

# CA Release Automation - Integrations

## JetBrains TeamCity Plug-In

Date: 20-Nov-2017





## CA Release Automation - Integrations

This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2017 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

# Table of Contents

---

JetBrains TeamCity Plug-In 5.0.9 .....	6
Supported TeamCity Versions .....	6
Supported Platforms .....	7
Supported Browser Versions .....	7
What's New .....	8
Install the Plug-In .....	8
Upgrade the TeamCity Plug-In .....	9
TeamCity Plug-In Limitations .....	9
Create a Build Step to Run the Template .....	10
Configure the Deployment Step Definition .....	10
Generic CA Release Automation Deployment Processes .....	11
Create a Release Operations Center Deployment Template .....	12
Create Artifact Types and Definitions .....	12
Configure to Run Asynchronous Deployment Plans .....	13

# JetBrains TeamCity Plug-In

# JetBrains TeamCity Plug-In 5.0.9

The JetBrains TeamCity Plug-In enables you to execute Release Operations Center deployments on multiple environments that are generated from deployment plans. The deployment plans are created from templates after a build has finished.

The TeamCity plugin always creates a new artifact package, even if the packages use existing artifact versions. If an existing artifact package name is used, an error is generated.

- [JetBrains TeamCity Plug-In 5.0.9 \(see page 6\)](#)
- [Install the Plug-In \(see page 8\)](#)
- [Upgrade the TeamCity Plug-In \(see page 9\)](#)
- [TeamCity Plug-In Limitations \(see page 9\)](#)
- [Create a Build Step to Run the Template \(see page 10\)](#)
- [Configure the Deployment Step Definition \(see page 10\)](#)
- [Generic CA Release Automation Deployment Processes \(see page 11\)](#)
- [Create a Release Operations Center Deployment Template \(see page 12\)](#)
- [Create Artifact Types and Definitions \(see page 12\)](#)
- [Configure to Run Asynchronous Deployment Plans \(see page 13\)](#)



**Important!** Only Release Operations Center 5.0 and higher deployment plans execute with the TeamCity Plug-In. To execute v4.7 processes and templates, use the v1.0 Plug-In.

## Supported TeamCity Versions

The following versions are supported:

- 2017.1 (10.1)
- 9.0.3
- 8.1.5
- 8.1.4
- 8.1.2
- 8.0.6
- 7.1.5

## Supported Platforms

- Linux
- Windows

## Supported Browser Versions

- Google Chrome - 35.0 and later
- Mozilla Firefox - 17.0 and later
- Microsoft Internet Explorer - 9 and later



**Important!** The Plug-in UI is not fully supported on Microsoft IE. For more information, see [TeamCity Plug-In Limitations \(see page 9\)](#).

When you use the Plug-In, you can implement a continuous integration solution in the following steps:

### **In Automation Studio:**

1. Create a generic process, or processes, that execute post-build actions for a new TeamCity build job.
2. Publish the processes and assign an environment to each process.

### **In Release Operations Center:**

1. Create a deployment template that executes the new Automation Studio processes.
2. Assign the server, create the parameters, and define the artifacts that complete the template creation process.

### **In TeamCity:**

1. Create a project as usual, define parameters for Release Operations Center values, such as the application and release name.
2. Configure a TeamCity build job that runs the matching Release Operations Center deployment plan template as a step after a regular build step.
3. Run the TeamCity job as usual.
4. Monitor and update the status of deployments that run on different environments.

## What's New

The following updates are made for version 5.0.9:

- Previously, when users tried to stop a build from the TeamCity plug-in, the build would be stopped within TeamCity, but not within Release Automation. This issue has been resolved.

The following updates are made for version 5.0.8:

- The issue of the generated manifest grouping two steps into one has been resolved.
- A **Timeout** field has been added. This field specifies the timeout window for the connection to content sources.  
**Default:** 120 seconds

The following updates are made for version 5.0.7:

- Support for TeamCity 2017.1 (v 10.1) added and tested for CA Release Automation 6.4 and 6.5. Note: TeamCity has switched to the common JetBrains versioning scheme of <year>.<release within year>. Thus, what was to be TeamCity 10.1 has become 2017.1.
- The error message, "Number Format Exception", displays when TeamCity triggers Release Automation.

The following updates are made for version 5.0.6:

- The http artifact API issue fixed.

The following updates are made for version 5.0.5:

- Support for TeamCity v 9.0.3 added.

The following updates are made for version 5.0.4:

- Enables Build Numbers that contain a special character (dot) be added when you run a deployment.

The following updates are made for version 5.0.3:

- Support for TeamCity v 8.1.5 added.  
The TeamCity build step continues to wait until the RA Release final state, pass, or fail, is reached.

## Install the Plug-In

To execute Release Operations Center templates after a build step is processed, install the Plug-In.

**Follow these steps:**

1. Log in to support.ca.com.

2. Click Download Center, specify the search parameters, and click GO.
3. Select the TeamCity Plug-in, and click Download.
4. Select the delivery type.



**Note:** We recommend HTTP via Download Manager.

The ZIP file is downloaded to a specified directory.

5. Extract the DeploymentPlan\_Automation\_Plugin.zip file.
6. Copy the ZIP file to the TeamCity *JetBrains\plugins* folder.
7. Stop the TeamCity service, and then restart the service.  
The plug-in is automatically installed.

## Upgrade the TeamCity Plug-In

To enable new Release Operation Center functionality, upgrade the TeamCity Plug-In.

### Follow these steps:

1. Select and download the upgrade file to a specified directory.
2. Copy the ZIP file to the TeamCity *JetBrains\plugins* folder.
3. Remove the 5.0 version ZIP file from the *JetBrains\plugins* folder.
4. Stop the TeamCity service, and then restart the service.  
The plug-in is upgraded.

## TeamCity Plug-In Limitations

- TeamCity 2017: Stopping a build does not stop the deployment in CA Release Automation. The TeamCity build status shows a continuous status of “stopping”.
- The TeamCity UI displayed in IE 8 compatible mode does not support HTML5 features. As a result, When you upload a package XML during the plug-in configuration using the IE browser, the upload will not work. The same functionality works on Google Chrome and Mozilla Firefox browser.  
**Workaround For IE browsers:** Copy the contents of the XML and paste the contents on the XML edit box on the plug-in UI.

- When you generate package XML during plug-in configuration, the number of artifacts versions you can specify is limited to 5. This is not a limitation with the upload package XML option.
- You cannot assign the RA template property to non-string deployment parameters.

## Create a Build Step to Run the Template

To complete the TeamCity build step configuration, create the matching Release Operations Center template build step.

**Follow these steps:**

1. In TeamCity, select Administration.
2. Select Create build configuration.
3. Type the name for the build configuration.
4. Click the VCS settings.
5. Click Add build step.
6. Select Run RA Deployment in the Runner type list.



**Note:** If Run RA Deployment is not selected in the Runner type list, the Plug-In installation fails.

7. Complete the New Build Step configuration options.

A build step is created in TeamCity to run the template.



**Note:** All boxes that are labeled with a red asterisk are required. For the boxes with a question mark icon, click the icon for Tool-Tip information

## Configure the Deployment Step Definition

The Plug-In provides the option to select a running a CA Release Automation template when you add a TeamCity build step.



**Note:** When you plan to configure a build step to run a CA Release Automation template:

- The Run RA Deployment process is run after a TeamCity build step.
- Multiple Run RA Deployment build steps are configured in the TeamCity job.

During the build step creation, the input boxes have a parameters icon to the right of the box. Click the icon, and a parameter list appears.



**Note:** The following data items are mandatory:

- Host Name
- Port
- User Name
- Password
- Application Name
- Project Name
- Deployment Template Category Name
- Deployment Template Name
- Deployment Environments
- Deployment Stage to Execute

In the definition of a Run RA Deployment step, you can use the TeamCity generated Build parameters in the format: `#{build}`.

In order for the REST API calls to execute successfully the entered user must have the following permissions:

- Release Template Designer at the application level
- Release Designer in the target environment

## Generic CA Release Automation Deployment Processes

Generic CA Release Automation deployment templates are deployment definition containing preparatory process steps to run for a deployment.

Create the deployment processes in Automation Studio and a matching execution template in Release Operations Center before configuring a TeamCity Post-build Actions template.

Automation Studio requires the following steps to create a process that can be executed from a Release Operations Center template:

- Each process must be assigned to the application environment.
- Each process must be published.

## Create a Release Operations Center Deployment Template

Create the Release Operations Center generic process template with the release definitions and preparatory steps that are related to a TeamCity post-build action.

### Follow these steps:

1. Create a deployment template in Release Operations Center that executes the generic deployment processes created in Automation Studio. Each process is executed as a unit in a template step.
2. Set the Agents for each server type in the template.
3. Set the template status to Active.



**Note:** To create a generic template, use Automation Studio user input parameters, values, and template properties when possible. In the TeamCity Post-build Action configuration, you can define new values for properties.

## Create Artifact Types and Definitions

In Release Operations Center, you can create artifact types and definitions that are based on the build job output artifacts. The Plug-In creates an artifact package using the artifacts and associates it with the deployment plan.

### Follow these steps:

1. Click **Artifact Management**, and click **New**.

2. Create the artifact type, and specify a **Name**.
3. Add the **Components** and **Attributes**, and click **Save**.  
The artifact and definitions are created.

## Configure to Run Asynchronous Deployment Plans

To run deployment plans simultaneously, configure both the TeamCity Plug-In and Automation Studio. This configuration enables the deployment to run without stopping due to a block or failure. The default settings do not enable this function if there are dependencies.

### Follow these steps.

1. In the TeamCity Plug-in, select **Run Deployment Plans Asynchronously**, and click **Save**.
2. In Release Operations Center, click **Designer, Application Model**.
3. Select the **Application**, and click the **Server Types** tab.
4. Mouseover the server type and click **Edit Server Type**.
5. Clear the **Requires exclusive execution** check box and click **Save**.
6. Click **Releases, Deployment Plan by Project**, and double-click the **Deployment Plan**.
7. Click **Deploy**.  
Deployment plans continue to run simultaneously.  
For more information, see [Create Processes](#) in the CA Release Automation documentation.