

Upgrade Notes

Changes from 2017.2.3 to 2017.2.4

The Inspections (.NET) and Duplicates Finder (.NET) build steps were renamed to Inspections (ReSharper) and Duplicates finder (ReSharper)

Changes from 2017.2.2 to 2017.2.3

Build revisions

In all the build configurations, the builds run before the upgrade will not be reused in the chains and will run anew (only the last build on the default branch is affected if branches are used). This can also result in a clean checkout in the first run build for VCSs like Perforce. The behavior is similar to that after VCS root editing.

Security

It is recommended to add "`teamcity.artifacts.restrictRequestsWithArtifactReferer=true`" [internal property](#) to enhance security of the server.

Changes from 2017.2.1 to 2017.2.2

Known issues

(Fixed 2017.2.3) If you use the Artifacts plugin and get the "Invalid RSA public key" browser message on opening build step settings, please apply the [workaround](#).

Under Windows, when TeamCity server is started as a service, "logs\teamcity-winservice.log" file is not created and server startup errors are nowhere to be seen. [Details](#)

IDE Plugins

It is highly recommended to update the IDE plugins for all users to the latest version and then add the "`teamcity.uploadPersonalPatch.requireAuthorization=true`" [internal property](#) to enhance security of the server.

Perforce VCS Root executable paths

Since TeamCity 2017.2.2, the field which specifies the path to p4 works only on the agent side, for agent-side checkout.

For the server, the p4 binary should be present in the PATH of the TeamCity server (or can be specified via the `teamcity.perforce.customP4Path` [internal property](#)).

The `teamcity.perforce.p4PathOnServerWhitelist` [internal property](#) can be used to specify a semi-colon-separated list of allowed p4 paths. The paths from this list can be set in VCS Root p4 path parameter for the server side (to restore old behavior).

Mercurial VCS Root properties

Since TeamCity 2017.2.2, a number of Mercurial VCS root properties change their behavior for security reasons.

- the "HG command path" is used on the TeamCity server only if included into the [whitelist](#)
- the "Clone repository to" property is hidden if the VCS root doesn't have it already and is ignored by default. To make TeamCity display the property in all VCS roots, add the `teamcity.hg.showCustomClonePath=true` [internal property](#). The value of the VCS root property is respected only if it is included into the whitelist specified by the `teamcity.hg.customClonePathWhitelist` [internal property](#), which is a semi-colon-separated list of directories where a clone is allowed. Use `/path/to/dir/*` to allow clones to the child directories of the `/path/to/dir`.
- the "Mercurial config" is ignored on the server. If you need to enable some Mercurial plugins, please do that in the global `.hgrc` on the TeamCity server machine.

Changes from 2017.2 to 2017.2.1

Kotlin DSL changes

The versions in pom.xml were updated: `kotlin.version` is updated to 1.2.0, `teamcity.dsl.version` is updated to 2017.2.1. The dependency on `kotlin-stdlib` is replaced with the dependency on `kotlin-stdlib-jdk8` in order to provide access to additional functionality available in jdk8 (e.g. named groups in regexps). The dependency on the deprecated `kotlin-runtime` and the redundant dependency on `kotlin-compiler-embeddable` were dropped.

Now TeamCity provides a parent maven project for Kotlin DSL which defines the `teamcity.dsl.version` and `kotlin.version` properties. With such a parent project, you will not have to update your pom.xml after each TeamCity upgrade.

The easiest way to apply these changes is to run the 'Download settings in Kotlin format' action in project admin area and use the pom.xml from the zip produced by the TeamCity server.

Bundled Java used in Docker images

The bundled Java used in Docker images has been updated to 8u151.

Changes from 2017.1.x to 2017.2

Known issues

(Fixed 2017.2.1) TFS in Java working mode (when Team Explorer is not installed on the machine) report "TFS subsystem was destroyed" errors. See [TW-52685](#) for details.

Upgrading using Windows installer can take significant time if your TeamCity installation directory contains lots of nested directories (e.g. TeamCity Data Directory is under it). The long stage can occur after "Extract: Uninstall.exe..." progress message. In you encounter this long step, please wait for the completion of the operation (the installer runs `icacls.exe` utility as a nested process). To prevent the issue it is recommendd to [move the data directory](#) out of TeamCity server installation home.

Perforce branch specification change

There is a breaking change which requires your action if you use Perforce streams with enabled feature branches and you're using a non-default branch filter.

Starting from TeamCity 2017.2, Perforce VCS Roots use the same format for Perforce streams and TeamCity feature branches specification.

In the VCS Root branch specification for Perforce, `+:stream_name` must now be replaced with `+://stream_depot/stream_name`. Also, for better presentation of stream names in the UI, you may want to replace the default branch specifications like `+:* with +://your_stream_depot/*.`

This change was made in the scope of fixing [TW-48038](#).

Server process restart

Now if the server process is stopped unexpectedly or killed, the process will automatically restart. The server should be stopped using the `teamcity-server.bat/sh stop` command which performs a graceful stop.

Bundled plugins

Docker integration plugin

The Docker integration plugin is bundled since TeamCity 2017.2.x. If you installed the plugin for the previous version manually, please [remove it](#).

.NET CLI plugin

The .NET CLI (.NET Core) plugin is bundled since TeamCity 2017.2.x. If you installed the plugin for the previous version manually, please [remove it](#).

During upgrade all existing .NET Core build steps will be converted into .NET CLI steps and existing .NET Core plugin will be

disabled.

Note: The `DotNetCore` and `DotNetCore_Path` agent configuration parameters will be changed to `DotNetCLI` and `DotNetCLI_Path`; please consider updating your agent requirements which depend on these parameters.

REST API

REST API uses version 2017.2. The previous versions of the API are still available under `/app/rest/2017.1` (`/app/rest/10.0`), `app/rest/9.1`, `/app/rest/9.0`, `/app/rest/8.1`, `/app/rest/7.0`, `/app/rest/6.0` URLs.

buildType entity

has "templates" sub-element now instead of "template" to support multiple templates.

build entity

No longer expose boolean "running" attribute, textual "state" attribute with value "running" is used instead.

Windows Versions Support

Windows XP and Vista are no longer the supported versions of Windows for the TeamCity Server and Agent. While the server and agent can still work on these old versions, we do not target the versions during our development. [Let us know](#) if the support for the versions is important for your TeamCity usage.

J2EE Servlet 2.5 container is no longer supported

J2EE Servlet container version 2.5 is not supported since TeamCity 2017.2. TeamCity does not guarantee support for Tomcat 6.x and Jetty 7.x implementing Servlet 2.5. For .war distribution (not recommended, .tar.gz distribution is recommended), TeamCity supports Apache Tomcat 7+, J2EE Servlet 3.0+ and JSP 2.2+.

Other

- The bundled Tomcat 8.5. restricted usage of special characters in the URL including curly bracket symbols (`{ }`). [Details](#)
- TeamCity integration with IntelliJ-based IDEs no longer supports StarTeam and Visual Source Safe version controls.

Changes from 2017.1.4 to 2017.1.5

The bundled JetBrains dotCover has been updated to version 2017.2

The SSH Agent build feature started to report a build problem if it fails to start an SSH agent with the specified SSH key (in the scope of [TW-42707](#)). Previously errors were only logged, but not reported as build problems. As a result, builds with invalid SSH Agent settings would start to fail after the upgrade.

Changes from 2017.1.3 to 2017.1.4

Known issues

TFS Personal support lists all build configurations for TFVC VCS root. See [TW-51497](#) for details.

Changes from 2017.1.2 to 2017.1.3

[TW-50148](#) was fixed and the DSL API documentation was improved. If you need these changes for local development, please update the [maven dependency version](#) to 2017.1.3.

Now TeamCity server runs 'git gc' automatically to improve performance of git operations. This requires a git client to be installed on the server and be available to the server via the PATH environment variable. If a native git client cannot be found, then the corresponding health report is shown. For TeamCity to find the git client, the client needs to be installed on the server machine and added to `$PATH` (the server restart is required afterwards). Instead of modifying PATH, the path to the git client can be specified via the `teamcity.server.git.executable.path` [internal property](#).

Changes from 2017.1.1 to 2017.1.2

No noteworthy changes.

Changes from 2017.1 to 2017.1.1

The bundled IntelliJ IDEA has been updated to 2017.1.2

Changes from 10.0.x to 2017.1

Known issues

Editing cloud profile cancels all builds on profile agents. See [TW-49616](#) for details.

TeamCity Versioning Changes

Since 2017, TeamCity adopts the common JetBrains versioning scheme that identifies versions by year following the pattern: "<year>.<number of the feature release within the year>.<bugfix update number>". The current version is TeamCity 2017.1 formerly known as TeamCity 10.1.

Update settings in Kotlin DSL

In this version TeamCity settings format has been changed and if your settings are stored in Kotlin DSL, you might need to [update Kotlin DSL scripts](#) before continuing to use them. Check any related server health reports after the server upgrade.

CSRF Protection: Modifying GET requests and Proper Proxy configurations

TeamCity now implements [CSRF protection](#) to improve web UI security and this introduces several changes in behavior which might affect your installation. In particular:

- If you use a reverse proxy before TeamCity, the proxy should not change the original "Host" request header (this typically requires configuring the proxy to set the Host header to the original request value). Also, it should not modify the "Origin" and "Referer" headers present in the original request. While this has been the recommended setup for a long time, now it becomes critical for the TeamCity web UI functioning;
- if you use non-bundled clients which perform HTTP GET requests to TeamCity, some of the GET requests (those which change the state of the server, like `http://server/action.html?add2Queue=XXX`) stop working in 2017.1, please change the requests to use POST instead of GET;
- the non-browser clients which reuse authentication by supplying the TCSESSIONID cookie with the request, need to be updated to supply the "Origin" HTTP header with the value the same as the host the request is being sent to.

If the check fails, you get the HTTP 403 response with the details of the failed check. The details are also logged into `teamcity-auth.log`.

Old IPR Runner

The old (TeamCity 6.0) IPR runner has been removed from the TeamCity. It was deprecated and has not been offered as an option since TeamCity 6.0, and now it is gone completely (the corresponding build configurations will not run anymore).

Log4j configuration

It is recommended to overwrite the server's `conf\teamcity-server-log4j.xml` file with the content of the `conf\teamcity-server-log4j.xml.dist` file which represents the default logging configuration which has changed in this release. If you do need a custom logging configuration, consider using [logging presets](#) instead of modifying `conf\teamcity-server-log4j.xml`.

The same overwriting from the sibling `.dist` file is recommended for the TeamCity agents.

The `conf\teamcity-*-log4j.xml.dist` file is created after the first start of the upgraded TeamCity version.

Builds metadata storage (NuGet feed)

Builds metadata storage will be re-created and builds will be reindexed right after the upgrade. As a result, immediately after

the upgrade, the TeamCity internal NuGet feed will not contain all of the packages. During the builds reindexing, the following messages may appear in the `teamcity-server.log`:

```
INFO - .index.BuildIndexer (metadata) - Enqueued next 100 builds for indexing, builds left: 2000, last build id: 3813
INFO - .index.BuildIndexer (metadata) - Enqueued next 100 builds for indexing, builds left: 1900, last build id: 3713
```

"builds left:" indicates the number of builds left to process. Note that TeamCity starts reindexing from the most recent builds, so all fresh builds should appear in the TeamCity NuGet feed in a relatively short time.

To increase the metadata indexing speed you could use the [following tip](#).

REST API

REST API has only minor changes, so the same API is exposed under the `app/rest/10.0` and `/app/rest/2017.1` URLs. API version has been updated to 2017.1 though to reflect the changes. The build's node "triggeredBy" now has more correct values of "type" attribute for the builds started after 2017.1 upgrade. In particular, the "buildType" value is not used anymore, the "finishBuild", "snapshot", etc. values are used instead.

Visual Studio Add-in fails to install from TeamCity UI

As a workaround could be used [ReSharper web installer](#). See [TW-51680](#) for details.

Changes from 10.0.4 to 10.0.5

If you are using TFS with agent-side checkout, note that due to the fix of [TW-48555](#) TeamCity will have to re-create TFS workspaces, which may result in a clean checkout on the agents after the upgrade.

Changes from 10.0.3 to 10.0.4

Precedence of %dep.ID.NAME% parameter references

When `%dep.ID.NAME%` is used and there several different builds accessible via (direct or indirect) artifact dependencies, the result of the reference resolution could have used any of the builds without any guaranteed precedence.

Since 10.0.4 `dep.` parameter resolution works as follows:

1. if there is a snapshot dependency, the build from the same chain wins.
2. if there is no snapshot dependency and several builds are accessible via an artifact dependency, the build with a greater `buildId` wins. If there are several artifact dependencies from a single build configuration, only the first one is considered.

Updates

AWS SDK has been updated to 1.11.66 to support new instance types (r4.4xlarge, f1.16xlarge, t2.2xlarge, t2.xlarge, r4.2xlarge, r4.xlarge, r4.large, r4.16xlarge, r4.8xlarge, f1.2xlarge).

Changes from 10.0.2 to 10.0.3

Amazon EBS-Optimized Instances

The behavior of [EBS-optimization](#), enabled by default since TeamCity 10.0, is changed similarly to what EC2 console offers:

- 1) EBS-optimization is turned on by default for c4.*, m4.* and d2.* (non-configurable)
- 2) EBS-optimization is turned off by default for any other instance types.
- 3) EBS-optimization can be turned on for instances that support it (such as c3.xlarge, etc.) by checking the appropriate box when configuring the image of the Amazon cloud profile

Bundled tools updates

The bundled dotCover has been updated to version 2016.2.2

Changes from 10.0.1 to 10.0.2

- The known issue mentioned for 10.0.1 is fixed.
- The bundled JetBrains dotCover has been updated to version 2016.2.
- Jabber integration is now [more restrictive](#) with regard to SSL connection checking. If you are using jabber.org for sending notifications, and face a problem regarding SSL certificate, you should either enable "Use legacy SSL" option or (better) change JVM version for running TeamCity to at least 1.8.0_101.
- Precedence of %dep.ID.NAME% parameters resolution was changed unintentionally in case of several different builds used as dependency. See [TW-47518](#) for details.

Changes from 10.0 to 10.0.1

All known issues mentioned for 10.0 are fixed.

Known Issues

(fixed in 10.0.2) TeamCity server temp folder can fill up if an [agent tool](#) is installed as a directory in `<TeamCity Data Directory>/plugins/.tools`. [Details and workaround](#).

Changes from 9.1.x to 10.0

Known Issues

(These known issues are fixed in 10.0.1)

Failed to collect TFS changes - From version `<x>` is greater then current version `<y>`

If you use TFS version control and get "Error collecting changes for VCS repository ... Failed to collect TFS changes - From version `x` is greater then current version `y`" error, either commit a new change so that it appears in the affected build configuration, or install an [updated plugin](#) (related [issue](#)).

Project administrator may not be able to re-define parameters inherited from the parent project

See request [TW-46372](#) for details and possible workaround.

Upgrade from TeamCity version before 8.1 fails with "Can't take exclusive lock when db lock is not held"

See request [TW-46385](#) for details and possible workaround.

Subversion VCS roots with `svn+ssh://` protocol can report "Host key (xxx) can not be verified."

See request [TW-46489](#) for details and for the plugin with the fix

Changes in agent properties reporting .NET 4.x runtime

TeamCity agents before version 10 used to report `DotNetFramework4.0_*` properties whenever any of 4.x versions of .NET framework were installed on the agent. Since TeamCity 10, `DotNetFramework4.0_*` properties are reported only when 4.0 runtime (without updates) is installed. For 4.5.*, 4.6.* updates corresponding `DotNetFramework4.N_*` properties are reported. This change in behavior allows for more precise requirements definition.

If after the upgrade you get incompatible agents with Unmet requirement: `Exists=>DotNetFramework4.0_x86(/x64)` exists message, review the explicit requirements in your build configuration. If your build is compatible with any .NET 4.x runtime (it is a most common case) please use `Exists=>DotNetFramework4.*_x86(/x64)` requirement. If you would like to run on .NET 4.5+ agents - use `Exists=>DotNetFramework4.(5|6).*` requirement.

Some third-party plugins are known to be affected. On the upgrade, make sure to upgrade xUnit plugin to version 1.1.2+

Ignored tests table optimization

During upgrade TeamCity will optimize data in the `ignored_tests` table (we do that in order to speedup the TeamCity built-in backup / restore process). In some rare cases, when this table contains millions of rows, the process of table optimization may take significant time - possibly a few hours. Among other data, the table contains the reasons why a particular test in a build was marked as ignored. If this information for old builds is not very important for you, you can start TeamCity server with the additional [JVM option](#):

```
-Dteamcity.truncateIgnoreReasonConverter.copyReasons=false
```

In this case TeamCity will not copy the ignore reasons into the new, optimized table, and this particular step of the upgrade process will run much faster.

Java 8

Starting from TeamCity 10, the TeamCity server [requires Java 8](#) JRE/JDK (included in the Windows `.exe` distribution).

TeamCity agents currently require Java 1.6+, but starting from the next TeamCity version, the minimum requirement for [the Java on the agent](#) will be Java 8 (included in agent's Windows `.exe` distribution). It is recommended that you now consider upgrading the agents Java.

Java memory options change

It is recommended to remove the "`-XX:MaxPermSize=...`" [JVM option](#) from `TEAMCITY_SERVER_MEM_OPTS` environment variable, if previously configured. (This is due to the fact that Java 8 [does not use](#) permanent generation (PermGen) anymore)

Agent requirements and artifact dependencies disabling

Agent requirements and artifact dependencies can be disabled now. TeamCity plugins and REST API- based code using a version of API prior to TeamCity 10 is likely to ignore the disabled status of these settings.

TFS

TeamCity comes with cross-platform TFS integration: to work with TFS, you no longer need to install a TeamCity server on a Windows machine.

Visual Studio Online Work Items plugin

Visual Studio Online Work Items plugin is obsolete since TeamCity 10.0 and can be safely removed. TeamCity 10.0 has a built-in integration with [Team Foundation Work Items](#) which supports TFS 2010+ and Visual Studio Team Services. After upgrade, TeamCity will detect the existing issue tracker connections of this plugin and convert them into TFS Work Items.

Default Checkout mode for newly created build configurations

The default setting for the VCS checkout mode on creating new build configurations has changed: now TeamCity will check out the sources on the agent before the build. If the agent-side checkout is not possible, TeamCity will use the server-side checkout. Explicit server-side or agent-side checkout is still in place. The new default applies only to newly created build configurations; all the existing ones will work as configured before.

Project-based Agent Management Permissions

New TeamCity installations now have different agent management permissions assignments: Project Administrator role does not include (global) Agent Manager role. Instead, Project administrator role has [agent-project permissions](#) which allow to manage agents from the agent pools with only projects where user has Project Administrator role.

Existing installations are not affected by this change in order not to change the user permissions. However, it is recommended to review the Project Administrator role and consider excluding "Agent Manager" role and adding the following permissions:

- Enable / disable agents associated with project
- Start / Stop cloud agent for project
- Change agent run configuration policy for project
- Administer project agent machines (e.g. reboot, view agent logs, etc.)
- Remove project agent

- Authorize project agent

UI Changes

Server Administration UI

The new Administration | Tools page allows setting up tools to be used by appropriate plugins. Tools are automatically distributed to all build agents and can be used in related runners.

New Create project / Create build configuration buttons

The new Create subproject and Create build configuration buttons have a drop-down now allowing you to select whether you want to create a project from scratch (manually), from URL, or using the popular version control systems GitHub.com and Bitbucket.

NuGet-related UI

NuGet settings page is removed. NuGet.exe can be installed using the new Tools page; to set up TeamCity as a NuGet Server, go to the Administration | NuGet Feed page.

Tests-related UI

The Problematic Tests tab is no longer available and the View all tests failed within the last 120 hours link is removed from the Current Problems tab.

TeamCity now detects [Flaky tests](#) displayed on the dedicated tab for a given project.

Visual Studio Add-in

The legacy version of the TeamCity Visual Studio Add-in is no longer supported. Visual Studio 2005 and 2008 are not supported.

TeamCity Visual Studio Add-in is shipped [as a part of ReSharper Ultimate](#). After installation, the TeamCity Add-in will be available under the RESHARPER menu in Visual studio.

Note that the installer will remove the pre-bundle products versions: TeamCity and ReSharper versions prior to 9.0, dotCover prior to 3.0, dotTrace prior to 6.0.

ReSharper Ultimate does not support the Visual Studio versions 2005 and 2008.

IntelliJ IDEA Compatibility

IntelliJ IDEA 12.1 and older as long as other IntelliJ-based products released prior to year 2013 no longer supported by [IntelliJ Platform Plugin](#) .

Snapshot dependencies builds rebuilding

After the server upgrade, the builds used as snapshot dependencies may be rebuilt once even if the snapshot dependency has the "Do not run new build if there is a suitable one" option set to ON. This is done to fix the [issue](#).

Perforce

Clean checkout will be enforced in builds with Stream-based and Client-based Perforce VCS Roots.

Subversion

Starting from TeamCity 10, TeamCity does not accept by default connections to SVN servers accessed by https:// protocol with non-trusted server SSL certificate. To enable access with such certificates, you should either import the certificate to server JVM keychain, or enable VCS Root option "Accept non-trusted SSL certificate" (Enable non-trusted SSL certificate in 10.0) ([issue](#)).

Bundled tools updates

Ant runner: the bundled Ant distribution has been upgraded from 1.9.6 to 1.9.7

.NET dotCover coverage: the bundled dotCover is updated to 2016.1

ReSharper command line tools: the bundled R# CLT is updated to 2016.1

Java inspections and duplicates: the bundled IntelliJ IDEA is updated to 2016.2

GitHub Issue Tracker

If you were using the TeamCity-GitHub [third-party plugin](#) prior to TeamCity 10.0, you can safely [remove](#) it: the built-in TeamCity integration will detect the existing connection to GitHub issue tracker and pick up your settings automatically.

NuGet Support

Configuration parameters `teamcity.tool.NuGet.CommandLine.%NUGET_VERSION%.nupkg` are not reported anymore. `teamcity.tool.NuGet.CommandLine.%NUGET_VERSION%` parameters should be referenced instead.

e.g instead of using `%teamcity.tool.NuGet.CommandLine.DEFAULT.nupkg%` parameter reference `%teamcity.tool.NuGet.CommandLine.DEFAULT%` should be used.

Build Statistics

Several statistic values (metrics) has been reworked and renamed:

BuildCheckoutTime into `buildStageDuration:sourcesUpdate`

BuildArtifactsPublishingTime into `buildStageDuration:artifactsPublishing`

ArtifactsResolvingTime into `buildStageDuration:dependenciesResolving`

Old keys are still supported in charts definitions.

REST API

REST API uses version 10.0. The previous versions of the API are still available under `/app/rest/9.1`, `/app/rest/9.0`, `/app/rest/8.1`, `/app/rest/7.0`, `/app/rest/6.0` URLs.

Requests for a set of items with the locator addressing a single item which resulted in 404 responses previously will now return an empty set as a more consistent approach. For example, `.../app/rest/builds?locator=id:<non-existent build id>`. REST debug logging might have diagnostics message with more details as to the case.

Requests for set of items can return not all/incomplete results (with zero or more items included) and provide `"nextHref"` sub-element with the link to retrieve the next "page" of items. The search result is complete when no `"nextHref"` sub-element is provided.

Some requests for set of items (e.g. `.../app/rest/vcs-roots` and `.../app/rest/vcs-root-instances`) will use paged results by default when queried without a locator (they used to list all the items). Add the `"count:NNN"` locator dimension to set page size.

Finding builds (`.../app/rest/builds/... URL`)

When performing builds scan to find those matched by the locator specified, by default for performance reasons TeamCity will return partial result limited by scanning only 5000 most recent builds. To process a larger portion of the history, check the `"nextHref"` attribute returned or set the `"lookupLimit"` locator dimension to a larger value.

Previously, until specifically requested the builds from non-default branch as well as canceled, personal and failed to start builds were not returned. Now these filtered out builds are returned by default for running and queued builds queries as well as when filtering by agent or user. Use `"defaultFilter:true/false"` locator dimension to manage the default filtering explicitly. Also, `"number:NNN"` locator now adheres to the same default logic: only "usual" finished builds from default branch are searched by number and several builds can be returned if found.

Finding VCS roots (`.../app/rest/vcsRoots/... URL`)

(minor)A VCS root locator with `project` and `buildType` specified used "project" as the context for finding "buildType". This is no longer the case, the `buildType` locator should be full one to find the build configuration.

Build Configuration's Artifact Dependencies (entities returned by `.../app/rest/buildTypes/... URL`)

The `"artifact-dependencies"` sub-element of the `buildType` element now uses textual generated ids instead of numeric ones which depended on the order previously. This also affects requests for artifact dependencies modification.

The `"agent-requirements"` sub-element of the `buildType` element now uses generated ids instead of the parameter name as id. This also affects requests for agent requirements modification.

Editing agent requirements (`.../app/rest/buildTypes/.../artifact-requirements/... URL`)

Previously, on adding a new agent requirement for the same parameter, the existing one was overridden by the new one; now a new one is added.

Previously, on adding a new agent requirement, the parameter name was derived from the "id" attribute of the "agent-requirement" node. Since TeamCity 10, the parameter name is derived from the "property-name" property.

Test and problem occurrences (`.../app/rest/testOccurrences, .../app/rest/problemOccurrences URL`)

The sorting of the returned results is changed for some of the queries compared to the previous versions. For example, the `.../app/rest/testOccurrences?locator=build:(xxx)` request now returns the tests in the order they were run in the build. Previously, test occurrences were sorted by the new status and then by name. Problem occurrences were sorted by problem id. Also, the "build" dimension in the test/problem-related locators now supports multiple builds so for the requests which matched several builds via the "build" dimension, all the builds will be processed; previously only the first matching build was processed.

Entities

The " **property** " entity used to have its "own" Boolean attribute indicating whether the parameter is redefined in the build configuration as opposed to inherited from a template/project. Now the attribute is renamed to 'inherited' and its value is inverted.

The " **vcs-root** " and " **vcs-root-instance** " used to have "status" and "lastChecked" attributes. Now VCS root instance has "status" element (with `current` node which has `status` and `timestamp` attributes) and VCS root does not have the data as it produced undefined results in case of several VCS root instances.

The " **test** " entity "id" became String instead of Long (due to [inability](#) to represent some Java Long value in JavaScript)

Build type and template

The " **settings** " node used to contain all the supported settings. Now only those defined in the build configuration or template are present. The same is true for `.../app/rest/buildTypes/XXX/settings/*` requests: only the values changed from defaults are present.

Compatible agents

When querying for compatible agents, only the agents which can actually run the builds are now returned. By default, unauthorized, disconnected and disabled agents are not listed. This behavior differs from that in previous versions which had a number of discrepancies. Affected requests and entities: `.../app/rest/agents?locator=compatible:(...)`;

`.../app/rest/agents/.../compatibleBuildTypes` and `incompatibleBuildTypes`; nested nodes `Agent.compatibleBuildTypes`, `QueuedBuild.compatibleAgents`, `BuildType.compatibleAgents`

Changes from 9.1.6 to 9.1.7

No noteworthy changes.

Changes from 9.1.5 to 9.1.6

Known Issues

There is a [known issue](#) in the bundled dotCover run on Windows XP and Vista. You can use the [hotfix](#) or the [workaround](#) provided. The issue will be fixed in the next dotCover release.

There is a [known issue](#) with [NuGet Credentials](#) not working for the Authenticated Feed URL of the internal TeamCity NuGet server on the local agents. As a workaround, instead of using `%teamcity.nuget.feed.auth.server%`, please specify the external server URL in the build step and the build feature.

NuGet

NuGet versions prior to 2.8.6 require .Net Framework 4.0+ installed on the build agent, NuGet 2.8.6 and later requires .NET 4.5.

NUnit

Since version 9.1.6, TeamCity does not support NUnit 3 beta versions (released before NUnit 3.0.0).

The "Run a process per assembly" option of the [NUnit runner](#) has been removed from NUnit 3 settings. Configure the desired behavior using the required [command line options](#) in the [corresponding field](#).

Bundled tools updates

Bundled IntelliJ IDEA updated to version # 143.1945 (roughly equivalent to 15.0.3 with a few additional fixes).

Bundled version of Maven 3.2.x updated to 3.2.5.

Performance Monitor

Note on permissions: to [monitor performance](#) of a build agent run as a Windows [service](#), make sure the user starting the agent is member of the Performance Monitor Users group.

Changes from 9.1.4 to 9.1.5

Known Issues

There is a [known issue](#) in the bundled dotCover run on Windows XP and Vista. You can use the [hotfix](#) or the [workaround](#) provided. The issue will be fixed in the next dotCover release.

Product icons

JetBrains product icons are updated in accordance with the [new JetBrains branding](#).

Git

Since TeamCity 9.1.5, `git sparse-checkout` is [disabled](#) by default. To enable it in a TeamCity project, add the `teamcity.git.useSparseCheckout=true` [parameter](#) to this project.

Gradle: Breaking change compared to 9.1.2

Gradle runner `system.* properties`, introduced in TeamCity 9.1.2, defined for the build as JVM's system properties of the Gradle process, will not work since 9.1.5. Now the TeamCity system properties can be accessed in Gradle scripts as Gradle properties (similarly to the ones defined in the `gradle.properties` file) and are to be referenced as follows:

- a) name allowed as a Groovy identifier (the property name does not contain dots): `customUserProperty`
- b) name not allowed as a Groovy identifier (the property name contains dots, e.g. `build.vcs.number.1`): `project.ext["build.vcs.number.1"]`

Bundled tools updates

The bundled JetBrains IntelliJ IDEA (IDEA inspections and duplicates) has been updated to version 15.0.2

.Net tools updates

JetBrains ReSharper command line tools (.NET inspection and duplicates) have been updated to match ReSharper 10.0.2 release

TeamCity Visual Studio Addin Web installer updated to ReSharper 10.0.2 release

Bundled JetBrains dotCover updated to version 10.0.2

Changes from 9.1.3 to 9.1.4

Known Issues

Certain roles/permissions configurations can result in error loading roles and no ability for regular users to view projects. In such cases the "Circular reference is detected between roles" critical server error is displayed on the server administration pages for those logged in as server administrator. Please check the [workaround](#) for the issue.

Git agent-side checkout may malfunction (details at [TW-43202](#)) when the `teamcity.git.use.native.ssh=true` parameter is specified in a build configuration or in the agent config. To fix that, install the [#snapshot-34](#) build of the Git-plugin.

Git agent-side checkout works incorrectly with git client versions 1.7.0-1.7.4: the checkout directory contains files only, all directories are missing (details at [TW-43330](#)). To workaround the problem, add the `teamcity.git.useSparseCheckout=false` parameter in the Root TeamCity project.

TeamCity Windows binaries signatures

Since 9.1.4 the TeamCity Windows binaries are signed with SHA-2 code-signing certificate following the [Microsoft SHA-2 policies](#). This means that on systems prior to Windows XP SP3, the executables will not pass code signing verification; newer Windows systems require the corresponding security update from Microsoft.

Bundled tools updates

Bundled Oracle JRE (in both Server and Agent.exe installers) has been updated to version 1.8.0_66 (32-bit)

.Net tools updates

JetBrains ReSharper command line tools (.NET inspection and duplicates) have been updated to match ReSharper 10.0 release

TeamCity Visual Studio Addin Web installer updated to ReSharper 10.0 release

Bundled JetBrains dotCover updated to version 10.0

Changes from 9.1.2 to 9.1.3

Known Issues

There is a [known issue](#) in the bundled dotCover 3.2 which could cause a build's failure with the following exception: "System.Security.VerificationException: System.Security.VerificationException: Operation could destabilize the runtime". The issue is fixed in dotCover 10.0 bundled with TeamCity 9.1.4 release.

Bundled JVM (Server Windows installer and Agent Windows installer) is updated which results in disabled SSL RC4 cipher suite for outgoing HTTPS connections. For example this makes connections to CloudForge SVN servers non-functional with "SSLHandshakeException: Received fatal alert: handshake_failure" error. ([details](#))

Changes from 9.1.1 to 9.1.2

Known issues

The Command line runner can fail to execute a custom script if it has a non-default hashbang specified at the beginning of the script: [TW-42498](#)

When using Amazon EC2 cloud integration, an AMI-image, containing build agent versions 9.1.2-9.1.5 will appear twice with name EC2-i-abcdefgh and EC2-i-abcdefgh-1 consuming 2 licenses. To fix the issue please use an AMI-image with agent 9.1.6+. Just updating server to 9.1.6 won't help, if AMI remains the same. ([details](#))

Build status icons

Build status icons updated to a more "standard" look and are of a bit larger now.

Bundled tools updates

JetBrains ReSharper command line tools (.NET inspection and duplicates) have been updated to match ReSharper 9.2 release

TeamCity Visual Studio Addin Web installer updated to ReSharper 9.2 release

Bundled JetBrains dotCover updated to version 3.2

Bundled Oracle JRE (in both Server and Agent .exe installers) updated to version 1.8.0_60 (32-bit)

Changes from 9.1 to 9.1.1

Bundled Jacoco coverage library updated to version 0.7.5

Perforce VCS Roots with disabled ticket authentication won't run 'p4 login' operation anymore if password authentication is disabled on the Perforce server.

I.e. if password authentication is disabled, "Use ticket-based authentication" option must be enabled on the VCS Root. [TW-42818](#)

Changes from 9.0.x to 9.1

Bundled Ant

Bundled Ant distribution has been upgraded from 1.8.4 to 1.9.6. Note that Ant build steps using bundled Ant will use another version of Ant after the server upgrade. Ant 1.9.6 requires Java 1.5 at least, so builds using Ant and running under Java 1.4 will stop working.

MSTest runner converted into Visual Studio Tests runner

MSTest runner is merged with VSTest console runner (previously provided as a separate plugin) into the Visual Studio Tests runner.

Note that after upgrade to TeamCity 9.1, MSTest build steps are automatically converted to the Visual Studio Tests runner steps, while VSTest steps remain unchanged.



If you have used VSTest.Console runner plugin, make sure that you have latest version (build 32407) installed. The plugin version can be viewed on Administration->Plugins List page. Earlier versions of this plugin are not compatible with TeamCity 9.1 and may cause malfunction of .Net related build runners which can manifest with "java.lang.NoSuchMethodError: jetbrains.buildServer.runner.NUnit.NUnitVersion.parse(Ljava/lang/String;)" build errors. The plugin can be downloaded from [its page](#).

Consider migrating your vstest.console execution steps to the bundled Visual Studio Tests runner.

MSTest installation agent properties

TeamCity agent automatically detects the installed MSTest and used to expose the locations in `system.MSTest.N.N` system properties.

Since TeamCity 9.1, the locations are exposed via `teamcity.dotnet.mstest.N.N` configuration parameters. Check [TW-41845](#) for a workaround if you cannot easily change the properties usage.

Nested test reporting

Previously TeamCity supported a case when one test could have been reported from within another test using [service messages](#). Now, after the fix of [TW-40319](#), starting another test finishes the currently started test in the same "flow". To still report tests from within other tests, you will need to specify another [flowId](#) in the nested test service messages.

REST API

REST API uses version 9.1. Previous versions of API are still available under `/app/rest/9.0`, `/app/rest/8.1`, `/app/rest/7.0`, `/app/rest/6.0` URLs.

Finding builds

Summary (tl;dr): Some build filtering rules has subtle changes. Most importantly, a queued build can now be returned instead of 404 when searching by build id and meaning of the "project" locator dimension has changed to be not recursive. Also, failed to start builds are now not included until "failedToStart:any" locator dimension is specified. Details:

Affected requests: `/app/builds/<locator>...`, `/app/builds?locator=<locator>`, `/app/buildTypes/<btLocator>/builds` and others with build locator

locator: id:<number> or taskId:<number>

- previously, if the matching build was a queued one, 404 (Not Found) was returned
- now the queued build is returned

locator: project:<id>...

- previously, all the builds belonging to build configurations of the project and all it's subprojects (recursively) were found
- now only the builds belonging to build configurations of the project specified are found. For finding the builds recursively, use "affectedProject:<id>." dimension. This makes the usage consistent with build type locators.

locator: tag:<text>

- previously, when "<text>" used ":" character, that used to treat the entire "<text>" as tag name
- now the "<text>" is parsed as a nested locator. For searching tags with ":" character, locator "tag:(name:(<tag>))" should be used

locator: <text>

- previously, if `<text>` is not a number, response was 400 (Bad Request) with "LocatorProcessException: Invalid single value: '`<text>`'. Should be a number." message.
- now search by build number across all builds on the server is performed (this is not recommended to be used on production servers). For not found builds 404 (Not found) response is returned

locator: id: `<number>`,xxx:yyyy

- previously, build was found by id "`<number>`", other dimensions were ignored
- now if the build found by id does not match other dimensions, response is 404 (Not Found)

locator: agent: `<agentLocator>`

- previously `<agentLocator>` was used as is, without applying any defaults (unauthorized agents were included until specifically excluded)
- now `<agentLocator>` has the same behavior as in `/app/rest/agents` request: unauthorized agents are excluded by default

Finding projects

Searching project by name used to return 404 error if several projects were matched. Now will return the first project found.

Build's artifacts

There are several bugs fixed in listing build artifacts via `/app/rest/builds/<locator>/artifacts/*` requests which can cause subtle changes in the results for the request. Check the new behavior if you relied on the response.

The most important changes are:

- the initial path specified via URL part is searched for without current locator value, it will not generate 404 responses until there is no such artifact on disk.
- archives are not treated as directories (do not have children elements) by default. Specify "browseArchives:true" to treat archives as directories (in "recursive:true" mode only one level of archives is treated as directories)

Agents

Agents which are not known to the system (which were deleted) used to have id "-1". Now "id", "pool" and some other entries are no longer included for such agents.

Xcode 7 Support

[Experimental support](#) for Xcode 7 has been added.

Issue trackers integration

Due to API changes, third party issue trackers integration plugins might not be compatible with TeamCity version 9.1. Old plugins will not work and will report "java.lang.NoSuchMethodError: jetbrains.buildServer.issueTracker.AbstractIssueProviderFactory.<init>(Ljetbrains/buildServer/issueTracker/IssueFetcher;Ljava/lang/String;)V" error in `teamcity-server.log` (more details in the [issue](#)). If you observe such errors, please contact the [plugin authors](#). If you are the author of affected plugin, please refer to the related notes in [Open API Changes](#).

Changes from 9.0.4 to 9.0.5

No noteworthy changes.

Changes from 9.0.3 to 9.0.4

No noteworthy changes.

Changes from 9.0.2 to 9.0.3

No noteworthy changes.

Changes from 9.0.1 to 9.0.2

No noteworthy changes.

Changes from 9.0 to 9.0.1

Known Issues

If you have enabled versioned settings for projects which use meta-runners in TeamCity 9.0, on upgrade and following commit into the settings VCS root, the meta runners will be deleted from the server. Workaround is to commit the meta-runners definitions into the settings repository manually. Related issue: [TW-39519](#).

Oracle 10.x JDBC driver is not supported anymore

Due to missing support for national character sets (nvarchar) in Oracle 10.x JDBC drivers, TeamCity 9.0.1 will ask to upgrade Oracle JDBC drivers to the latest version. The minimal supported version of Oracle JDBC driver is 11.1.

Changes from 8.1.x to 9.0

Known Issues

- If you have custom artifact cleanup rules configured which mention ".teamcity" directory, build logs can be deleted by the cleanup procedure. Make sure you have build logs backup before upgrade and remove all the custom artifacts cleanup rules with ".teamcity". Related issue: [TW-40042](#). This issue is fixed in 9.0.3 release.
- If you use Microsoft SQL Server database with TeamCity, after the scheduled cleanup background run, TeamCity UI pages can lock until the server restart. See [TW-39549](#) for details. This issue is fixed in 9.0.2 release.
- If you use LDAP authentication on the server and there are lots of login attempts on the server (e.g. there is an active REST-using script), OutOfMemory errors can occur and require server restart. Consider installing an LDAP plugin with a fix from the [issue](#). This issue is fixed in 9.0.1 release.
- If you have large Maven projects, you can see builds failing with OutOfMemoryError. This is caused by update of back-end embedded Maven to 3.2.3 which has bigger memory footprint. Consider increasing Build Agent [memory limits](#) Related issue: [TW-41052](#)

UUID in XML settings files

Since TeamCity 9.0, disk-stored XML settings definitions of projects, build configurations and VCS roots have unique non-human-readable id (uuid) stored. These ids are automatically generated and are assumed to globally unique. On settings files copying, you need to change/make unique not only id (file name) and name (across siblings) of the entity, but also remove it's uuid from the file. TeamCity will generate new uuid automatically.

Build logs storage

The location of the build logs in the internal format stored under [TeamCity Data Directory](#) has changed. The build log files in internal format are now stored under hidden build artifacts.

Namely, the location has changed from `system/messages/CHyy/xyy.*` to `system/artifacts/<PROJECT_EXTERNAL ID>/<BUILD CONFIGURATION NAME>/xyy/.teamcity/logs/buildLog.*`.

Old build logs are migrated to the new location on TeamCity server startup ([TW-37362](#)). To avoid this migration, `teamcity.skip.logs.migration` internal property should be set before server startup.

Builds re-indexing after upgrade

On the first server start after upgrade from a version prior to 9.0, the server will reindex all builds for the purpose of builds search functionality and NuGet feeds. During the indexing time, some builds will not be available in the search results and in NuGet feeds. The server can also behave in less performant manner way during the indexing. `teamcity-server.log` has corresponding logging. On indexing finishing, there are "BuildIndexer (search) - Finished re-indexing builds" and "BuildIndexer (metadata) - Finished re-indexing builds" lines in the log.

Integration with issue trackers

Since TeamCity 9.0, the issue trackers are configured on the project level instead of the global server-wide configuration. On the server upgrade, all existing issue tracker integrations are moved to the Root project, which makes them still accessible to all the projects on the server.

WebSocket connections and proxy servers

Since 9.0, TeamCity tries to establish WebSocket connections between the browser and the server for the UI updates. If you have a proxy server (like nginx) in front of the TeamCity web UI, make sure that the proxy is [configured](#) properly to support WebSocket connections.

If the proxy is misconfigured or does not support the WebSocket protocol, a server health item will be shown for TeamCity system administrators. In this case TeamCity will use plain old polling for the UI updates as before.

REST API

REST API uses version 9.0. Previous versions of API are still available under `/app/rest/8.1`, `/app/rest/7.0`, `/app/rest/6.0` URLs.

- Change bean: the `webLink` attribute is renamed to `webUrl` to match other beans ([TW-34398](#)).
- Sub-elements representing empty collections in some of the beans are no longer included into responses (used to be included as an empty tag in XML).
- the builds `changes` element does not include the "count" attribute by default (for performance reasons), count can still be included by providing fields parameter like: "fields=\$long,changes(count,href)"
- The `/app/rest/agents` request now returns all the authorized agents by default (used to include unauthorized connected agents as well)
- queued builds now have the `id` attribute instead of the `taskId` attribute (they are the same for new builds since TeamCity 9.0)

Build tags-related changes

The `/app/rest/builds/<buildLocator>/tags` build request now returns a different XML: `<tags count="1"><tag name="TAG"/></tags>` instead of `<tags><tag>TAG</tag></tags>`.

The same applies to the `/app/rest/buildTypes/<buildTypeLocator>/<buildTypeLocator>/buildTags` request.

The same change in the structure also applies to the build's entity nested "tags" element.

To create a tag, there is an old way to post a plain-text tag name to the `app/rest/builds/<buildLocator>/tags` URL.

When sending POST or PUT XML or JSON requests to the URL, the new XML format is to be used (`<tag name="TAG"/></tags>` instead of `<tag>TAG</tag>`).

Handling tests with the same name within a build

In TeamCity 9.0, multiple tests with the same name within the same build are considered a single test with an invocation count. If any of these test runs fail, the whole test is considered failed in the build. The related issue is [TW-24212](#).

This change results in drop of test number counters in builds which have multiple runs for the same test. If you have a build failure condition which relies on test number in the build, this change may affect you.

If you need the tests to be treated as separate ones, consider running them in the test suites with different names or otherwise changing the test/running logic to change the full test name displayed in TeamCity.

Database-related changes

The national character sets (nchar, nvarchar, nlob types) for text fields are now supported in MS SQL databases used by TeamCity. It is recommended to use the Microsoft native JDBC driver, as jTDS JDBC driver does not support the nchar and nvarchar characters. If you still use jTDS, please [migrate](#).

Upon upgrade and entering the normal working status, TeamCity starts a background process to move the entries from the `vcs_changes` database table to `vcs_change` table. This process is transparent and you can continue working with the server as usual. It has negligible impact on the server performance and the only affected logic is the projects import feature (it is recommended to be used only with backups taken after the process is completed). The progress of the process can be seen on the Backup section in the server administration, along with "TeamCity is currently optimizing VCS-related data in the database for better backup/restore performance" message.

The other important thing is that the data copying increases the size of the raw database storage.

If this is an issue for your case (e.g. it might be with Microsoft SQL Server database with set database size limit), it is recommended to ensure the database size limit is twice the current size before the upgrade. It is possible to perform database-specific procedures to shrink the storage to match the actually stored data after the VCS changes migration process finishes.

VCS Root-related changes

The Git and Mercurial VCS roots no longer provide the ability to specify a custom clone path on the server for new VCS roots. If you need this ability, set the following internal properties to `true` for git and mercurial respectively: `teamcity.git.showCustomClonePath`, `teamcity.hg.showCustomClonePath`.

Visual Studio Addin

The TeamCity Add-in installed as a part of [ReSharper Ultimate](#) will remove the pre-bundle products versions: TeamCity and ReSharper versions prior to 9.0, dotCover prior to 3.0, dotTrace prior to 6.0.

Besides, it will not use the settings provided by the 8.1 version. The traditional add-in downloaded from TeamCity server can still use settings from previous version.

Other

TeamCity agent installation via the Java web start installation package is no longer available.

Changes from 8.1.1 to 8.1.4

No noteworthy changes.

Changes from 8.1 to 8.1.1

Command Line Runner

The change in behavior introduced in 8.1 (see [below](#)) has been fixed. Command line runners using "Executable with parameters" option which were created/changed with TeamCity 8.1 can expose a change in behavior with the upgrade. The recommended approach is to switch to "Custom script" option instead of "Executable with parameters" in command line runner.

Separate download for VSTest.Console runner

VSTest console runner is no longer bundled with TeamCity and is available as a separate plugin. For download details, see the [plugin page](#)

Changes from 8.0.6 to 8.1

Known issue with creating MS SQL database with integrated security

When installing TeamCity anew and creating an MS SQL database with integrated security using the database setup UI, you may receive an error. We are planning to resolve it in the next bugfix, meanwhile use the following workaround:

1. After receiving the error, stop the TeamCity server.
2. Start the TeamCity server.
3. On the database configuration screen, fill out the required information. Do not use the Refresh button. Make sure the information specified is correct.
4. Continue with the configuration steps.

If for some reason the workaround above does not resolve the problem, do the following:

1. Start the TeamCity server, approve a new database creation, configure the MS SQL access with a login and password, without [integrated security](#).
2. Ensure that TeamCity works properly.
3. Stop the TeamCity server.
4. Modify the [database.properties](#) file: configure MS SQL connection string to use integrated security, and remove the login and password.
5. Start the TeamCity server again.

Known issue with VSTest.Console runner

A new "VSTest.Console" runner which first appeared in TeamCity 8.1 is in experimental state and is not recommended for production use at this time. It will not be present in TeamCity 8.1.x by default (will be available as a separate download).

Known issue with PowerShell runner

PowerShell runner plugin is broken in 8.1. Fix is available, please follow instructions in [issue comment](#).

Known issue with Command Line Runner

Command line runner using "Executable with parameters" option can process quotes (") and percentage signs (%) in a bit different way then in previous TeamCity versions (see details in the [issue](#)). To switch back to the previous (8.0) behavior you may specify `command.line.run.as.script=false` configuration parameter in a build configuration or in a project. The issue is fixed in 8.1.1.

The recommended approach is to switch to "Custom script" option instead of "Executable with parameters" in command line runner.

Memory Settings

If you have not [switched to 64 bit JVM](#) yet and use `-Xmx1300` memory setting for the server and the server is running on Windows, consider decreasing the setting to `-Xmx1200` as otherwise you might encounter "Native memory allocation (malloc) failed" JVM crash. See the [recommended memory settings](#) for details.

Actions menu

Some actions has moved under the "Actions" button available at the top-right of the page, near Run button.

These include:

"Label this build sources" on Changes tab of a build,
"Pause", "Copy", "Move", "Delete", "Associate with Template", "Extract Template", "Extract Meta-Runner" on build configuration settings administration page,
"Copy", "Move", "Delete", "Archive", "Bulk edit IDs" on project settings administration page.

Create Maven build configuration is not available by default

Action "Create Maven build configuration" is no longer available. Most of its functionality is covered by create project from URL and create VCS root from URL pages.

triggeredBy parameter from GroovyPlug plugin

The `build.triggeredBy` and `build.triggeredBy.username` configuration parameters provided by the [plugin](#) added by the plugin are now [available](#) without the plugin under `teamcity.build.triggeredBy` and `teamcity.build.triggeredBy.username` names respectively. Consider migrating to the latter set of parameters in your settings if you used the plugin's ones.

Shared Resources build feature

If the build takes lock on all values of a resource with custom values, these values are provided as lock values in build parameters. Corresponding issue: [TW-29779](#)

TeamCity Disk Space Watcher

The following [internal properties](#) define free disk space thresholds on the TeamCity server machine:

- `teamcity.diskSpaceWatcher.threshold` set to 500 Mb by default displays a warning on all the pages of the TeamCity Web UI.
- `teamcity.pauseBuildQueue.diskSpace.threshold` set to 50 Mb by default pauses the build queue.

The `teamcity.diskSpaceWatcher.softThreshold` property is removed.

PowerShell

The PowerShell plugin now uses the version that was specified in the UI as the `-Version` command line argument when executing scripts. Corresponding issue: [TW-33472](#)

REST API

The latest version of the API has not changed, it is still "8.0" while there are changes in the API detailed below. If you find this inconvenient for your REST API usages, please comment in the corresponding [issue](#).

Entities returned in the response of REST API requests might now exclude attributes/elements with empty/default values. This is relevant for boolean fields with "false" value and empty collections. The recommended approach is to make sure the client code assumes "false" as a value for not present boolean attributes/elements.

"projectName" of buildType node now contains full project name (with the names of the parent projects) instead of the short name of the project.

In the lists of builds, "startDate" attribute is not longer included in the "build" node. It has become an element instead of attribute to match the full build data representation. If your REST API usage is affected, check [a way](#) to get that element in a request for the list of builds.

Requests `/app/rest/buildTypes/XXX/parameters/YYY` and `/app/rest/projects/XXX/parameters/YYY` now support "text/plain" and "application/xml" responses. To get plain text response (which was the only supported way before 8.1) you will need to supply "Accept: text/plain" header to the request.

Password properties of the VCS roots are now included into the responses, just without values.

CCTray-format XML (`app/rest/cctray/projects.xml`) does not include paused build configurations now.

Response to the experimental request `/app/rest/buildTypes/XXX/investigations` has changed the format and got additional fields to cover tests and problem investigations. There is an internal property `rest.beans.buildTypeInvestigationCompatibility` to include removed sub-items. Please let us know via [support email](#) if you need to use the internal property.

Eclipse plugin

Dropped support of Subversion 1.4-1.6. Now only Subversion 1.7-1.8 working copies formats supported.

Changes from 8.0.5 to 8.0.6

No noteworthy changes.

Changes from 8.0.4 to 8.0.5

No noteworthy changes.

Changes from 8.0.3 to 8.0.4

First Cleanup

First Cleanup after server upgrade might take a bit more time then regularly if there are many builds on the server. Following cleanups will then run a bit faster then in previous versions.

Changes from 8.0 to 8.0.3

No noteworthy changes.

Changes from 7.1.x to 8.0

Project and Build Configuration IDs

This version introduces user-assignable IDs for projects and build configurations. This new ID is now used instead of internal id (projectN and btNNN) in at least:

- URLs of the web pages and artifact downloads
- in REST API
- project IDs are also used in directory names on the server under `<TeamCity Data Directory>\system\artifacts` instead of project names used prior to TeamCity 8.0

If you used any of the above, please, verify if you are affected by the change.

Learn more about IDs at [Identifier](#).

On upgrade, all the projects get automatically generated IDs based on their names.

Build configuration IDs are set to be equal to internal (btNNN) ids and can be later changed from the Administration UI via the Regenerate ID or Bulk Edit IDs actions.

Please note that the names of the projects and build configurations are no longer unique server-wide (are only unique within the direct parent project) and can contain any symbols which might be relevant if you used these in directory or file names.

Project settings format on disk

The format of the project settings storage on the disk under `<TeamCity Data Directory>\config` has been changed.

If you used any tools to read or update `project-config.xml` files, you will need to update the tools. It is recommended to use REST API or TeamCity open API (Java) to make changes so that the tools are not hugely affected by the format change.

Build Configuration templates

In version 8.0 build configuration templates support project hierarchy and TeamCity uses new rules:

- The TeamCity administration UI limits the use of templates only to those from the current project and its parents. On copying a project or a build configuration, the templates which do not belong to the target project or one of its parents are automatically copied.

- TeamCity no longer allows attaching a build configuration to a template if the template does not belong to the current project or one of its parents.
- Before version 8.0 it was possible to extract templates from a build configuration of one project to an unrelated project or to associate a build configuration in one project with a template in another. After upgrade to TC 8.0, such templates will become inaccessible in the current project. To reuse build configuration templates from an unrelated project, it is recommended to manually move them into the common parent project (or the Root project if you want them to be globally available).

JVM-originated agent parameters (os.arch and others)

The agent no longer reports system properties which come from the agent JVM: `system.os.arch`, `system.os.name`, `system.os.version`, `system.user.home`, `system.user.name`, `system.user.timezone`, `system.user.language`, `system.user.country`, `system.user.variant`, `system.path.separator`, `system.file.encoding`, `system.file.separator`.

All the aforementioned parameters are now reported as configuration parameters with the `teamcity.agent.jvm.` prefix instead.

If you used any of the parameters, make sure you update them to the new values.

IntelliJ IDEA project runner

IntelliJ IDEA project runner now uses IntelliJ IDEA's external make tool to build projects. Since this tool requires Java 1.6 to work, IntelliJ IDEA project runner now requires Java 1.6 (at least) too.

Clean-up for build configurations with feature branches

Build configurations with feature branches now process clean-up rules per-branch which can result in more builds preserved during clean-up than in previous versions. See [details](#).

Team Foundation Server integration

TFS now prefers Team Explorer 2012 to Team Explorer 2010 (if both are installed) for TFS operations

Compatibility with YouTrack

If you use JetBrains YouTrack and use its TeamCity integration features, please note that only YouTrack version 4.2.4 and later are compatible with TeamCity 8.0.

If you need earlier YouTrack versions to work with TeamCity 8.0, please [let us know](#).

REST API

External ids

There are changes in the API related to the new external ids for project/build types/templates as well as other changes. The old API compatible with TeamCity 7.1 is still provided under `"/app/rest/7.0"` URL.

If you used URLs with locators having "id" for projects, build configuration or templates (like `.../app/rest/projects/id:XXX` or `.../app/rest/buildTypes/id:XXX`), please, update the locators to one of the following:

- (recommended) `"id:EXTERNAL_ID"` (you can get the external ID in web UI URLs or via a request to `.../app/rest/projects/internalId:OLD_ID/id`)
- just `"ANY_ID"` to find the entity either by its internal, external id or name (use with caution: you can find more than you expect)
- `"internalId:INTERNAL_ID"` to find the entity by the internal id
You can also use the `"/app/rest/7.0/"` URL prefix instead of `"/app/rest/"` to work with 7.0-version of REST API which still uses internal IDs except for finish build trigger properties.

Also, it is possible to set the internal property `rest.compatibility.allowExternalIdAsInternal=true` to turn on the compatibility mode so that `id:xxx` locators will search also by the internal id. Note that this will be dropped in the future versions of TeamCity and is not recommended for use.

Other Changes

Requests for builds `".../builds/<locator>/..."` and `".../builds?locator=<locator>"` no longer return personal and canceled builds by default. To include those, make sure you add `personal:any,canceled:any` to the locators.

The `"relatedIssues"` element of the build entity no longer contains a full list of related issues. It has only the `"href"` attribute whose value can be used to get the related issues via a separate request.
There is also an internal property `"rest.beans.build.inlineRelatedIssues"` which can be set to `true` to return the `"relatedIssues"` node back for compatibility. See [TW-20025](#) for details. Also, the `".../builds/xxx/related-issues"` URL is renamed to `".../builds/xxx/relatedIssues"`.

The "source_buildTypeId" property is dropped from snapshot and artifact dependency nodes. Instead, the "source-buildType" sub-element is added with a reference to the build type.

Creating dependencies is still supported with the "source_buildTypeId" property, but is deprecated. There is an internal property "rest.compatibility.includeSourceBuildTypeInDependencyProperties" which can be set to `true` to include the "source_buildTypeId" property back.

In version 8.0 VCS roots support project hierarchy:

- When creating a VCS root, the `project` element should always be provided now. The element supports the `locator` attribute to specify the project.
- the `shared` attribute is dropped from the VCS root: after upgrade, such VCS roots are attached to the root project (with the "_Root" ID) and become globally available.
- when copying projects and build configurations, the `shareVCSRoots` attribute is no longer present. To make the VCS root available to projects and build configurations, move it to the parent/root project and then proceed with the copying.

It is recommended to create projects hierarchy which corresponds to organizational/settings sharing structure and move the VCS roots to most nested umbrella projects. Users then can be granted "Create / delete VCS root" role in the project to be able to edit VCS roots. Please note that users can edit a VCS root only if it is used in the projects they have "Edit project" permission for.

The "template" attribute in a build configuration template node is renamed to "templateFlag".

PUT for `/users/<locator>/roles` and `/userGroups/<locator>/roles` now accepts list of roles as it should and replaces existing roles instead of accepting single roles and adding it.

Many of PUT and POST requests which used to return nothing now return the entities created.

Open API changes

[See details](#)

Shared Resources plugin

If you used the [Shared Resources plugin](#) with TeamCity 7.1.x, make sure to remove it as it is now bundled. See the [upgrade instructions](#).

Queue Manager plugin

If you used the [QueueManager plugin](#), make sure to remove it as it is now bundled. See the [upgrade instructions](#)

Bundled Maven

Maven bundled with TeamCity upgraded to version 3.0.5.

HTTPS connections from agents to server

If your agents connect to the TeamCity server by HTTPS protocol, and after upgrade agents fail to connect with error messages like:

```
javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake_failure
```

then you should change Tomcat SSL connector configuration, i.e. add the following attribute to SSL connector and restart TeamCity server:

```
sslEnabledProtocols="TLSv1,SSLv3,SSLv2Hello"
```

The issue only manifests when the server runs under Java 1.7.

See also:

- http://mail-archives.apache.org/mod_mbox/tomcat-users/201302.mbox/%3C512559F7.4080001@gmail.com%3E
- <http://youtrack.jetbrains.com/issue/TW-30221#comment=27-561843>

Changes from 7.1.4 to 7.1.5

teamcity.build.branch parameter semantics has changed, see <http://youtrack.jetbrains.com/issue/TW-23699#comment=27-448002>

Changes from 7.1.3 to 7.1.4

No noteworthy changes.

Changes from 7.1.2 to 7.1.3

No noteworthy changes.

Please check up-to-date list of [known regressions](#) for the version in our issue tracker.

Changes from 7.1.1 to 7.1.2

Possible issues with hg server-side checkout

There is a known issue with 7.1.2 release: [TW-24405](#) which can reproduce when server-side checkout, labeling or file content viewing are used for Mercurial repository.

If you experience the error with message "abort: destination 'hg1' is not empty", please install the patch attached to the issue.

Other known issues

Please also check a list of [known regressions](#) for the version in our issue tracker.

Changes from 7.1 to 7.1.1

No noteworthy changes.

Changes from 7.0.x to 7.1

Windows service configuration

Since version 7.1, TeamCity uses its own service wrapping solution for the TeamCity server as opposed to that of default Tomcat one in previous versions.

This changes the way TeamCity service is configured (data directory and server startup options including memory settings) and makes it unified between service and console startup.

Please refer to the updated [section](#) on configuring the server startup properties.

Agent windows service started to use OS-provided environment variables. Once Agent server (and JVM) are x86 processes, agent will report x86 environment variables. The change may affect your CPU bitness checks. See [MSDN Blog](#) on how to check if machine supports x64 by reported environment variables

Default location for TeamCity Data Directory when installed with Windows installer

This is only relevant for fresh TeamCity installations with Windows installer. Existing settings are preserved if you upgrade an existing installation.

Windows installer now uses `%ALLUSERSPROFILE%\JetBrains\TeamCity` location as default one for [TeamCity Data Directory](#). In TeamCity 7.0 and previous versions that used to be `%USERPROFILE%\BuildServer`.

Windows domain login module

When TeamCity server runs under Windows and Windows domain user authentication is used, TeamCity now uses another library (Waffle) to talk to the Windows domain.

Under Linux the behavior is unchanged: jCIFS library is used as it were.

Unless you specified specific settings for jCIFS library in `ntlm-config.properties` file, your installation should not be affected.

If you experience any issues with login into TeamCity with your Windows username/password after upgrade, please provide details [to us](#). In the mean time you can switch to using old jCIFS library. For this, add `teamcity.ntlm.use.jcifs=true` line into [internal properties file](#).

Please note that jCIFS library approach can be deprecated in future versions of TeamCity, so the property specification is not recommended if you can go without it.

Checkout directory change for Git and Mercurial

Build configurations that have either Git or Mercurial VCS roots and use default checkout directory will perform clean checkout upon upgrade. The clean checkout will be triggered by changed default checkout directory name. Further builds will reuse the checkout directory more aggressively (all builds using different branches but using the same VCS root will use the same directory). This affects agent- and server-side checkouts.

Perforce agent checkout workspace names change

Build configurations using Perforce agent-side checkout will perform clean checkout once after server upgrade. This is related to changed names for automatically generated Perforce workspaces.

SVN revision format

For changes, detected in external repositories, SVN revision got format `NNN_MMM:EXTUUID_CHANGEDATE`, where `NNN` - revision of the main repository, `MMM` - revision of externals repository, `EXTUUID` - UUID of externals repository, `CHANGEDATE` - change timestamp. This change may affect plugins/REST api clients which use revision of the last build change somehow.

Eclipse IDE plugin compatibility

Since TeamCity 7.1, Eclipse version 3.3 (Europa) is no longer supported by TeamCity Eclipse plugin. Eclipse 3.8 and Eclipse 4.2 (Juno) are now supported.

Default schema when Microsoft SQL Server is used as an external database

Starting with version 7.1 TeamCity works only with a single database schema unlike previous versions when it could work with tables in any schemas of the database server.

TeamCity-related tables should now be located in the database schema which is set as default one for the database user used by TeamCity to connect to the database.

This change may require reconfiguration of the database to set default schema for the user used by TeamCity server to connect to the database.

Please check that all TeamCity-related tables are located in the default user's schema before performing the upgrade. (e.g. using the 'sys.tables' view)

If the default user's schema is not set right, TeamCity can report "TeamCity database is empty or doesn't exist. If you proceed, a new database will be created." message on the first start of newer TeamCity.

To change user's default schema, use the 'alter user' SQL command.

For the default schema description, see the "Default Schemas" section in the [corresponding documentation](#).

Open API changes

See [details](#)

Changes from 7.0.1 to 7.0.4

No noteworthy changes.

Changes from 7.0 to 7.0.1

HTML report tabs URLs Change

If you use direct links for build-level or project-level [report tabs](#), please update the links as they will [change](#) after upgrade. The change is necessary to make the feature more reliable.

Changes from 6.5.x to 7.0

(Known issue) Build can hang or produce memory error for NUnit and other .Net test runners

Affected are: .Net test runners (NUnit, MSTest, MSpec) as well as TeamCity NUnit console launcher.

Reproduces when path to test assemblies has several deep paths without wildcards ("*").

Visible outcome: build hangs or fails with OutOfMemoryException error after "Starting ...JetBrains.BuildServer.NUnitLauncher.exe" link in the build log.

The issue ([TW-20482](#)) is fixed and the fix will be included in the next release. Patch with a fix is [available](#).

Minimum Supported Project JDK for Ant Runner

Starting with this version Ant runner requires minimum of JDK 1.4 in runtime build part (was 1.3 previously). This means that you will not be able to use TeamCity Ant runner if your project uses JDK 1.3 for compilation or tests running.

For projects that require JDK 1.3 you can use command-line runner instead and configure "XML report processing" build feature to parse test reports.

Supported Java for Server and Agent

Starting with this version the following requirements

- TeamCity server should be run with JRE 1.6 or above (was 1.5 previously). TeamCity .exe distribution is already bundled with appropriate Java. For .tar.gz or .war TeamCity distributions you might need to install and configure server [manually](#).
- TeamCity agent should be run with JRE 1.6 or above (was 1.5 previously). Agent .exe distribution is already bundled with appropriate Java. If you used .zip agent distribution or installed the TeamCity agent with TeamCity version 5.0 or earlier, you might need [manual steps](#). If you run TeamCity 6.5.x, please check "Agents" page of your existing TeamCity server: the page will have a yellow warning in case any of the connected agents are running JDK less than 1.6.



"Important!"

If any of your agents are running under JDK version less than 1.6, the agents will fail to upgrade and will stop running

on the server upgrade. You will need to recover them manually by installing JDK 1.6 and making sure the agents will use it.

Project/Template parameters override

In TeamCity 7.0 project parameters have higher priority than parameters defined in template, i.e. if there is a parameter with some name and value in the project and there is parameter with the same name and different value in template of the same project, value from the project will be used. This was not so in TeamCity 6.5 and was **changed** to be more flexible when template belongs to another project.

Build configuration parameters have the highest priority, as usual.

Support for Sybase is discontinued

From this version support for Sybase as external database is shifted back into "experimental" state.

The reason for this decision is that it does not seem like the database is actively used with TeamCity, and supporting it requires a significant effort from TeamCity team which otherwise can be directed to improving more important areas of the product.

While it should be still **possible**, we do not recommend using Sybase as an external database and we are not planning to provide support for the Sybase-related issues.

Please consider using one of the other **databases supported**. If you use Sybase, please migrate to another database before upgrading TeamCity.

REST API Changes

- Several objects got additional attributes and sub-elements (e.g. BuildType, VcsRoot). Please check that your parsing code still works.
/buildTypes/ path: BuildType object dropped runParameters field (as well as /<locator>/runParameters path is dropped) in favor of steps collection and /<locator>/steps/ path.
- A **bug** fixed which resulted in non-array JSON representation of single element arrays for some resources. Please check if your code is affected.
- in build object, "dependency-build" element is renamed to "snapshot-dependencies", revisions/revision/vcs-root is renamed to revisions/revision/vcs-root-instance (and it points to resolved VCS root instance now), revisions/revision/display-version is renamed to "version".
- in buildType object, "vcs-root" element is renamed to "vcs-root-entries"

Old version of the REST API is available via /app/rest/6.0/... URL in TeamCity 7.0. Please update your REST-using code as future versions of TeamCity might drop support for 6.0 protocol.

Minimum version of supported Tomcat

If you use TeamCity .war distribution, please note that Tomcat 5.5 is no longer supported. Please update Tomcat to version 6.0.27 or above (Tomcat 7 is recommended).

Swabra

swabra.handle.exe.path and handle.exe.path configuration parameters are no longer supported for providing path to Sysinternals handle.exe on agents. See the [plugin page](#) for the details.

Open API Changes

Classes from jetbrains.buildServer.messages.serviceMessages package like jetbrains.buildServer.messages.serviceMessages.BuildStatus no longer depend on jetbrains.buildServer.messages.Status class. To make your code compatible with TeamCity 6.0 - 7.0 you can use jetbrains.buildServer.messages.serviceMessages.ServiceMessage#asString methods, for example:

```
ServiceMessage.asString("buildStatus", new HashMap<String, String>() {{
    put("text", "Errors found");
    put("status", "FAILURE");
}});
```

See also [Open API Changes](#)

Changes from 6.5.4 to 6.5.6

No noteworthy changes

Changes from 6.5.4 to 6.5.5

(Known issue infex in 6.5.6) .NET Duplicates finder may stop working, the patch is available, please see this comment: <http://youtrack.jetbrains.net/issue/TW-18784#comment=27-261174>

Changes from 6.5.3 to 6.5.4

No noteworthy changes

Changes from 6.5.3 to 6.5.4

No noteworthy changes

Changes from 6.5.2 to 6.5.3

No noteworthy changes

Changes from 6.5.1 to 6.5.2

Maven runner

Working with MAVEN_OPTS has changed again. Hopefully for the last time within the 6.5.x iteration. (see <http://youtrack.jetbrains.net/issue/TW-17393>)

Now TeamCity acts as follows:

1. If MAVEN_OPTS is set TeamCity takes JVM arguments from MAVEN_OPTS
2. If "JVM command line parameters" are provided in the runner settings, they are taken instead of MAVEN_OPTS and MAVEN_OPTS is overwritten with this value to propagate it to nested Maven executions.

Those who after upgrading to 6.5 had problems of not using MAVEN_OPTS and who had to copy its value to the "JVM command line parameters" to make their builds work, now don't need to change anything in their configuration. Builds will work the same way they do in 6.5 or 6.5.1.

Changes from 6.5 to 6.5.1

(Fixed known issue) Long upgrade time and slow cleanup under Oracle

Changes from 6.0.x to 6.5

(Known issue) Long upgrade time and slow cleanup under Oracle

On first upgraded server start the database structures are converted and this can take a long time (hours on a large database) if you use Oracle external database ([TW-17094](http://youtrack.jetbrains.net/issue/TW-17094)). This is already fixed in 6.5.1.

Agent JVM upgrade

With this version of TeamCity we added semi-automatic upgrade of JVM used by the agents. If there is a Java 1.6 installed on the agent, and the agent itself is still running under the Java 1.5, TeamCity will ask to switch agent to Java 1.6. All you need is to review that detected path to Java is correct and confirm this switch, the rest should be done automatically. The operation is per-agent, you'll have to make it for each agent separately. Note that we recommend to switch to Java 1.6, as at some point TeamCity will not be compatible with Java 1.5. Make sure newly selected java process has same firewall rules (i.e. port 9090 is opened to accept connections from server)

IntelliJ IDEA Coverage data

Coverage data produced by IntelliJ IDEA coverage engine bundled with TeamCity 6.5 can only be loaded in IntelliJ IDEA 10.5+. Due to coverage data format changes older versions of IntelliJ IDEA won't be able to load coverage from the server.

IntelliJ IDEA 8 is not supported

Plugin for IntelliJ IDEA no longer supports IntelliJ IDEA 8.

Unsupported MySQL versions

Due to [bugs](#) in MySQL 5.1.x TeamCity no longer supports MySQL versions in range 5.1 - 5.1.48. TeamCity won't start with appropriate message if unsupported MySQL version is detected. Please upgrade your MySQL server to version 5.1.49 or later.

Finish build properties are displayed

Finished builds now display all their properties used in the build on "Parameters" tab. This can potentially expose names and values of parameters from other builds (those that the given build uses as artifact or snapshot dependency). Please make sure this is acceptable in your environment. You can also manage users who see the tab with "View build runtime parameters and data" permissions which is assigned "Project Developers" role by default.

PowerShell runner is bundled

If you installed [PowerShell plugin](#) manually, please remove it from `.BuildServer/plugins` as a fresh version is now bundled with TeamCity.

Changed settings location

- XML test reporting settings are moved from runner settings into a dedicated [build feature](#).
- "Last finished build" artifact dependency on a build which has snapshot dependency is automatically converted into dedicated "Build from the same chain" source build setting.

Responsibility is renamed to Investigation

A responsibility assigned for a failing build configuration or a test is now called investigation. This is just a terminology change to make the action more neutral.

If you have any email processing rules for TeamCity investigation assignment activity, please check if they need updating to use new text patterns.

REST API Changes

Several objects got additional attributes and sub-elements (e.g. "startDate" in reference to a build, "personal" in a change). Please check that your parsing code still works.

Cleaning Non-default Checkout Directories

In previous releases, if you have specified build [checkout directory](#) explicitly using absolute path, TeamCity would not clean the content of the directory to free space on the disk.

This is no longer the case.

So if you have absolute path specified for the checkout directory and you need the directory to be present on agent for other build or for the machine environment, please set `system.teamcity.build.checkoutDir.expireHours` [property](#) to "never" and re-run the build. Please take into account that using [custom checkout directory](#) is not recommended.

If you are using one of `system.teamcity.build.checkoutDir.expireHours` properties and it is set to "never" to prevent the checkout directory from automatic deletion, the directory might be deleted once after TeamCity upgrade. Running the build in the build configuration once after the upgrade (and within 8 days from the previous build) will ensure that the directory preserves the "protected" behavior and will not be automatically removed by TeamCity.

Free disk space

This release exposes [Free disk space feature](#) in UI that was earlier only available via setting build configuration properties. While the old properties still work and take precedence, it is highly recommended to remove them and specify the value via "Disk Space" build feature instead. Future TeamCity versions might stop to consider the properties specified manually.

Command line runner

`@echo off` which turns off command-echoing is added to scripts provided by "Custom script" runner parameter. To enable command-echoing add `@echo on` to the script.

Windows Tray Notifier

You will need to upgrade windows tray notifier by uninstalling it and installing it again. Unfortunately, auto-upgrade will not work due to issues in old version of Windows Tray Notifier.

Maven runner

- In earlier TeamCity versions Maven was executed by invoking the 'mvn' shell script. You could specify some parameters in MAVEN_OPTS and some in UI. Maven build runner created its own MAVEN_OPT by concatenating these two (`%MAVEN_OPTS%+jvmArgs`). In this case, if some parameter was specified twice - in MAVEN_OPTS and in UI, only the one specified in MAVEN_OPTS was effective. Starting with TeamCity 6.5 Maven runner forms direct java command. While this approach solves many different problems, it also means that MAVEN_OPTS isn't effective anymore and all JVM command line parameters should be specified in build runner settings instead of MAVEN_OPTS.
- Those who had to manually setup surefire XML reporting for Maven release builds in TeamCity 6.0.x because otherwise tests weren't reported, now can forget about that. Since TeamCity 6.5 surefire tests run by `release:prepare` or `release:perform` goals are automatically detected. So don't forget to switch surefire XML reporting off in the build configuration settings to avoid double-reporting!

Email sending settings

Please check email sending settings are working correctly after upgrade (via Test connection on Administration > Server Configuration > EMail Notifier). If no authentication is needed, make sure login and password fields are blank. Non-blank fields may cause email sending errors if SMTP server is not expecting authentication requests.

XML Report Processing

Tests from Ant JUnit XML reports can be reported twice (see [TW-19058](#)), as we no longer automatically ignore TESTS-xxx.xml report.

To workaround this avoid using `*.xml` mask and specify more concrete rules like `TEST-*.xml` or alike that will not match report with name starting with "TESTS-"

Open API Changes

Several return types have changes in TeamCity open API, so plugins might need recompilation against new TeamCity version to continue working.

Also, some API was deprecated and will be discontinued in later releases. It is recommended to update plugins not to use deprecated API.

See also [Open API Changes](#)

Changes from 6.0.2 to 6.0.3

No noteworthy changes

Changes from 6.0.1 to 6.0.2

Maven and XML Test Reporting Load CPU on Agent

If you use Maven or XML test reporter and your build is CPU-intensive, you might find important the [known issue](#). Patch is available, fixed in the following updates.

Changes from 6.0 to 6.0.1

No noteworthy changes

Changes from 5.1.x to 6.0

Visual Studio Add-in and Perforce

There is critical bug in TeamCity 6.0 VS Add-in when Perforce is enabled. This can cause Visual Studio hangs and crashes. The fixed add-in version is [available](#). (related [issue](#)). The issue is fixed in TeamCity 6.0.1.

TFS checkout on agent

TFS checkout on agent might refuse to work with errors. Patch is available, see the [comment](#). Related [issue](#). The issue is fixed in TeamCity 6.0.1.

Error Changing Priority class

You may encounter a browser error while changing priority number of a priority class. A patch is available in a related [issue](#). The issue is fixed in TeamCity 6.0.1.

IntelliJ IDEA Compatibility

IntelliJ IDEA 6 and 7 are no longer supported in TeamCity plugin for IntelliJ IDEA.

Also, if you plan to upgrade to IntelliJ IDEA X (or other JetBrains IDE) please review this [known issue](#).

Build Failure Notifications

TeamCity 6.0 differentiates between build failure occurred while running a build script and one occurred while preparing for the build. The errors occurring in the latter case are called "failed to start" errors and are hidden by default from web UI (see "Show canceled and failed to start builds" option on Build Configuration page).

Since TeamCity 6.0, there is a separate notification rule "The build fails to start" which applies for "failed to start" builds. All the rest build failure notifications relate to build script-related failures.

Please note that on upgrade, all users who had "The build fails" notification on, will automatically get "The build fails to start" option to preserve old behavior.

Properties Changes

`teamcity.build.workingDir` property should no longer be used in non-runner settings. For backward compatibility, the property is supported in non-runner settings and is resolved to the working directory of the first defined build step.

Swabra and Build Queue Priorities Plugins are Bundled

If you have installed the plugins previously, please remove them (typically from `.BuildServer/plugins`) before starting upgraded TeamCity version.

Maven runner

Java older than 1.5 is no longer supported by the agent part of Maven runner. Please make sure you specify 1.6+ JVM in Maven runner settings or ensure `JAVA_HOME` points to such JVM.

NUnit and MSTest Tests

If you had NUnit or MSTest tests configured in TeamCity UI (sln and MSBuild runners), the settings are extracted from the runners and converted to a new runner of corresponding type.

Please note that implementation of tests launching has changed and this affected relative paths usage: in TeamCity 6.0 the working directory and all the UI-specified wildcards are resolved based on the build's [checkout directory](#), while they used to be based on the directory containing `.sln` file. Simple settings are converted on TeamCity upgrade, but you might need to verify the runners contain appropriate settings.

"%" Escaping in the Build Configuration Properties

Now, two percentage signs (%%) in values defined in Build Configuration settings are treated as escape for a single percentage sign. Your existing settings are converted on upgrade to preserve functioning like in previous versions. However, you might need to review the settings for unexpected "%" sign-related issues.

.Net Framework Properties are Reported as Configuration Parameters

In previous TeamCity versions, installed .Net Frameworks, Visual Studios and Mono were reported as System Properties of the build agents.

This made the properties available in the build script.

In order to reduce number of TeamCity-specific properties pushed into the build scripts, the values are now reported via Configuration Parameters (that is, without "system." prefix) and are not available in the build script by default. They still be used in the Build Configuration settings via %-references by their previous names, just without "system." prefix.

Ipr runner is deprecated in favor of IntelliJ IDEA Project runner

Runner for IntelliJ IDEA projects was completely rewritten. It is not named "IntelliJ IDEA Project" runner. Previously available Ipr runner is also preserved but is marked as deprecated and will be removed in one of the further major releases of TeamCity. It is highly recommended to migrate your existing build configurations to the new runner.

Please note that the new runner uses different approach to run tests: you need to have a shared Run Configuration created in IntelliJ IDEA and reference it in the runner settings.

Cleanup for Inspection and Duplicates data

Starting from 6.0 Inspection and Duplicates reports for the builds are cleaned when build is cleaned from history, not when build's artifacts are cleaned as it used to be.

Inspection and Duplicates runners require Java 1.6

"Inspections" and "Duplicates (Java)" runners now require Java JDK 1.6. Please ensure Java 1.6 is installed on relevant agents and check it is specified in the "JDK home path" setting of the runners.

XML Report Validation

If you had invalid settings of "XML Report Processing" section of the build runners, you might find the Build Configurations reporting "Report paths must be specified" messages upon upgrade. In this case, please go to the runner settings and correct the configuration. (related [issue](#))

Open API Changes

See [Open API Changes](#)

Several jars in `devPackage` were reordered, some moved under `runtime` subdirectory. Please update your plugin projects to accommodate for these changes.

REST API Changes

Several objects got additional attributes and sub-elements. Please check that your parsing code still works.

Perforce Clean Checkout

All builds using Perforce checkout will do a clean checkout after server upgrade. Please note that this can impose a high load on the server in the first hours after upgrade and server can be unresponsive while many builds are in "transferring sources" stage.

Changes from 5.1.2 to 5.1.3

Path to executable in Command line runner

The [bug](#) was fully fixed. The behavior is the same as in pre-5.1 builds.

Changes from 5.1.1 to 5.1.2

Jabber notification sending errors are displayed in web UI for administrators again (these messages were disabled in 5.1.1). If you do not use Jabber notifications, please pause the Jabber notifier on the Jabber settings server settings page.

Changes from 5.1 to 5.1.1


Path to executable in Command line runner


The [bug](#) was partly fixed. The behavior is the same as in pre-5.1 builds except for the case when you have the working directory specified and have the script in both checkout and working directory. The script from the working directory is used.

Path to script file in Solution runner and MSBuild runner

The [bug](#) was fixed. The behavior is the same as in pre-5.1 builds.

Changes from 5.0.3 to 5.1

 If you plan to upgrade from version 3.1.x to 5.1, you will need to modify some dtd files in `<TeamCity Data Directory>/config` before upgrade, read more in the issue: [TW-11813](#)

 NCover 3 support may not work. See [TW-11680](#)

Notification templates change

Since 5.1, TeamCity uses [new template engine](#) (Freemarker) to generate notification messages. New default templates are supplied and customizations to the templates made prior to upgrading are no longer effective.

If you customized notification templates prior to this upgrade, please review the new notification templates and make changes to them if necessary. Old notification templates are copied into `<TeamCity Data Directory>/config/_trash/_notifications` directory. Hope, you will enjoy the new templates and new extended customization capabilities.

External database drivers location

JDBC drivers can now be placed into `<TeamCity Data Directory>/lib/jdbc` directory instead of `WEB-INF/lib`. It is recommended to use the new location. See details at [Setting up an External Database#Database Driver Installation](#).

PostgreSQL jdbc driver is no more bundled with TeamCity installation package, you will need to [install](#) it yourself upon upgrade.

Database connection properties

Database connection properties template files have changed their names and are placed into `database.<database-type>.properties.dist` files under `<TeamCity Data Directory>/config` directory. They follow [.dist files convention](#).

It is recommended to review your `database.properties` file by comparing it with the new template file for your database and remove any options that you did not customize specifically.

Default memory options change

We changed the default [memory option](#) for PermGen memory space and if you had `-Xmx` JVM option changed to about 1.3G and are running on 32 bit JVM, the server may fail to start with a message like: `Error occurred during initialization of VM Could not reserve enough space for object heap Could not create the Java virtual machine.`

On this occasion, please consider either:

- switching to 64 bit JVM. Please consider the [note](#).
- reducing PermGen memory via `-XX:MaxPermSize` JVM option (to previous default 120m)
- reducing heap memory via `-Xmx` JVM option

Vault Plugin is bundled

In this version we bundled [SourceGear Vault VCS plugin](#) (with experimental status). Please make sure to uninstall the plugin from `.BuildServer/plugins` (just delete plugin's zip) if you installed it previously.

Path to executable in Command line runner

A [bug](#) was introduced that requires changing the path to executable if working directory is specified in the runner. The bug is partly fixed in 5.1.1 and fully fixed in 5.1.3.

Path to script file in Solution runner and MSBuild runner


A [bug](#) was introduced that requires changing the path to script if working directory is specified in the runner. The bug is fixed in 5.1.1.

Open API Changes

See [Open API Changes](#)

Changes from 5.0.2 to 5.0.3

No noteworthy changes.

 There is a known issue with .NET duplicates finder: [TW-11320](#)
Please use the patch attached to the issue.

Changes from 5.0.1 to 5.0.2

External change viewers

The `relativePath` variable is now replaced with relative path of a file without checkout rules. The previous value can be

accessed via `relativeAgentPath`. More information at [TW-10801](#).

Changes from 5.0 to 5.0.1

No noteworthy changes.

Changes from 4.5.6 to 5.0

Pre-5.0 Enterprise Server Licenses and Agent Licenses need upgrade

With the version 5.0, we announce changes to the upgrade policy: Upgrade to 5.0 is not free. Every license (server and agent) bought since 5.0 will work with any TeamCity version released within one year since the license purchase. Please review the detailed information at [Licensing and Upgrade](#) section of the official site.

Bundled plugins

If you used standalone plugins that are now bundled in 5.0, do not forget to remove the plugins from `.BuildServer/plugins` directory.

The newly bundled plugins are:

- Mercurial
- Git (JetBrains)
- REST API (was provided with YouTrack previously)

Other plugins

If you use any plugins that are not bundled with TeamCity, please make sure you are using the latest version and it is compatible with the 5.0 release. e.g. You will need the latest version of [Groovy plug](#) and other properties-providing extensions. Pre-5.0 notifier plugins may lack support for per-test and assignment responsibility notifications.

Obsolete Properties

The system property "build.number.format" and environment variable "BUILD_NUMBER_FORMAT" are removed. If you need to use build number format in your build (let us know why), you can define build number format as `%system.<property name>%` and define `<property name>` system property in the build configuration (it will be passed to the build then).

Oracle database

If you use TeamCity with Oracle database, you should add an addition privilege to the TeamCity Oracle user. In order to do it, log in to Oracle as user SYS and perform

```
grant execute on dbms_lock to <TeamCity_User>;
```

PostgreSQL database

TeamCity 5.0 supports PostgreSQL version 8.3+.

So if the version of your PostgreSQL server is less than 8.3 then it needs to be upgraded.

Open API Changes

See [Open API Changes](#)

Changes from 4.5.2 to 4.5.6

No noteworthy changes.

Changes from 4.5.1 to 4.5.2

Here is a critical issue with Rake runner in 4.5.2 release. Please see [TW-8485](#) for details and a fixing patch.

Changes from 4.5.0 to 4.5.1

No noteworthy changes.

Changes from 4.0.2 to 4.5

Default User Roles

The roles assigned as default for new users will be moved to "All Users" groups and will be effectively granted to all users

already registered in TeamCity.

Running builds during server restart

Please ensure there are no running builds during server upgrade.

If there are builds that run during server restart and these builds have test, the builds will be canceled and re-added to build queue (TW-7476).

LDAP settings rename

If you had LDAP integration configured, several settings will be automatically converted on first start of the new server. The renamed settings are:

- `formatDN` — is renamed to `teamcity.auth.formatDN`
- `loginFilter` — is renamed to `teamcity.auth.loginFilter`

Changes from 4.0.1 to 4.0.2

Increased first cleanup time

The first server cleanup after the update can take significantly more time. Further cleanups should return to usual times.

During this first cleanup the data associated with deleted build configuration is cleaned. It was not cleaned earlier because of a bug in TeamCity versions 4.0 and 4.0.1.

Changes from 4.0 to 4.0.1

"importData" service message arguments

id argument renamed to type and file to path. This change is backward-compatible. See [Using Service Messages](#) section for examples of new syntax.

Changes from 3.1.2 to 4.0

Initial startup time

On the very first start of the new version of TeamCity, the database structure will be upgraded. This process can increase the time of the server startup. The first startup can take up to 20 minutes more than regular one. This time depends on the size of your builds history, average number of tests in a build and the server hardware.

Users re-login will be forced after upgrade

Upon upgrade, all users will be automatically logged off and will need to re-login in their browsers to TeamCity web UI. After the first login since upgrade, Remember me functionality will work as usual.

Previous IntelliJ IDEA versions support

IntelliJ IDEA plugin in this release is no longer compatible with IntelliJ IDEA 6.x versions. Supported IDEA versions are 7.0.3 and 8.0.

Using VCS revisions in the build

`build.vcs.number.N` system properties are replaced with `build.vcs.number.<escaped VCS root name>` properties (or just `build.vcs.number` if there is only one root). If you used the properties in the build script you should update the usages manually or switch compatibility mode on. References to the properties in the build configuration settings are updated automatically. Corresponding environment variable has been affected too.

[Read more.](#)

Test suite

Due to the fact that TeamCity started to handle tests suites, the tests with suite name defined will be treated as new tests (thus, test history can start from scratch for these tests.)

Artifact dependency pattern

Artifact dependencies patterns now support [Ant-like wildcards](#).

If you relied on `***` pattern to match directory names, please adjust your pattern to use `*/` instead of single `*`.

If you relied on the `***` pattern to download only the files without extension, please update your pattern to use `.*` for that.

Downloading of artifacts with help of Ivy

If you downloaded artifacts from the build scripts (like Ant `build.xml`) with help of Ivy tasks you should modify your `ivyconf.xml` file and remove all statuses from there except "integration". You can take the `ivyconf.xml` file from the following page as reference: <http://www.jetbrains.net/confluence/display/TCD4/Configuring+Dependencies>

Browser caches (IE)

To force Internet Explorer to use updated icons (i.e. for the Run button) you may need to force page reload (Ctrl+Shift+R) or delete "Temporary Internet Files".

Changes from 3.1.1 to 3.1.2

No noteworthy changes.

Changes from 3.1 to 3.1.1

No noteworthy changes.

Changes from 3.0.1 to 3.1

Guest User and Agent Details

Starting from version 3.1, the Guest user does not have access to the agent details page. This has been done to reduce exposing potentially sensitive information regarding the agents' environment. In the Enterprise Edition, the Guest user's roles can be edited at the Users and Groups page to provide the needed level of permission.

StarTeam Support

Working Folders in Use

Since version 3.1 when checking out files from a StarTeam repository TeamCity builds directory structure on the base of the working folder names, not just folder names as it was in earlier versions. So if you are satisfied with the way TeamCity worked with StarTeam folders in version 3.0, ensure the working folders' names are equal to the corresponding folder names (which is so by default).

Also note, that although StarTeam allows using absolute paths as working folders, TeamCity supports relative paths only and doesn't detect absolute paths presence. So be careful and review your configuration.

StarTeam URL Parser Fixed

In version 3.0 a user must have followed a wrong URL scheme. It was like `starteam://server:49201/project/view/rootFolder/subfolder/...` and didn't work when user tried to refer a non-default view. In version 3.1 the native StarTeam URL parser is utilized. This means you now don't have to specify the root folder in the URL, and the previous example should look like `starteam://server:49201/project/view/subfolder/...`

Changes from 3.0 to 3.0.1

Linux Agent Upgrade

- Due to an issue with Agent upgrade under Linux, Agent auxiliary Launcher processes may have been left running after agent upgrades. Versions 3.0.1 and up fix the issue. To get rid of the stale running processes, after automatic agent upgrade, please stop the agent (via `agent.sh kill` command) and kill any running `java jetbrains.buildServer.agent.Launcher` processes and start the agent again.

Changes from 2.x to 3.0

Incompatible changes

Please note that TeamCity 3.0 introduces several changes incompatible with TeamCity 2.x:

- `build.working.dir` system property is renamed to `teamcity.build.checkoutDir`. If you use the property in your build scripts, please update the scripts.
- `runAll.bat` script now accepts a required parameter: `start` to start server and agent, `stop` to stop server and agent.
- Under Windows, `agent.bat` script now accepts a required parameter: `start` to start agent, `stop` to stop agent. Note that in this case agent will be stopped only after it becomes idle (no builds are run). To force immediate agent stopping, use `agent.bat stop force` command that is available under both Windows and Linux (`agent.sh stop force`). Under Linux you can also use `agent.sh stop kill` command to stop agents not responding to `agent.sh stop force`.

Build working directory

Since TeamCity 3.0 introduces ability to configure VCS roots on per-Build Configuration basis, rather than per-Project, the default directory in which build configuration sources are checked out on agent now has generated name. If you need to know the directory used by a build configuration, you can refer to `<agent home>/work/directory.map` file which lists build configurations with the directory used by them. See also [Build Checkout Directory](#)

User Roles when upgrading from TeamCity 1.x/2.x/3.x Professional to 3.x Enterprise

When upgrading from TeamCity 1.x/2.x/3.x Professional to 3.x Enterprise for the first time TeamCity's accounts will be assigned the following roles by default:

- Administrators become System Administrators
- Users become Project Developers for all of the projects
- The Guest account is able to view all of the projects
- Default user roles are set to Project Developer for all of the projects

Changes from 1.x to 2.0

Database Settings Move

Move your database settings from the `<TeamCity installation folder>/ROOT/WEB-INF/buildServerSpring.xml` file to the `database.properties` file located in the TeamCity configuration data directory (`<TeamCity Data Directory>/config`).

See also:

[Administrator's Guide: Licensing Policy](#)

.NET inspection and duplicates

inherited



If you were using the TeamCity-GitHub [third-party plugin](#) prior to TeamCity 10.0, you can safely **remove** it: the built-in TeamCity integration will detect the existing connection to GitHub issue tracker and pick up your settings automatically.

`teamcity.TruncateIgnoreReasonConverter.copyReasons`

```
Error
  collecting changes
for
  VCS repository
...

Failed
  to collect TFS changes
-

From
  version x
is
  greater
then
  current version y
```

The [bug with temp tool folders|<https://youtrack.jetbrains.com/issue/TW-46648>] introduced in the previous version has been fixed.