



**Hewlett Packard**  
Enterprise

**Codar**

Software version: 1.81

# **Continuous Integration, Continuous Deployment and Continuous Testing by HPE Codar using TFS**

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

© 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

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 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

<b>Legal notices</b> .....	<b>2</b>
<b>What is Codar?</b> .....	<b>4</b>
<b>Why is Codar required?</b> .....	<b>4</b>
<b>Microsoft Team Foundation Server (TFS)</b> .....	<b>4</b>
<b>End-to-end flow</b> .....	<b>4</b>
<b>Continuous integration and deployment using Codar and TFS (build tool)</b> .....	<b>5</b>
Software required for the integration .....	5
Configuring TFS .....	5
Steps to create a Team Project in TFS web .....	6
<b>Send documentation feedback</b> .....	<b>12</b>

# What is Codar?

HPE Codar is a continuous delivery solution that provides deployment and release management of complex multi-tier applications across the application lifecycle. It automates the deployment of applications by embracing existing content from Chef, HPE SA, and so on and representing this content as components. These components can be used in a graphical topology designer to create an application model.

One of the important features of Codar is that the model is used to trigger deployments automatically with TFS.

This document provides information about integrating Codar with TFS.

# Why is Codar required?

Software engineering builds are subject to continuous deployment and testing on the principles of frequent code commits, build automation, faster and frequent builds, automated application deployment, and test automation. On top of continuous integration, software development teams also continuously deliver qualified software applications to their testing and production teams. One of the challenges that most software development teams face in the process of continuous integration and continuous delivery is the ability to automate the deployment of applications in a simple and consistent manner and run tests on the deployed instance. Codar is built to solve this problem.

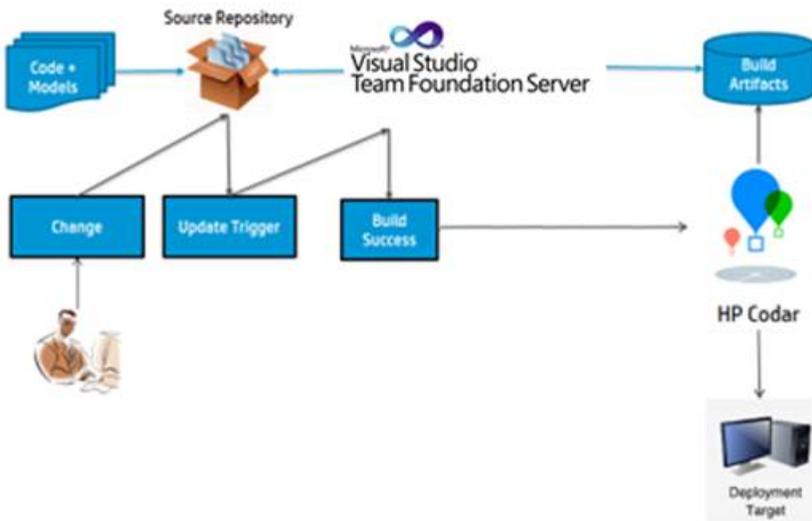
This white paper describes how to integrate Codar with TFS.

# Microsoft Team Foundation Server (TFS)

TFS is a Microsoft product that provides source code management, reporting, automated builds, lab management, and release management capabilities.

# End-to-end flow

The image below shows the end-to-end to flow of how Codar and TFS are integrated. In this use case, TFS is the orchestrator.



*n*

The developer makes a code change and checks it into source control. The source control system sends an update trigger and TFS triggers a build. After the build is successful, TFS invokes the Codar plugin which in turn invokes the Codar API and triggers a deployment.

**Note:** The integration between TFS and Codar supports only string type of property values.

## Continuous integration and deployment using Codar and TFS (build tool)

### Software required for the integration

For continuous deployment, you must integrate TFS with Codar. To integrate TFS with Codar, you must install the following software:

1. TFS 2015 Update 2 or later version. This is required to create a team project from the web portal.
2. MSSQL 2014.

### Configuring TFS

After installing the software listed at Software required for the integration, configure TFS as follows:

1. Log in to TFS.
2. Click 'Administer' tab on Home Page.

A separate web page opens, with the following tabs:

- Control Panel
- Extensions
- Agent Pools.

3. Click the 'Agent Pools' tab.

The left pane displays the available agent pools. By default, there is only one pool named as Default.

4. You can add new pools, by clicking the 'New Pool' button above the left pane.
5. You must now add an agent to the pool. To do so, click the 'Download Agent' button above left pane. A .zip file will be downloaded.
6. Extract the .zip file.
7. Go to the extracted folder using Windows Explorer and run the ConfigureAgent.cmd file.
8. When prompted to enter an agent name, specify any name.
9. Specify the URL of your Team Foundation Server.
10. Specify the pool name if you created a new pool. By default, 'Default' pool will be selected.
11. Specify the path of work folder.
12. When prompted to select if you want to install the agent as Windows Service, select Y/N, as per your requirement.

The system creates the agent, if you selected yes in the above step.

13. To verify, click the 'Agent Pools' tab. The newly created agent will be listed in the right pane.

Once the agent is created, you can schedule builds for your project.

## Steps to create a Team Project in TFS web

1. Open the web portal for TFS in the following URL format:- `http://<ipAddress>:<port>/tfs`.
2. If prompted, enter the credentials given during TFS installation.
3. Select 'Administration Overview' from the gear icon in the top right. The 'Administration Overview' page opens in a separate browser tab.
4. Click '*View the collection administration page*' to create the team project.

**Control panel**

Control panel | Access levels | Legacy Extensions | Agent pools

Administer your server

**TEAM FOUNDATION**  
Select a collection, team project or team you want to administer

Search...  
DefaultCollection

Collection name **DefaultCollection**  
Status Online  
Team projects 0

**Administration tasks**  
[Manage collection security and group membership](#)  
[View the collection administration page](#)

5. Click 'New team project'.

Control panel > DefaultCollection

Overview | Security | Build | Agent queues | Extensions\*

Not all collection level administration is exposed in the web experience. For all administration operations at the collection level please contact your administrator.

**Collection profile**

Name  
**DefaultCollection**

Description

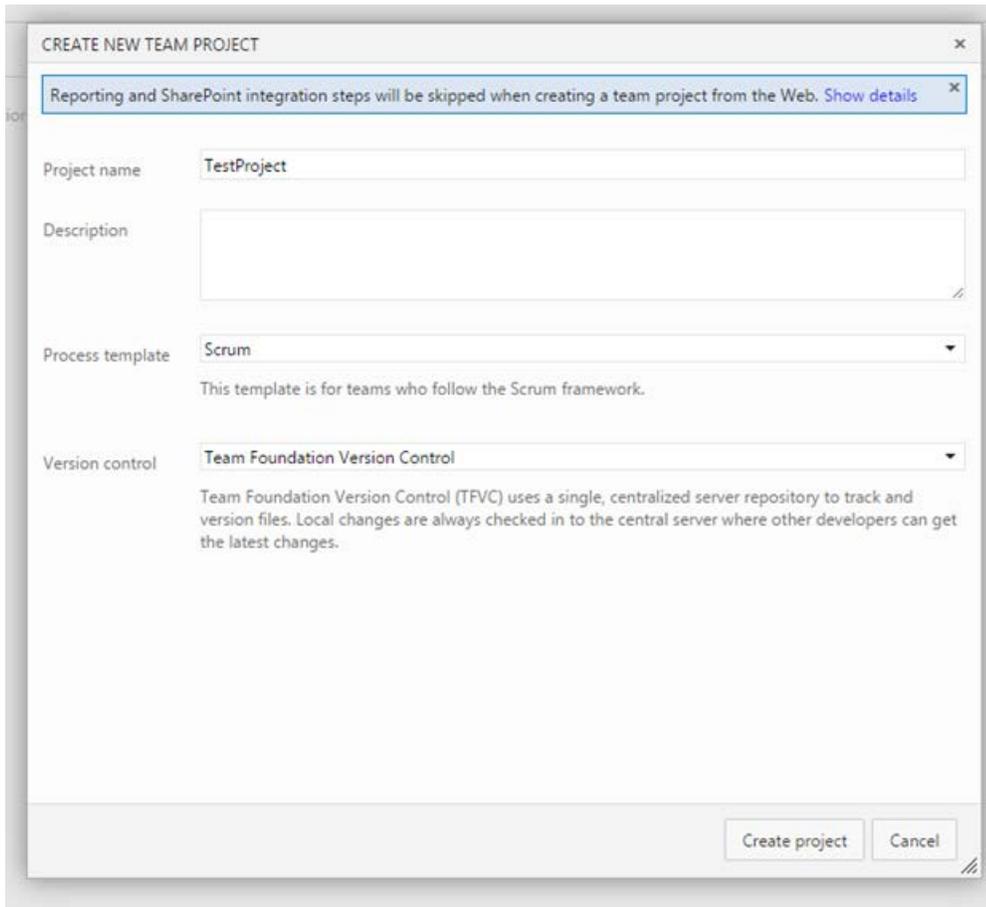
**Projects**

New team project | ↻

**Project name**

No projects available.

6. Provide a name for your new Team project in 'Project name'..
7. Select 'Scrum' as 'Process template'.
8. Select 'Team Foundation Version Control' as 'Version control'
9. Click 'Create Project'.



10. Now, come back to TFS Web Portal homepage and click on the created team project.

11. From the 'Home', 'Code', 'Work', 'Build', 'Test' and 'Review' tabs, select the 'Code' tab.

**Note:** If you need to create more than one team project, follow the above steps to create the team project. Then, come back to TFS Web Portal homepage and navigate to the desired project.

Upload Codar plugin to TFS:

1. In the 'About Team Foundation Server' page, click 'Browse for Extensions' icon  at the top right corner and select 'Browse for extensions'.
2. In the 'Team Foundation Server Extensions' page, click 'Manage Extensions'.
3. In the 'Manage Extensions' page, click 'Upload new extension'.

**Note:** If you have already installed the Codar plugin, you must uninstall/remove it from the 'Manage Extensions' page, and also delete the install folder of the plugin, which is by default found in the following location: C:\agent\tasks. Next, log out of TFS, re-login, and do a fresh installation of the plugin. This may also require a redoing of the build definition.

4. In the 'Upload new item' dialog box, click 'Browse' button and navigate to the Codar plugin.  
The plugin must have the .vsix extension.
5. Select the Codar plugin and click 'Upload' button.  
The Codar plugin now appears in the 'Manage Extensions' page.
6. Mouse-over the Codar plugin and click 'Install'.
7. In the 'Collection' dialog, select a Team Project Collection where you need to install the plugin.
8. Click 'Confirm'.

To create a new build definition:

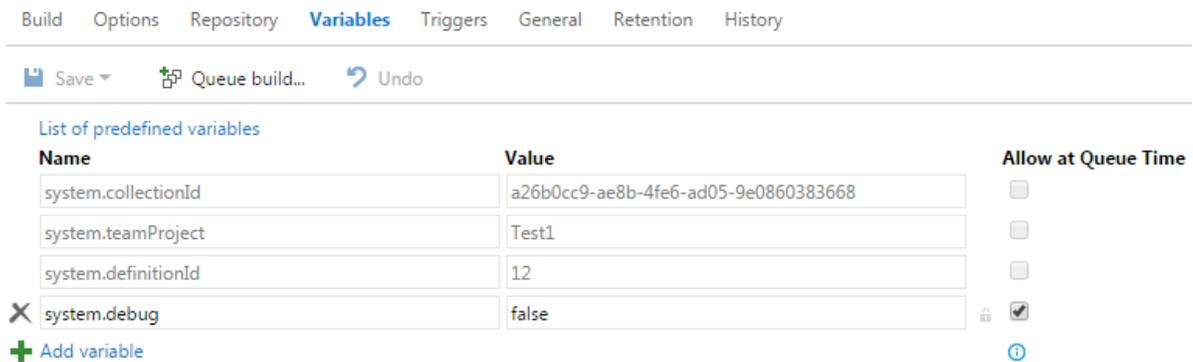
1. In Explorer, select the project for which you want to create a new build definition.
2. On the 'Build' menu, click '+' icon to create a **New Build Definition**.
3. Select the 'Empty' option to create the build definition.
4. Click 'Next'.
5. Specify the following settings:
  - Repository source
  - Repository
  - Default agent queue
6. Click 'Create'.

To Add Build Steps:

1. Right click on the created build definition, and select 'Edit' option.
2. Click 'Add build step'.
3. In the 'Add Task' window, click 'Deploy', search for 'HPE Codar', and add the task.
4. Close the 'Add Task' window.

To Add Variables:

1. Click 'Variables' tab and verify if the following variables are present with values:
  - system.collectionID
  - system.teamProject
  - system.definitionID
2. Add the following variable:
  - system.debug : false

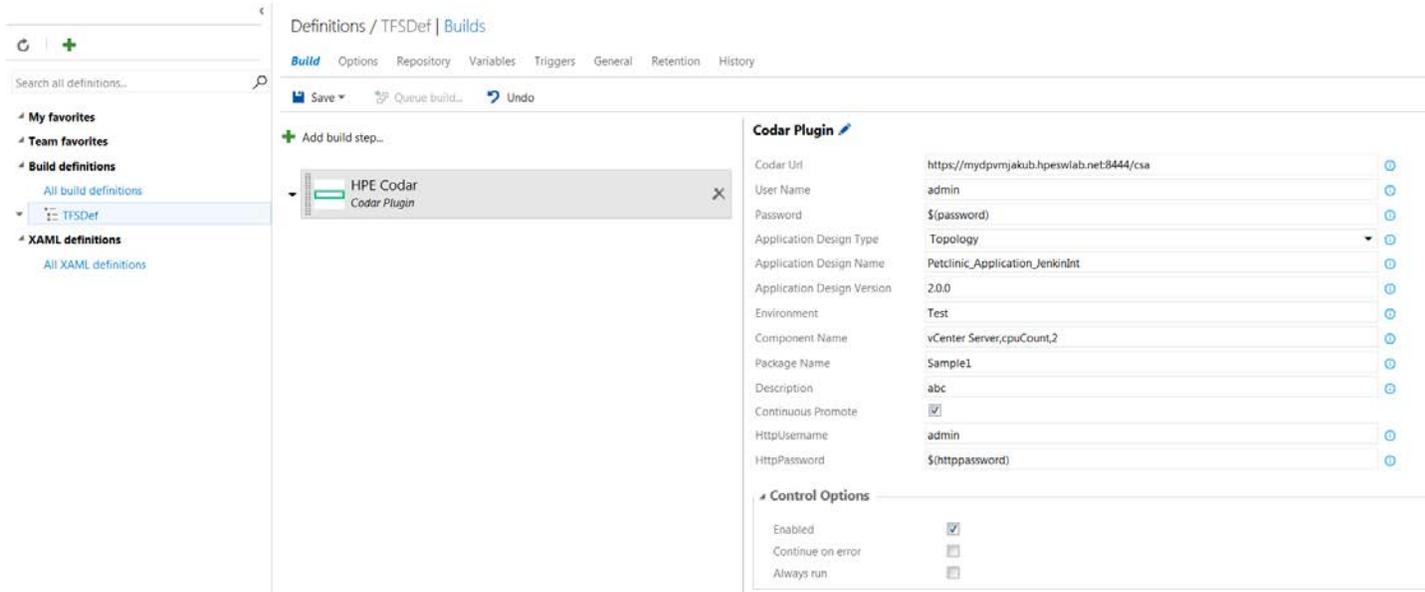


**Note:** If you want to mark the password as hidden, you may do so by adding a variable and clicking the padlock icon next to that field.

To add input parameters for the Codar plugin

1. Click 'Build' tab.
2. Enter following input parameters:
  - CodarUrl : <https://<Ip Address>:<port>/csa>
  - User Name : <Codar UserName>
  - Password : < Codar Password>: If you have declared a variable for password, it can be referenced here as \$(CodarPwd), where CodarPwd is the variable declared in the 'Variables' tab.
  - Application Design Type : <Codar Application Design Type>
  - Application Design Name: <Codar Application Design Name>

- Application Design Version: <Codar Application Design Version>
- Environment: <Codar Application Environment>. This field is mandatory only if Continuous Promote is not selected. Corresponding valid environment should exist or should be created in Codar before building the application.
- Component Name : <component Name, Property Name, Property Value>
- Package Name: <Any package name>
- Description : <Any Description>
- Continuous Promote : <True/false>
- HttpUsername: < httpusername>
- HttpPassword: < httppassword>: If you have declared a variable for password, it can be referenced here as \$(CodarHttpPwd), where CodarHttpPwd is the variable declared in the 'Variables' tab.



3. Click Save.
4. Click 'Queue build' to build the application.
5. Provide the following information:

Field	Description
CodarUrl	Host name or IP address of the server on which Codar is installed appended by port and context.  Example: https://10.2.14.12:8444/csa
Username	Codar user name  <b>Note:</b> Do not use the default administrator user because it may cause a security issue. After installing Codar and configuring LDAP, a user can be added to the Codar Integration user role or Codar Application Architect role. Use the credentials for the Codar Integration user or Codar Application Architect user here.
Password	Password of the Codar user. To encrypt this field, you need to create an environment variable in the 'Variables' tab with the name as 'password' and click the padlock icon next to this variable field. You then need to assign this variable to this password field by

	entering the '\$' sign followed by the field value enclosed within parentheses. Example: \$(password). Here, 'password' is the password value set in the 'Variables' tab.
Httpusername and httppassword	User name and password to download the artifact. This is required for Codar to pull deployment artifacts from any http location. To encrypt the http password field, you need to create an environment variable in the 'Variables' tab with the name as 'httppassword' and click the padlock icon next to this variable field. You then need to assign this variable to this password field by entering the '\$' sign followed by the field value enclosed within parentheses. Example: \$(httppassword). Here, 'httppassword' is the password value set in the 'Variables' tab.
Packagename	Package name for the Codar application
Description	Package description
Environment	Environment in which the provider to be used for deployment is contained. Environments are created in the <b>Resource provider</b> tile under Codar. For details about environments, see the Codar Administration Guide.
UUID	UUID is the OO flow Id for continuous deployment.
Component Name	<p>Specify the component properties of the design that is parameterized within the build. The component name has three parts: component name, property name, and property value.</p> <ul style="list-style-type: none"> <li>• Component name – Name of the component in the application design. The component name can be obtained from the design.</li> <li>• Property name – This is the property name and not the display name. Property name can be retrieved using the REST_API call &lt;codar url&gt;/api/codar/app-package/&lt;packageid&gt;/properties. Refer to the API Reference Guide for more information.</li> <li>• Property value – Value of the property that must be parameterized.</li> </ul> <p>For example, the package property can be:</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px 10px;">PetCline App</div> <div style="border: 1px solid black; padding: 2px 10px;">artifacturl</div> <div style="border: 1px solid black; padding: 2px 10px;">petclinic.war</div> </div> <p>You can specify multiple component names by separating each of them with a pipe sign ( ).</p>

**Note:** TFS integration does not support sequenced designs with Subscriber Component property.

# 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](mailto:clouddocs@hpe.com).

We appreciate your feedback!