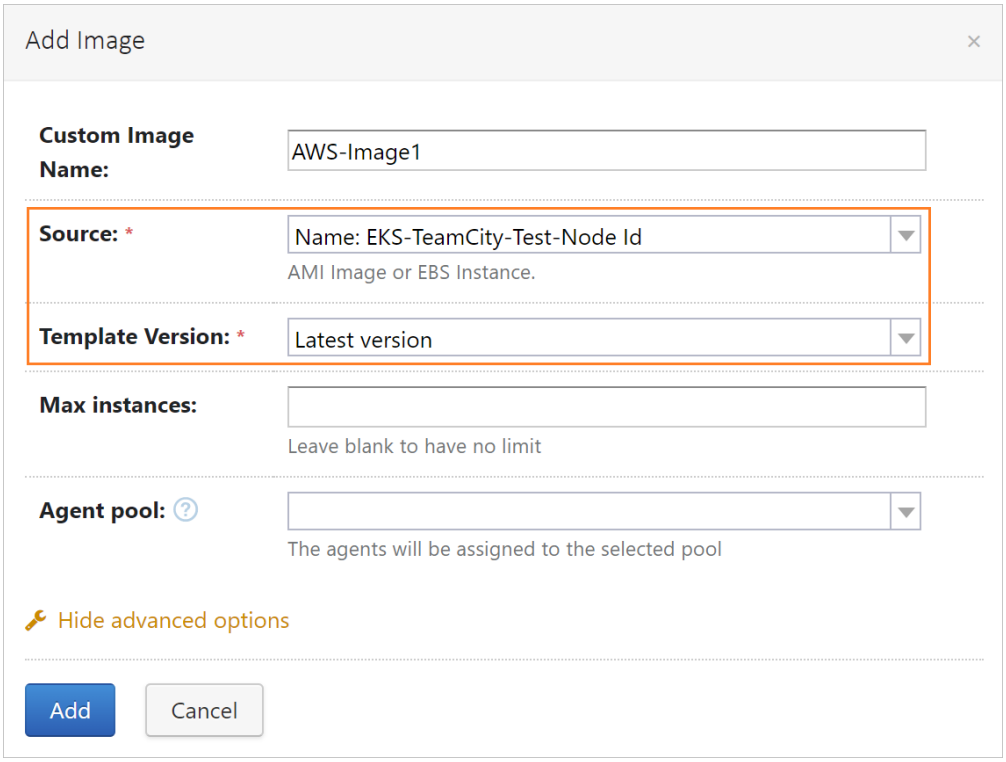
Kanpur 2019.2 EAP1 (build 70479) Release Notes

On this page:

Support of EC2 launch templates

TeamCity now supports Amazon EC2 launch templates for cloud instances. Many of the EC2 users appreciate the launch templates since they allow reusing a once defined launch specification for all new instances, which eliminates the need to describe the launch settings every time new instances are requested.

When you are adding an image in the TeamCity cloud profile connected to the Amazon server, TeamCity automatically detects images and instances available on this server. And now, it also detects existing launch templates. Simply select a template as a Source and specify its version, and TeamCity will request instances based on the template parameters.

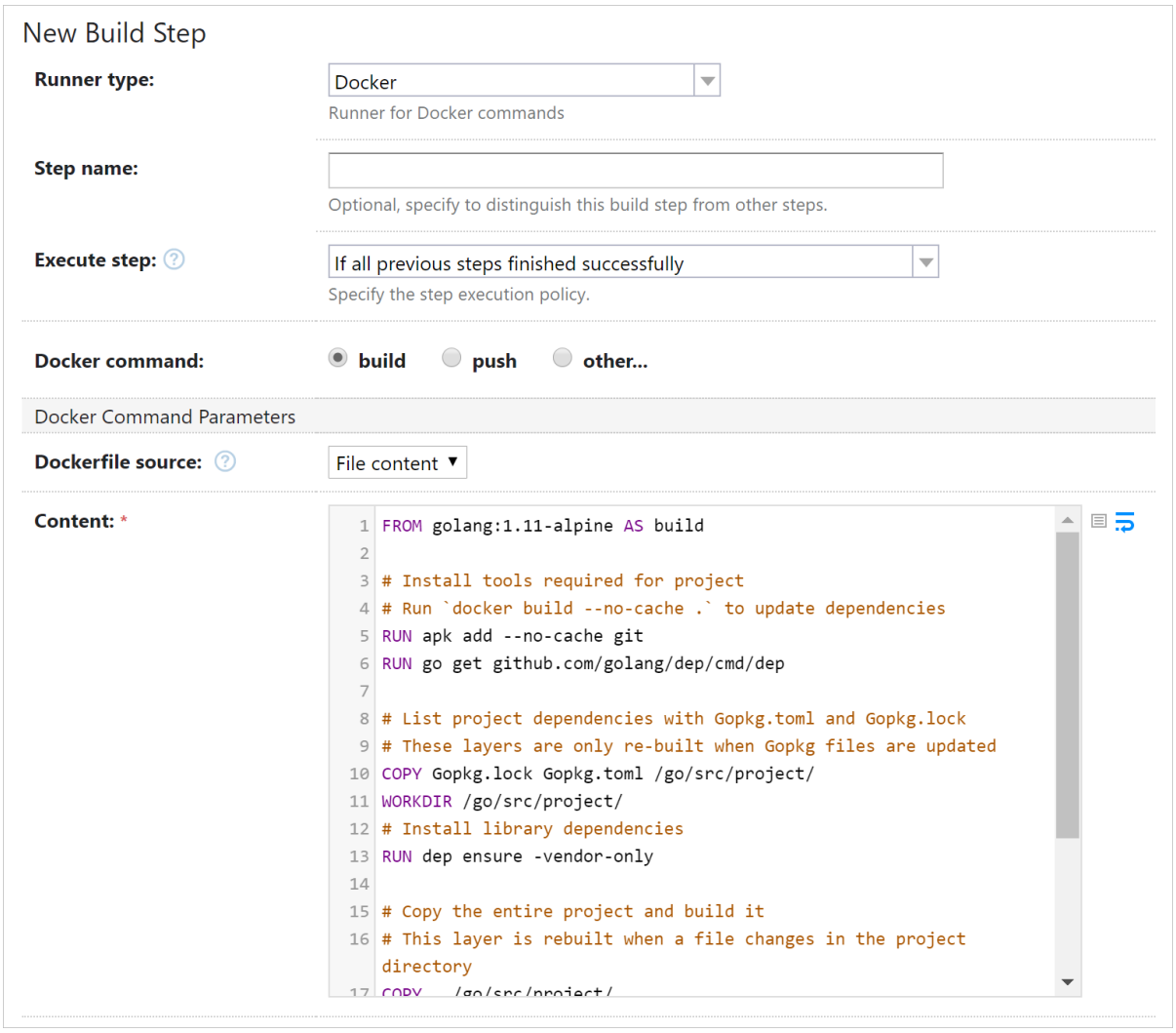


Code highlighting in build runners' scripts

We've added automatic code highlighting and line numbering for scripts of the following build runners: Command Line, Ant, PowerShell, Docker, NAnt, and Rake, as well as for the configuration of Amazon EC2 images.

To improve readability, you can also apply soft wraps to the code by clicking next to the input form.

Example of Dockerfile highlighting in TeamCity:



New features of experimental UI

The following features are currently available in the TeamCity experimental UI. To switch to the experimental UI, click in the upper right corner of the page.

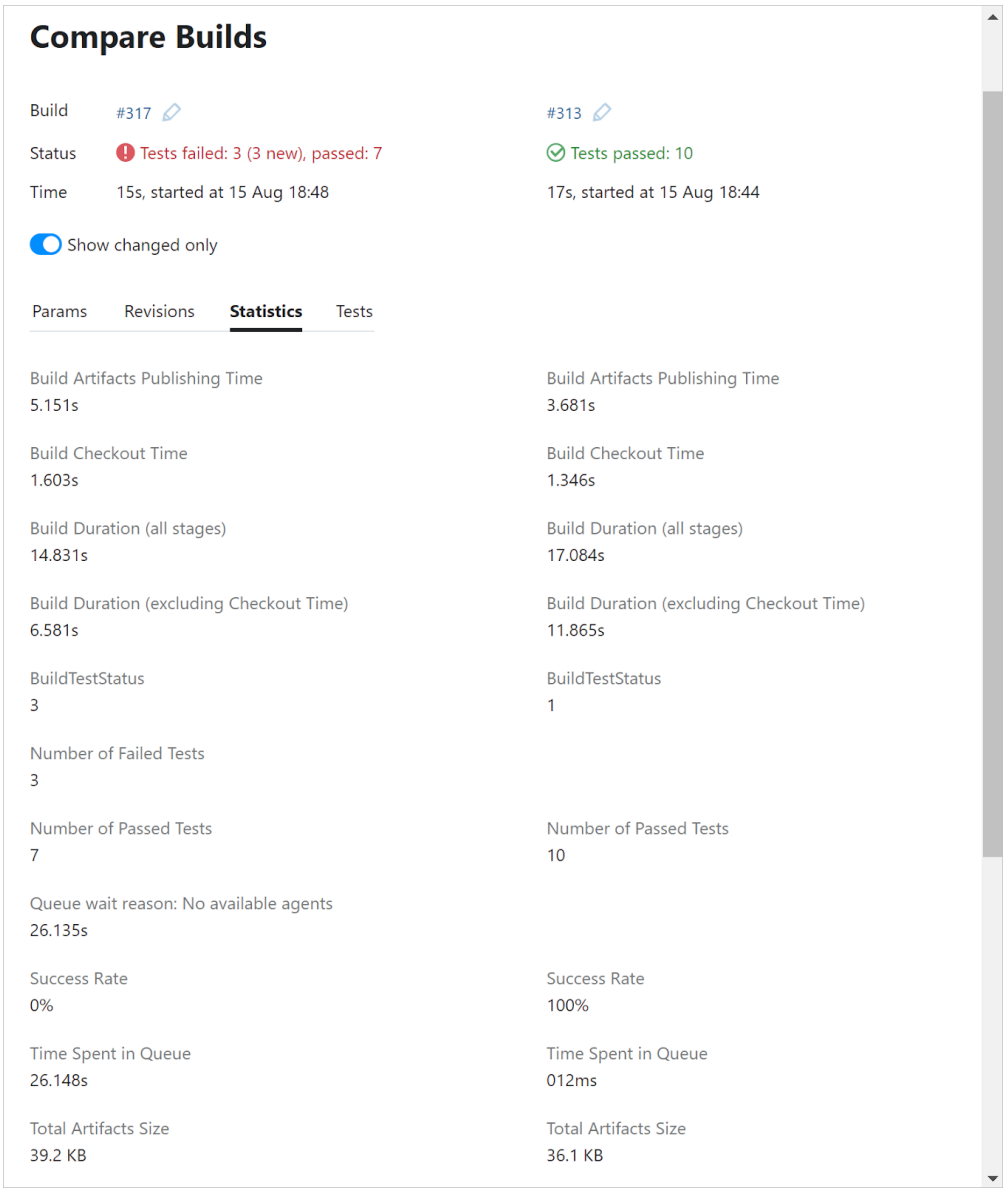
Comparing builds

With the Compare Builds feature, you can select two builds from the same configuration and review their details side-by-side:

- build environment and parameters, such as source branches, revisions, and agents used for running builds
- statistics
- failed, ignored, and passed tests

This feature allows for easier monitoring and is especially helpful when multiple users manage and monitor builds. For example, if a build has no changes in the project code but fails for no obvious reason, you can compare this build with the last successful build and analyze their differences to find the most probable cause of the failure.

To compare a finished build with another, open the action menu of this build in the build list, click Compare with, and select a build for comparison.

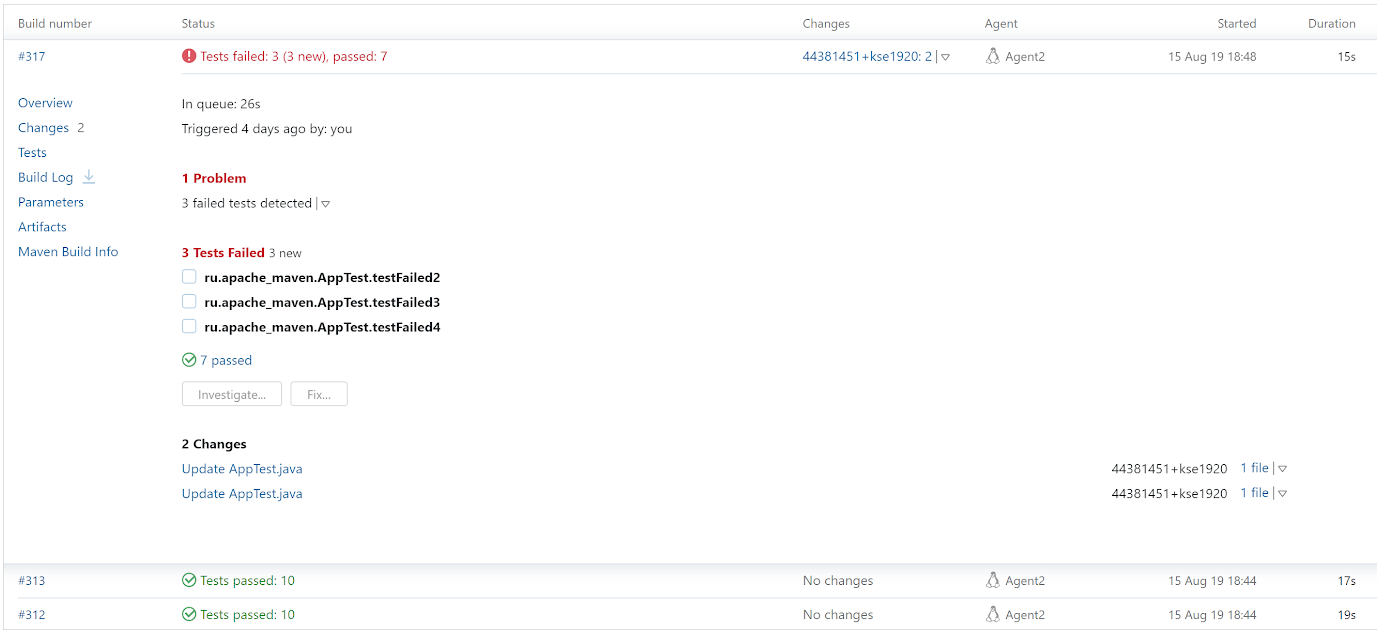


Extended build preview in build list

TeamCity now allows you to preview the most important build results directly in the list of builds. On the Build Configuration Home page, click on a build in the list to see the following details:

- all build changes
- build problems and failed tests
- snapshot dependencies

With this preview, you also receive quick access to any tab of the build results.



Other improvements

- Optimized multinode setup
  Since this EAP, agents can load server-side patches from secondary nodes – not only from the main server, as it was before.
  Server-side patches are mostly used when an agent cannot find a VCS client executable (for example, Git or Perforce) on an agent machine. In this case, the agent will request the server to create a patch with VCS changes and send it to the agent. Now, if you assign the "VCS repositories polling" responsibility to a secondary node, the agents will be able to request patches from this node as well, which significantly reduces the load on the main server.
- Improved Docker integration

  Once the Docker Compose build runner creates a Docker image on an agent, it is stored on this agent to be used for the later builds. Now, TeamCity will automatically detect that the Dockerfile, used for building the image, has changed and will force the Docker Compose step to rebuild this image.
- All fixed issues

Notes on discontinued Running Builds Node

The Running Builds Node is no longer supported since this EAP. In a multinode setup, you can instead configure a secondary node with the "Processing data produced by running builds" responsibility.

Note that the secondary node offers more features than the Running Builds Node and thus might require as many hardware resources as a regular TeamCity server. Refer to Estimate Hardware Requirements for TeamCity for notes on the recommended hardware.