TeamCity

JetBrains TeamCity Comparison

TeamCity is a continuous integration and continuous delivery server developed by JetBrains. It provides out-of-the-box continuous unit testing, code quality analysis, and early reporting on build problems. TeamCity supports Java, .NET and Ruby development and integrates perfectly with major IDEs, version control systems, and issue tracking systems.

Key benefits

Easy initial setup and **low maintenance costs**, thanks to all main functionality being available out of the box.

Time savings on regular and repetitive build engineering and developer tasks due to intelligent features and deep integration with and understanding of the development process.

Adjusts to multiple development styles and frameworks so your **teams can perform better.**

Low entry barrier thanks to informative and rich interfaces.

Extendable architecture to accommodate to your organization's special needs.

Ready to scale as your organization grows, with advanced scalability features and appropriate licensing model.

Professional support. The product is being continuously developed and regularly updated by a team of 20+ JetBrains developers. You receive professional enterprise level support from the development team via both public and private channels.

Learn more about TeamCity at **jetbrains.com/teamcity**

Key unique features

TeamCity is a mature solution providing all of the standard functionality of a CI and CD server, as you can see from the table below. In addition, it offers a number of unique features, derived from JetBrains' deep expertise in the area, for a much more convenient experience unmatched by the competition.

Projects hierarchy allows you to group builds in a tree structure, helping not only better represent the structure of your organization, but also to share common settings or enforce common practices. It also helps simplify maintenance of large installations by making it possible to delegate administration tasks to development teams.

Reuse settings with build configuration templates and configuration parameters. Reuse a set of build steps with meta-runners.

Real-time tracking (as builds are still running) shows new commits appearing in a code base, newly failed tests or new problems reported, to help resolve problem faster and keep the main code base 'green'.

Test failure analysis allows you to quickly find the first build where test started to fail as well as changes which caused this failure, view history of test execution, and determine flaky (unstable) tests.

Sophisticated **integration with version control systems** allows to check-out only specific parts of the code base, check-out from several repositories at once, and view or download diffs right from the web interface.

Configure **dependencies between builds**, pass parameters or artifacts from build to build, ensure that dependent builds get the same snapshot of the code base, and prevent unnecessary rebuilding of dependencies.

Integration with IDEs (IntelliJ IDEA, Eclipse, Visual Studio) lets you test local changes on the CI server before committing them into version control. It works great with centralized VCS systems, such as Subversion, Perforce, or TFS.

Built-in **static code analysis**, based on powerful IntelliJ IDEA and ReSharper engines, lets you track code base health and fail builds if it gets worse.

Built-in **code coverage** engines based on IntelliJ IDEA and JetBrains dotCover.

Built-in **statistics engine** allows you to configure charts, report custom statistic values, or fail builds if statistical values get out of the specified range.

Built-in **health reporting** and settings suggestions help you quickly identify problems with server configuration and opt for more efficient project configuration.

Powerful **extensibility features** allow you to write custom plugins with the help of Java API, or easily report custom statistical values and tests from a custom testing framework with the help of service messages — no plugin writing required.

How TeamCity Compares to Other CI/CD Tools

JetBrains has extensively researched various Continuous Integration and Continuous Deployment tools to come up with a comparison table. Based on our research, we think it makes sense to compare TeamCity only to Jenkins as the most popular and feature-rich alternative to TeamCity.

We tried to make the comparison as comprehensive and neutral as we possibly can. As both of the products mentioned in the document are being actively developed and their functionality changes on a regular basis, this comparison applies to specific releases indicated in the table headers.

Please note that TeamCity is being compared to Jenkins — the free and open-source tool — not the Cloudbees Jenkins Platform, which is an enterprise solution by Cloudbees based on Jenkins OSS.

Feature	TeamCity 2017.1	Jenkins 2.0
Installation and setup		
Distribution type	Standalone	Standalone
Supported platforms	Windows, Linux, macOS, JavaEE	Windows, Linux, macOS
Distribution packages	.exe, .tar.gz, .war	.exe, .pkg, .deb, .rpm
Docker image available?	✓	✓
Version control support		
Git	✓	✓*
Mercurial	✓	✓*
Subversion	✓	✓
Perforce	✓	✓*
TFVC	✓	✓*
CVS	✓	✓*
StarTeam	✓	✓*
ClearCase	✓	✓*

** — marks functionality available via external plugins. Installation, maintenance, and upgrades require additional effort.*

Feature	TeamCity 2017.1	Jenkins 2.0
VCS interoperability		
Storing projects settings in VCS	✓ (Git, Mercurial, SVN, Perforce, TFS) — per project	✓ (Git, SVN) — entire server only
Feature branches	✓	✓*
Advanced VCS build triggering options	✓	×
Checking out multiple repositories in a single build	✓	Not fully
Partial sources checkout	✓ (via checkout rules)	×
Automatic merge	✓ (Git, Mercurial)	✓* (Git)
VCS labelling (tagging)	✓	✓*
View diff on changed files	✓	✓*
Continuous integration		
Remote run and pre-tested commit	✓ (for Subversion, Perforce, TFS)	×
Real-time build progress reporting	✓	×
Flaky tests reporting	✓	✓*
Newly failed tests detection	✓	×
Assign build problem investigations to team members	✓	✓*
Risk group test reordering	✓	×
Run builds with custom parameters	✓	✓
Build promotion	✓	✓*
Historical builds	✓	×
Customizable notifications	Email, Jabber, in IDE*, Windows Tray notifier*, Slack*, browser*	Email*, Jabber*, Windows tray*, Slack*, browser*

Feature	TeamCity 2017.1	Jenkins 2.0
Cloud integrations		
Amazon EC2	✓	✓*
Microsoft Azure	✓	✓*
VMware vSphere	✓	✓*
Google Cloud	×	✓ (artifact storage)
Configuration		
Projects hierarchy (display and settings propagation)	✓	×
Artifact dependencies	✓	✓
Build chains (pipeline)	✓	✓*
Reuse of the same build in different pipelines	✓	×
Build templates	✓	✓*
Programmatic project creation (configuration DSL)	✓ (Kotlin-based DSL)	✓* (Groovy-based DSL)
Build failure conditions	✓	✓*
Shared (lockable) resources	✓	✓*
Parameters in builds	✓	✓* (no password parameters)
Automatic files cleanup on agents	✓	✓*
Built-in NuGet feed	✓	×
Embedded artifacts storage	✓	✓
Issue tracker integrations	JIRA, YouTrack, GitHub, TFS, Bitbucket	JIRA*, GitHub*, TFS*, Bitbucket*

* — *marks functionality available via external plugins. Installation, maintenance, and upgrades require additional effort.*

Feature	TeamCity 2017.1	Jenkins 2.0
Build infrastructure		
Multiple build agents	✓	✓
Agent pools	✓	×
Installing tools on agents	✓	×
Agents auto-update to a new version	✓	✓ (Linux only)
Scalability		
Ability to connect multiple servers via common cross-server navigation	✓	✓*
Two-node cluster configuration	✓	✓*
Automatic switch to failover server	×	✓* (paid plugin)
Agent-side checkout	✓	✓
Server-side checkout	✓	×
Commit hooks in REST API	✓	✓*
Builds history and monitoring		
Storing build data	✓	✓
Test history	✓	✓
Statistics reports and dashboard	✓	✓
Build tags	✓	✓
Favorite builds	✓	×
Project change log	✓	✓*
Structured build log	✓	×

* — *marks functionality available via external plugins. Installation, maintenance, and upgrades require additional effort.*

Feature	TeamCity 2017.1	Jenkins 2.0
Code quality tracking (for Java, JavaScript, .NET)		
First-class integration with JetBrains code analysis tools	✓	×
Duplicate code analysis	✓ (bundles IntelliJ IDEA and ReSharper code analysis tools)	✓*
Static code analysis	✓ (based on IntelliJ IDEA and ReSharper inspections)	✓*
Code coverage	✓	✓*
Code quality reports in IDE	✓	×
Extensibility and Customization		
REST API	✓	✓
Service messages (custom testing frameworks integration)	✓	×
Meta-runners (reusable sets of build steps)	✓	×
Number of third-party plugins	300+	1000+
System maintenance		
Backup and restore	✓	✓
Projects import	✓	✓*
Projects export	✓	× (only jobs)
Disk usage report	✓	✓*
Build time report	✓	✓*
Centralized server health report	✓	×
Automatic builds history cleanup	✓	✓*

* — *marks functionality available via external plugins. Installation, maintenance, and upgrades require additional effort.*

Feature	TeamCity 2017.1	Jenkins 2.0
User management		
User actions audit log	✓	✓*
LDAP integration	✓	✓*
User roles	✓	✓*
Group roles	✓	✓*
Social (as of April 2017)		
Stackoverflow (# of questions)	TeamCity: 13,772	Jenkins: 57,805
Twitter followers	@TeamCity: 6,200 @JetBrains: 56,200	@jenkinsci: 26,600
Technical support		
Public channels	Online documentation, public issue tracker, forum, Early Access Program for pre-release builds	Public wiki, public issue tracker, IRC channel, mailing lists
Private channels	Support via email (for Enterprise customers), phone support for critical cases	No (email support only with purchase of Cloudbees Jenkins Platform)
Pricing		
Enterprise version	From $1,999 for 3 build agents to $21,999 for 100 build agents	Jenkins is free and open-source**
Free version available?	Yes, for 20 build configurations (jobs) and 3 build agents. Full functionality.	
Additional licensing options	$299 per Additional Build Agent (adds 10 build configurations)	

** — marks functionality available via external plugins. Installation, maintenance, and upgrades require additional effort.*

*** Paid enterprise version of Jenkins is available from Cloudbees - pricing on request. Some of the plugins are also commercial.*

Accuracy of Comparison

We tried to make it as comprehensive and neutral as we possibly can. If you discover any inaccuracies in this table, please contact us at sales@jetbrains.com and we'll update it as soon as possible.

Integration with JetBrains Tools

TeamCity is a part of the JetBrains team tools stack, which additionally includes Upsource, a code review and repository browsing tool, and YouTrack, an issue tracking and agile project management tool. Team tools are integrated via Hub, permission and user management tool, which ensures single sign-on to all team tools and unified user and permission management.

To learn more about how JetBrains team tools work together, please check our website at **jetbrains.com/hub**.

Sales Contacts

If you need assistance with managing your licenses, selecting a licensing option, requesting assistance with a JetBrains offer, or any special request or suggestion for JetBrains tools, please do not hesitate to **contact us**.

Email:
sales@jetbrains.com

Phone:
USA: +1 888 672 1076
Europe and global: +420 2 4172 2501