

## Serena Deployment Automation plugin for TeamCity

- Install the SerenaDeploy-x.x.zip in to TeamCity using the Administration menu
- Go to an existing Build Configuration and add a new Build Step selecting Serena Deploy plugin.
- Enter the parameters for SDA (see screenshots as an example).
- In SDA, select None as the source type for the component you are uploading to as the plugin resides in TeamCity
- The plugin will optionally upload a new version into SDA using the built artefacts.
- The plugin will optionally create component version properties to the uploaded version.
- The plugin will optionally apply a status to the uploaded version.
- The plugin will optionally execute a process to deploy it.
- A parameter can be passed to the plugin to determine if publishing should occur. For example create a parameter "PublishToSDA" as below and then in the plugin parameter "Publish Version if" reference this parameter, i.e. %PublishToSDA%.

**Edit Parameter** [X]

**Name: \*** PublishToSDA

**Kind:** Configuration parameter

**Value:** false [icon]  
Type '%' for reference completion

**Spec:** [Edit...]  
checkbox checkedValue='true' description='Publish the outputs of the build to SDA' uncheckedValue='false' label='Publish Build to SDA' display='normal' [icon]  
Defines parameter control presentation and validation. Parameter specification format is `typeName[ key='value'[ key2='value2']]`, ServiceMessage syntax is used.

[Save] [Cancel]

- A parameter can be passed to the plugin to determine if deployment should occur. For example create a parameter "DeployFromSDA" as below and then in the plugin parameter "Deploy to Version if" reference this parameter, i.e. %DeployFromSDA%.

**Edit Parameter**

**Name: \***

**Kind:**

**Value:**  Type '%' for reference completion

**Spec:**

```
checkbox checkedValue='true' description='Deploy a specific build (usually the current one) from SDA. The build needs to have been published first.' label='Deploy from SDA' display='normal'
```

Defines parameter control presentation and validation. Parameter specification format is `typeName[ key='value'[ key2='value2']]`, ServiceMessage syntax is used.

- Deployment can be carried out in a separate step if required, however the component name and version name need to be set to ensure that deployment is initiated.

## Example Setup (1/2):

Deployment Server Settings:	
Serena DA URL: *	<input type="text" value="http://localhost:8080/serena_ra"/> <small>The URL for Serena DA, i.e. http://localhost:8080/serena_ra</small>
Username: *	<input type="text" value="admin"/> <small>The user name used to login to the Serena DA server.</small>
Password: *	<input type="password" value="*****"/> <small>The password used to login to the Serena DA server.</small>
Version Upload Settings:	
Publish Version:	<input checked="" type="checkbox"/> <small>Check here if you want to upload the outputs of this build into Serena DA.</small>
Publish Version if:	<input type="text" value="%PublishToSDA%"/> <small>Use a property to determine if the outputs of the build should be uploaded to Serena DA. If set this overrides the "Publish Version" checkbox.</small>
Component: *	<input type="text" value="online-web"/> <small>The name of the component in the Serena DA server which will receive the new version.</small>
Base Artifact Directory: *	<input type="text" value="%teamcity.build.workingDir%"/> <small>Specify the base directory where the artifacts are located. If empty the default value is %teamcity.build.workingDir% which is the base of the build workspace.</small>
Directory Offset:	<input type="text" value="target"/> <small>Specify a sub-directory in the workspace where to search for files to publish. Absolute directories will not work.</small>
Version Name:	<input type="text" value="%SDAVersion%"/> <small>The name of the new version that will be created in the Serena DA server. You can reference TeamCity parameters, e.g. %build.number%</small>
Includes:	<input type="text" value="*.war,context.xml,README"/> <small>A comma separated list of file filters to select the files to publish. If empty the default includes everything in the workspace.</small>
Excludes:	<input type="text"/> <small>A comma separated list of file filters to exclude from publishing. Leave blank to include everything.</small>
Version Properties:	<div><div>Enter properties below:</div><div><pre>build.number=%build.number% build.vcs.number=%build.vcs.number%</pre></div></div> <small>Newline separated list of quoted properties and values i.e. prop1=%teamcity.project.id% to create on the version.</small>
Version Status Settings:	
Add Status to Version:	<input checked="" type="checkbox"/> <small>Check here if you want to add a status to the version in Serena DA once it's uploaded.</small>
Version Status Name:	<input type="text" value="Functional Testing Complete"/> <small>The name of the status to apply to the version.</small>

## Example Setup (2/2):

Version Deployment Settings:	
Deploy Version:	<input checked="" type="checkbox"/> <small>Check here if you want to trigger a deployment of this version in Serena DA once it's uploaded.</small>
Deploy Version if:	<input type="text" value="%DeployFromSDA%"/> <small>Use a property to determine if a deployment of this version should be triggered in Serena DA once it's uploaded. If set this overrides the "Deploy Version" checkbox.</small>
Deploy Application:	<input type="text" value="Ring2Park Online"/> <small>The name of the application in Serena DA which will be used to deploy the new version.</small>
Deploy Environment:	<input type="text" value="Ring2Park-INT"/> <small>The name of the environment in Serena DA to deploy to.</small>
Deploy Process:	<input type="text" value="R2P Online - Deploy Web"/> <small>The name of the application process in Serena DA which will be used to deploy the new version.</small>
Deploy Properties:	<div><div>Enter properties below:</div><div><pre>prop1=%DeployFromSDA% prop2=%build.number%</pre></div></div> <small>Newline separated list of quoted properties and values i.e. prop1=%teamcity.project.id% to pass to Serena DADeployment Process.</small>

## Example Successful Build Output:

```
[13:56:39] - Step 2/2: Serena Deploy (Serena Deploy) (11s)
[13:56:39]   - [Step 2/2] Retrieving configuration
[13:56:39]     [Retrieving configuration] Serena DA URL: http://localhost:9090/da
[13:56:39]     [Retrieving configuration] Username: admin
[13:56:39]     [Retrieving configuration] Publish Version: true
[13:56:39]     [Retrieving configuration] Publish Version if: not set
[13:56:39]     [Retrieving configuration] Component: teamcity-test
[13:56:39]     [Retrieving configuration] Base Directory: C:\Program Files (x86)\JetBrains\TeamCity\buildAgent\work\cd3ac3818570ad6d
[13:56:39]     [Retrieving configuration] Directory Offset: target
[13:56:39]     [Retrieving configuration] Version Name: 33
[13:56:39]     [Retrieving configuration] Include Patterns: *.war
[13:56:39]     [Retrieving configuration] Version Property: teamcity.buildId = 33
[13:56:39]     [Retrieving configuration] Add Files to Existing Version: is true
[13:56:39]     [Retrieving configuration] Deploy Version: true
[13:56:39]     [Retrieving configuration] Deploy Version if: not set
[13:56:39]     [Retrieving configuration] Deploy Application: TeamCity Test
[13:56:39]     [Retrieving configuration] Deploy Environment: Integration
[13:56:39]     [Retrieving configuration] Deploy Process: Deploy
[13:56:39]     [Retrieving configuration] Deploy Property: teamcity.buildId = 33
[13:56:39]     [Retrieving configuration] Add Status to Version: is true
[13:56:39]     [Retrieving configuration] Status Name: BUILT
[13:56:39]     [Retrieving configuration] Using component id: b2da8761-5d89-45c4-a42a-55de5d8b8309
[13:56:39]   - [Step 2/2] Uploading Version (10s)
[13:56:39]     [Uploading Version] Component version "33" already exists with id: 8bb38190-4f2a-4ed3-887e-a28382c0dbcb
[13:56:39]     [Uploading Version] Adding files to existing version "33"
[13:56:41]     [Uploading Version] Created staging directory: 82287423-34b2-4834-8b1c-57d8f5c71e89
[13:56:49]     [Uploading Version] Added 1 files to staging directory...
[13:56:50]     [Uploading Version] Committing change set...
[13:56:50]     [Uploading Version] Created change set: SHA-256{57ceca5e63abb88c199ad63fcdf27b2a57c019ce8d2375b27e02c293ae1d9e41}
[13:56:50]     [Uploading Version] Labeling change set with label: 33
[13:56:50]     [Uploading Version] Done labeling change set!
[13:56:50]     [Uploading Version] Deleted staging directory: 82287423-34b2-4834-8b1c-57d8f5c71e89
[13:56:50]   - [Step 2/2] Adding properties to Version
[13:56:50]     [Adding properties to Version] Found component version property sheet id: 440c5a0f-ad98-4036-bb0f-142b7743cda3
[13:56:50]     [Adding properties to Version] Calling URI "http://localhost:9090/da/property/propSheet/components%26b2da8761-5d89-45c4-a4"
[13:56:50]   - [Step 2/2] Adding Status to Version
[13:56:50]     [Adding Status to Version] Applying status "BUILT" to Version 33
[13:56:50]     [Adding Status to Version] Calling URI "http://localhost:9090/da/rest/deploy/version/8bb38190-4f2a-4ed3-887e-a28382c0dbcb/"
[13:56:50]   - [Step 2/2] Deploying Version
[13:56:50]     [Deploying Version] Starting deployment of TeamCity Test to Integration
[13:56:51]     [Deploying Version] Deployment request URI is: http://localhost:9090/da/app#/application-process-request/e59ee3b7-f998-4ba
[13:56:51] + Publishing internal artifacts
[13:56:51]   Build finished
```