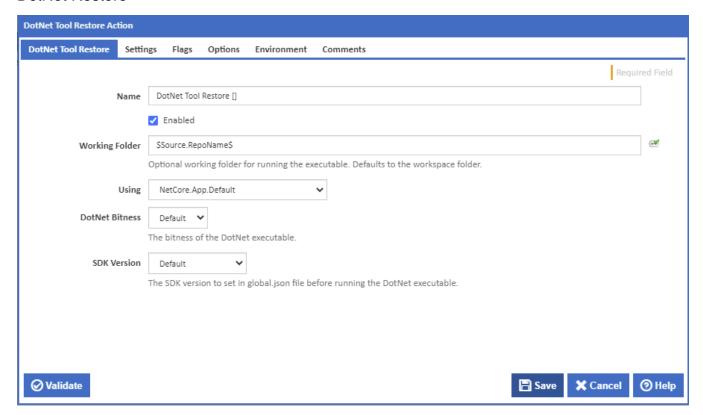# DotNet Tool Restore Action

> ⓘ The DotNet Tool Restore action in Continua CI is a wrapper around the .Net Core command line tools. If you're having trouble using the DotNet Restore action, please refer to the .NET Core Command Line Tools documentation.

The DotNet tool Restore action is used to restore .Net tools according to a manifest file.

DotNet Restore

Name

A friendly name for this action (will be displayed in the actions workflow area).

Enabled

Determines if this action will be run within the relevant stage.

Working Folder

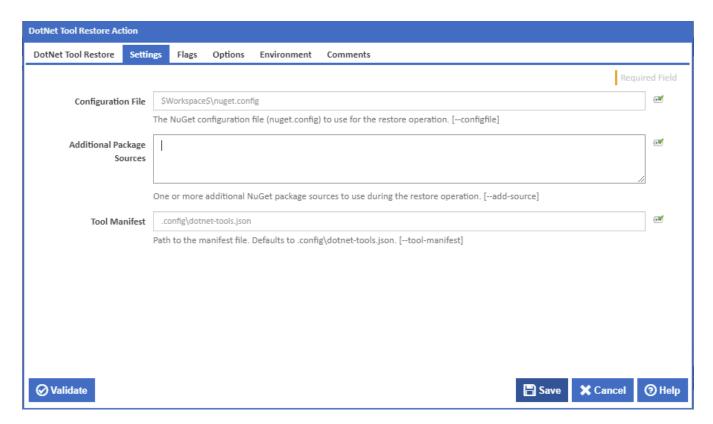Optional working folder for running the executable. Defaults to the project folder.

Using

The Using drop down is populated with any property collector whose namespace matches the pattern defined by the DotNet CLI actions. The pattern for this action is `^DotNet.Cli.*`

If you create a property collector for this action, make sure you select the **Path Finder PlugIn** type and give it a name that will match the pattern above in blue. Example names listed here, search the table's Plugin column for "**DotNet Restore**".

For more in-depth explanations on property collectors see Property Collectors.

Alternatively, you can select the **Custom** option from the Using drop down list and specify a path in the resulting input field that will be displayed. Please read Why it's a good idea to use a property collector before using this option.

Settings

DotNet Tool Restore   **Settings**   Flags   Options   Environment   Comments

| Required Field

Configuration File   `$Workspace$\nuget.config`

The NuGet configuration file (nuget.config) to use for the restore operation. [--configfile]

Additional Package Sources   `|`

One or more additional NuGet package sources to use during the restore operation. [--add-source]

Tool Manifest   `.config\dotnet-tools.json`

Path to the manifest file. Defaults to .config\dotnet-tools.json. [--tool-manifest]

⊘ Validate   💾 Save   ✕ Cancel   ⊘ Help

Configuration File

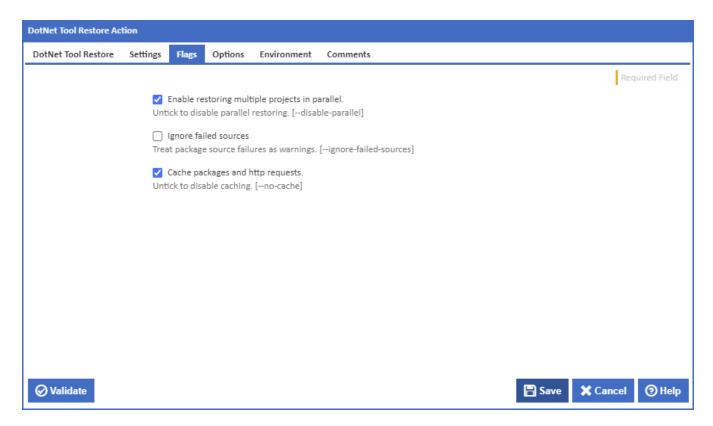The NuGet configuration file (NuGet.config) to use for the tool restore operation. [--configfile]

Additional Package Sources

One or more additional NuGet package sources to use during the tool restore operation. [--add-source]

Tool Manifest

Path to the tool manifest file which specifies while tool version to restore. If left blank then the dotnet executable will search for dotnet-tools.json in the working folder and .config subfolder. [--tool-manifest]

Flags

DotNet Tool Restore   Settings   **Flags**   Options   Environment   Comments

Required Field

☑ Enable restoring multiple projects in parallel.
Untick to disable parallel restoring. [--disable-parallel]

☐ Ignore failed sources
Treat package source failures as warnings. [--ignore-failed-sources]

☑ Cache packages and http requests.
Untick to disable caching. [--no-cache]

⊘ Validate   🖫 Save   ✖ Cancel   ⊙ Help

Enable restoring multiple projects in parallel

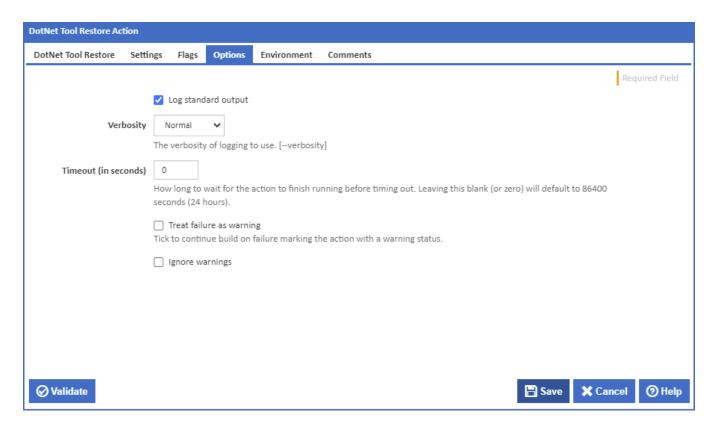If this is ticked, multiple restore operations are run concurrently. [--disable-parallel]

Ignore failed sources

If this is ticked, DotNet Tool Restore will treat failed sources as warnings. [--ignore-failed-sources]

Cache packages and HTTP requests

If this is ticked, packages and HTTP requests are cached. [--no-cache]

Options

Log standard output

If this is ticked, the command line output is written to the build log.

Verbosity

The amount of information detail to display in the build log. [--verbosity]

Timeout (in seconds)

How long to wait for the action to finish running before timing out. Leaving this blank (or zero) will default to 86400 seconds (24 hours).
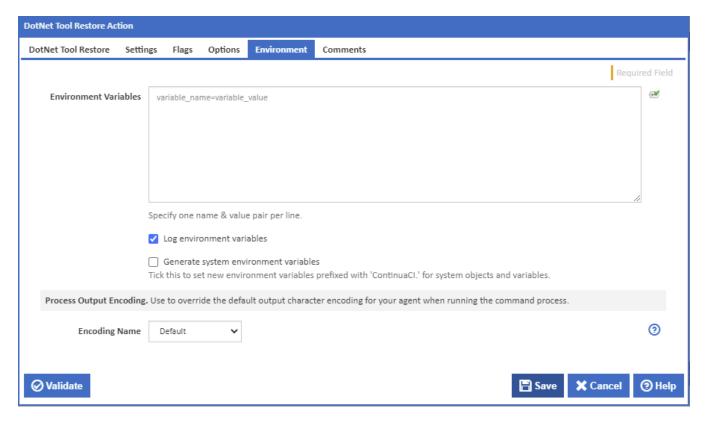
Treat failure as warning

Tick to continue build on failure marking the action with a warning status.

Ignore warnings

If this is ticked, any warnings logged will not mark the action with a warning status.

Environment

Environment Variables

Multiple environment variables can be defined - one per line. These are set before the command line is run.

Log environment variables

If this is ticked, environment variable values are written to the build log.

Generate system environment variables

Tick this checkbox to set up a list of new environment variables prefixed with 'ContinuaCI.' for all current system expression objects and variables.

Mask sensitive variable values in system environment variables

This checkbox is visible only if the '**Generate system environment variables**' checkbox is ticked.

If this is ticked, the values of any variables marked as sensitive will be masked with **** when setting system environment variables. Clear this to expose the values.