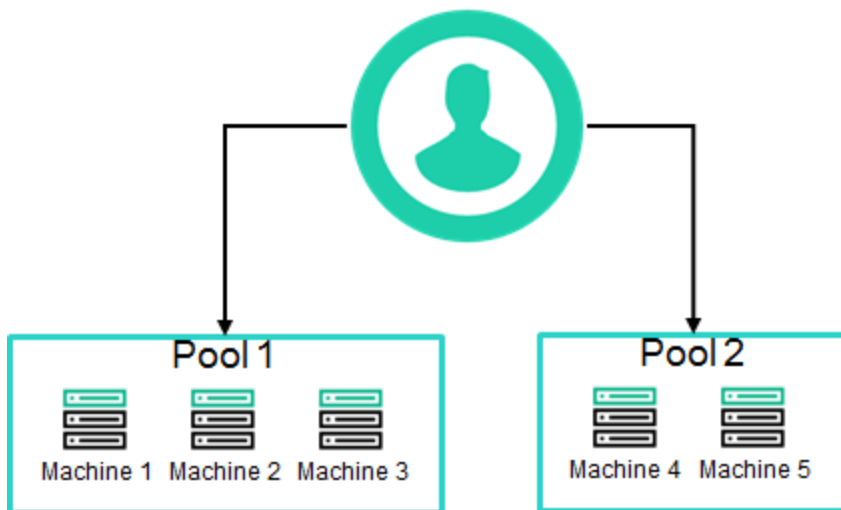
  Example:

  NAME,,IPORHOSTNAME,USERNAME,,

  abc,,10.2.12.1,admin,,

  abc,,10.2.12.2,admin,,

- Every row in csv file must have same number of fields equal to the number of headers.

Create Server Pools

A Pool is a set or collection of servers. The servers which are created in the previous step are grouped into logical structures called Pools.


Architect groups the machines into Pools

A Pool is a set or collection of servers. The servers which are created in previous step are grouped into logical structures called Pools. These Pools are used in lifecycle stages as explained later in the document. There are 2 kinds of Pools.

These Pools are used in lifecycle stages as explained later in the document. There are two kinds of Pools:

- Non-Shared Pool – This pool should consist of servers which are meant for specific applications and specific lifecycle stages. Servers in Non-Shared pool cannot be used for multiple applications and different lifecycle stages. For ex if a server meant for PetStore application in Development stage then the server should be added to Non-Shared Pool
- Shared Pool – This pool should consist of servers which can be used across different applications and different lifecycle stages. Typically servers which has Databases, servers which are used for creating containers etc.
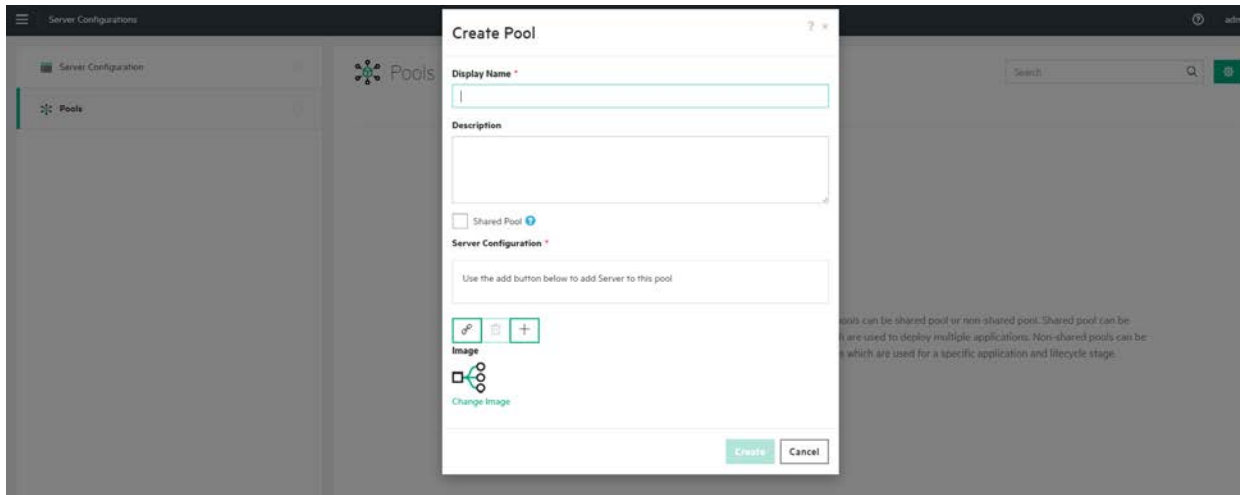
Notes:

- If a machine is added to Non-Shared Pool, then it cannot be added to Shared Pool. Example: If Server 1 is added in a Non-Shared Pool, then it cannot be used in any other Pool.
- Same servers can be added to multiple Shared Pools. Example: Server 1 can exist in 2 Shared Pools.

Architect will be able to edit or delete the Pools.

Existing Server Component

Codar has a new Component call "Existing Server (1.80.0000)" available from 1.80. This component is common for both Windows and Linux Platform. It has to be used for applications on existing servers. This component has the intelligence to pick the machines from the pools

"Existing Server (1.80.0000)" has following properties visible in designer:

- ipAddress – ipAddress/hostname of Existing Server.
  - This server should be added in Codar using "Add Server" as explained above. The IP Address is not validated at the time of adding server. It is the responsibility of the user to add valid machines with valid credentials which are up and running.
  - It is mandatory to provide ipAddress/hostname for the Test Run to be successful. The server associated to this ipAddress/hostname should not be tagged to non-shared pool.
  - During Package deployment in Pipeline Management, if ipAddress/hostname is specified for this component in the design, then the server belonging to this ipAddress/hostname should be the part of associated pools for the package and if no pools are associated then server should not be associated to non-shared pool.
- sshPort – Port number required to connect to the Server

"Existing Server (1.80.0000)" has following properties NOT visible in designer

- isBlocked – This is a flag is to govern if the same server can be chosen to deploy applications during "Scale Out"
  - By default, it is FALSE. Which means that there is a possibility of multiple applications deployed on the same Server (if no eligible server with zero deployment exists).
  - If it is set to TRUE in Component property, it will enforce that a single application is deployed on the server. If there are no eligible machines with zero prior deployments, then the deployment fails.
  - This property can be changed by performing a "Save As" from OOTB component.
- instanceId: - Service Instance Id. This is hidden and populated automatically.

Notes:

- Capability: - It is not supported in this release. This will available in future release.
- Scaling Group is not supported with this component in Test Run.

Existing Server Design

The following sample design shows how Existing Server can be used:
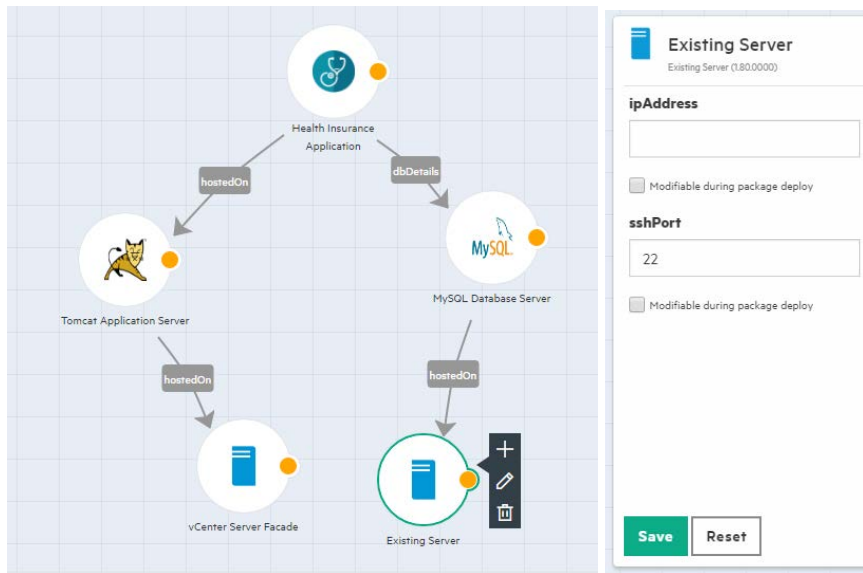
For Test Run of the design, Architect has to provide the ipAddress/Hostname and sshPort of the machines. These machines must be added using "Add Server" and the server **should not** be part of Non Shared pool.

Test Run will **NOT** pick the machines from the pools automatically. It uses the machine details provided by the user in the design or in the Test Run wizard.

Test Run for Scaling Design is not supported with this new component.

Associate Pools to Lifecycle Stages for Package Deployment

Configuring Existing Server Pools
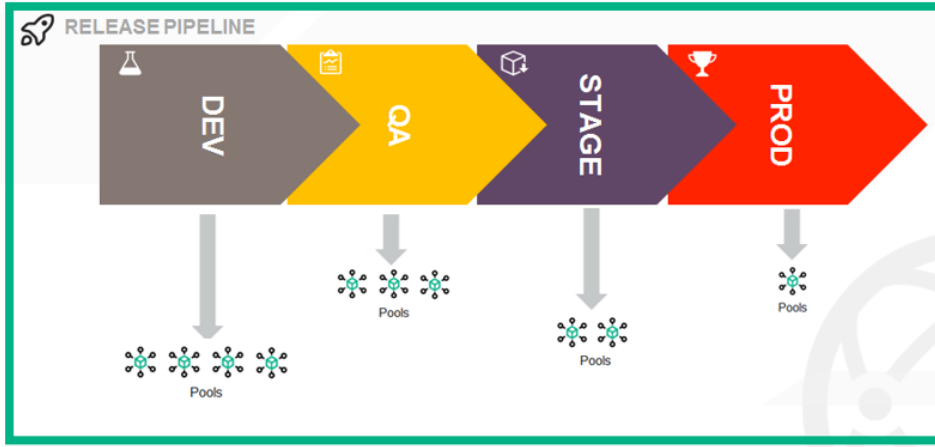
Architect can associate Pools to lifecycle stages in Package UI. It shows all the Pools created. Once the Non-Shared Pool is associated to a lifecycle stage, it cannot be used again. Shared Pool can be used multiple times across different lifecycle stages.
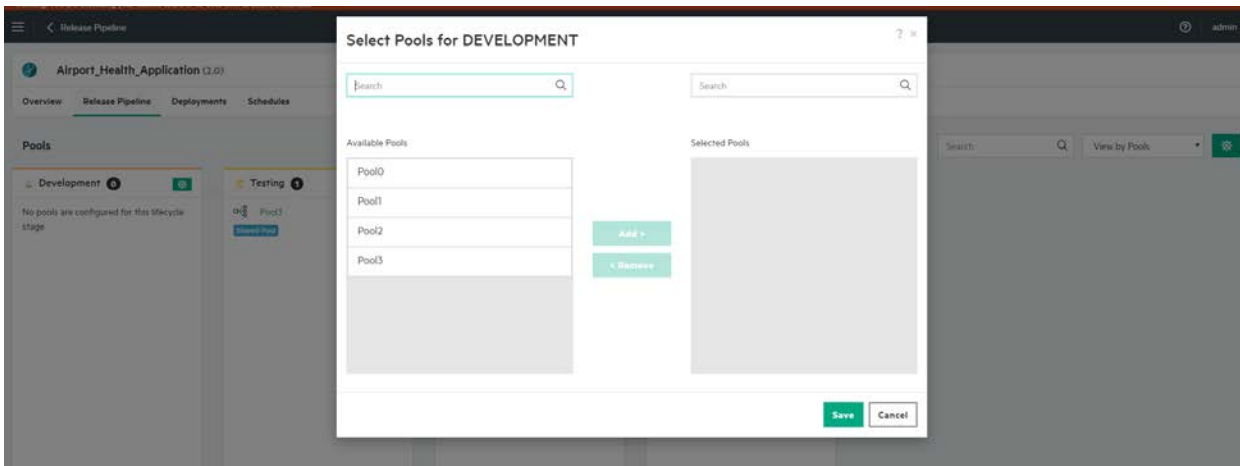
If the design consists of Existing Server component, then during package deployment, the component picks the Server details from Pools configured for the lifecycle stage. The Server is picked by following Algorithm:

- During Deployment, Codar will check if there are any Pools associated with the design for the lifecycle stage. If yes then it will pick the machine from the Pools. If all the machines are already used then it will pick up the oldest least used machine from the pool.
- Every time a machine is used for deployment in a lifecycle stage, an internal counter (not visible to user) is incremented.
- When the application is un-deployed, the counter is decremented.
- If no Pools are associated for the design for the lifecycle stage then it will look into all Pools which are not marked as Shared.

Notes:

- Pool can be delete even if it associated with lifecycle stage
- Servers can be deleted even if it is part of Pools

Scaling Group and Scale Out

Designs can be created with scaling group as shown below.
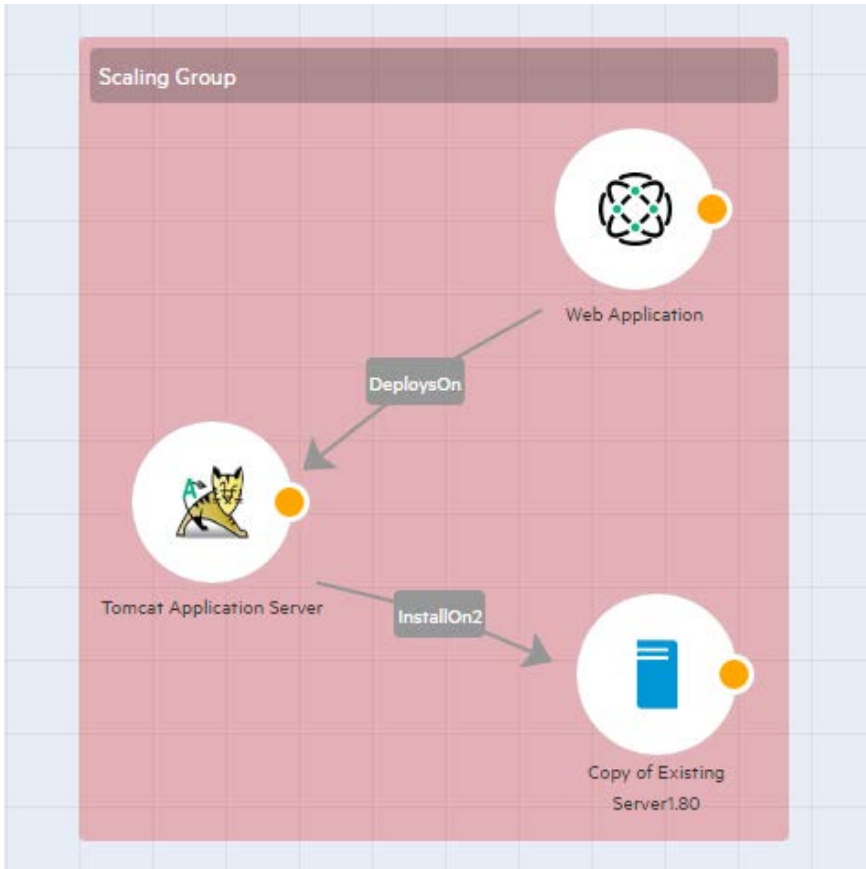
Scale out in Lifecycle Stages

To scale out the applications in different lifecycle stages, create a package in Release Pipeline. Deploy the package and upon successful deployment, Scale Out option is available. Based on the value provided, Scale out will create multiple instance of the stack.

For Scale Out, the pools should have enough number of servers associated with them for the number of instances specified at the time of scale out.

Note: During scale out flow, the "Modify flow" is called on the instances which are already deployed (Modify flow is also used for Redeploy). So the modify flow must not break the deployments already made on the Servers.

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!