Stages & Actions

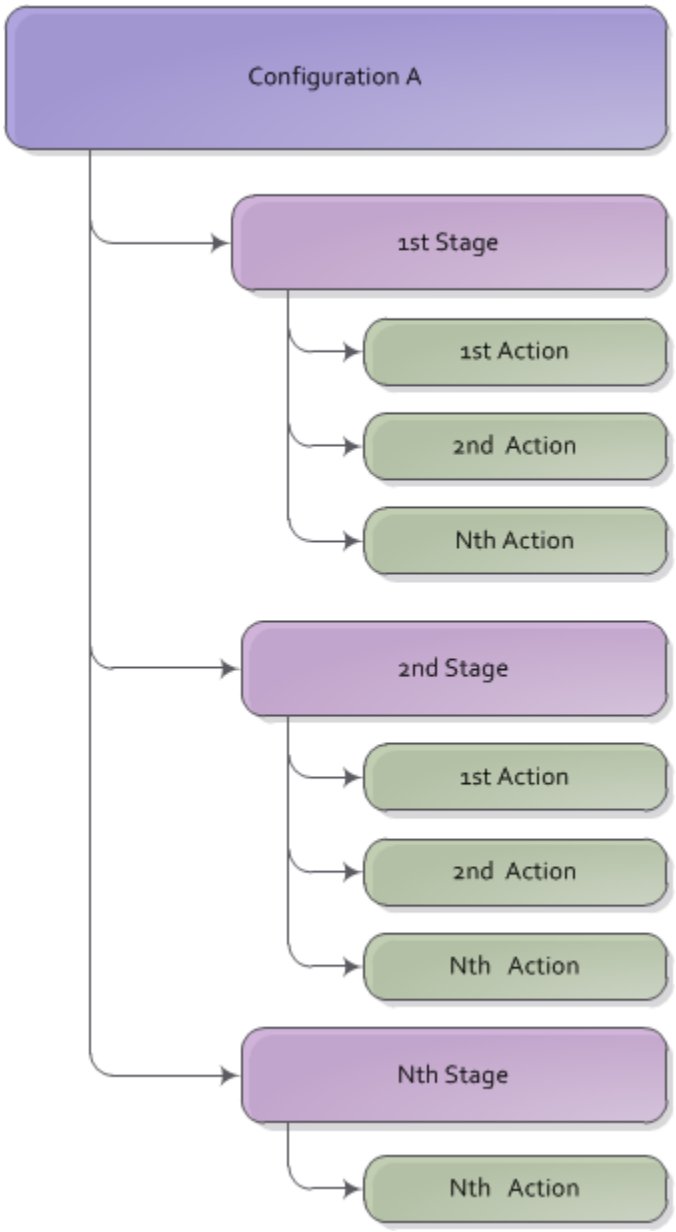
Stages

Stages are a mechanism in Continua CI that allows you to break apart your build process into multiple steps. Each configuration can contain multiple stages and each stage contains the individual actions that run during the build (see figures 1 and 2 for more details).

Stages can be set to either automatically continue to the next stage or wait for user confirmation.

When a stage is run, its Workspace and Repository rules are automatically fired. These rules determine which files are transferred between the server and agent, and which files are transferred between the repository and agent respectively. For more information, view the Workspace Rules and Repository Rules pages.

Figure 1. Relationship between Configurations, Stages and Actions.

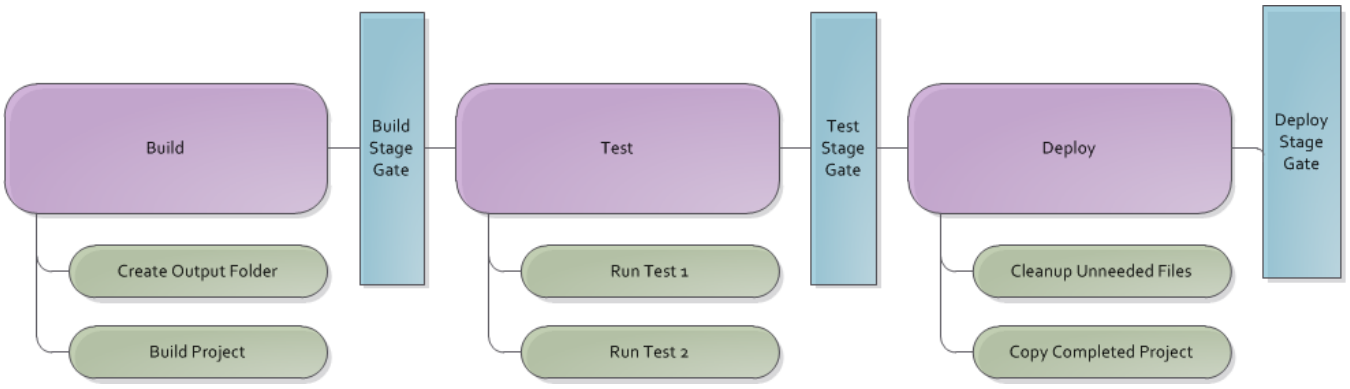


Stage Gates

Stage gates are coupled with stages and act as a road block after the completion of each stage. Stage gates allow you to define a list of conditions that must be met after that stage has finished. The build will only continue if these conditions are met. For example, if you have a test stage that runs tests on your compiled project then you can set the test stage gate to only continue the build if all tests completed successfully.

Continua CI comes with default stage gate conditions which can be summarized as "Continue to the next Stage if there were no Failed or Errored Unit Tests".

Figure 2. An Example Build Stages Pipeline



Actions

An action is a single step within a stage that usually performs a very simple task. These actions are the steps that actually perform your build process. Most actions in Continua CI act as a wrapper around an executable provided by the user. The built-in actions in Continua CI range from advanced actions like MSBuild, FinalBuilder and Ant to simple actions like copying or deleting a file. Actions can also be of the flow control type which dictate the execution flow of that particular stage. For a full list of actions see Actions.