

# XUnit Action



The XUnit action in Continua is a wrapper around the XUnit.Runner.Console.exe command line. If you're having trouble using the XUnit action, please refer to the [Command Line Reference](#).

The XUnit action can be used to run automated unit tests.

## XUnit

**XUnit Action**

XUnit

Include

Exclude

Options

Environment

Comments

Required Field

Name

XUnit []

☒ Enabled

Results File

Output\Tests\testresults.xml

The path and file name of the test output. This path is relative to the workspace and the file must end in .xml. Leave blank to use the name of the first assembly. [-xml]

Assembly File Patterns

\$Workspace\Output\\*\*Tests.dll

File names or patterns to match assembly files to test.

XUnit Version

Version 2.4

The version of XUnit being used to run the tests.

☐ Install XUnit on agent using NuGet

Using

XUnit.x86

Validate

Save

Cancel

Help

### 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.

### Results File

The path and file name of the test output. This path is relative to the workspace and the file must end in .xml (it will be appended if not). Leave blank to use the name of the first assembly. [-xml]

### Assembly File Patterns

The file names or patterns to match assembly files to test.

### Install XUnit on agent using NuGet

Tick this to automatically install XUnit on the agent using NuGet. Additional fields will be shown allowing you to enter the "Platform" and installation folder "Install Xunit To". A new **NuGet** tab will also be displayed with options for the NuGet command line.

### XUnit Version

The version of Xunit to use to run the tests. Choose Custom if the version is not listed - if you are installing using NuGet then a "XUnit Runner Version" field will be displayed for you to enter the version to install.

## XUnit Runner Version

Visible only if the "Install Xunit on agent using NuGet" is ticked.

The Xunit Runner Version.

## Using

When not installing using NuGet, the "Using" drop down is populated with any property collector whose namespace matches the pattern defined by the Xunit action. The pattern for this action is `^XUnit\.*`

The default property collector will search the environment PATH for "xunit.console.exe". 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 "XUnit".

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.

## NuGet

XUnit Action

XUnit

NuGet

Include

Exclude

Options

Environment

Comments

Required Field

Install Xunit To

\$Workspace\$\XUnitInstallation

Sources

source\_a;source\_b;source\_c

A list of packages sources to use for the install. Delimiter is semi-colon. Optional.

☐ Install prerelease packages

☒ Use cache

Use the machine cache as the first package source.

Platform

Default

Target Framework

.Net Framework 4.7.2

Used to select a targeted Xunit executable to run the tests.

Using NuGet

NuGet.Default

Validate

Save

Cancel

Help

### Install Xunit To

The folder NuGet should install Xunit to.

### Sources

A list of package sources to install the Xunit package from. Optional. You can separate multiple sources with semi-colons.

## Install prerelease packages

Optionally install prerelease versions of the XUnit package.

## Use cache

Optionally attempt to source the XUnit package from the NuGet machine cache.

## Platform

The platform which you are installing XUnit for. Choose Default to install the x86 version on 32-bit Windows or the x64 version on 64bit Windows. Choose XUnit.x86 to install the 32bit version on 64bit Windows.

## Target Framework

The targeted XUnit executable to run the tests.

## Using

The Using drop down is populated by any property collector properties whose namespace matches the pattern defined by the NuGet action. The pattern for this action is `^NuGet\.*`. The default property collector searches the environment path for "NuGet.exe".

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.

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.

## Include

XUnit Action

XUnit

Include

Exclude

Options

Environment

Comments

Required Field

Include Traits

Only run tests with any of the matching name/value traits. One per line. [-trait]

Include Methods

MyNamespace.MyClass.MyTestMethod

Only run any of the given test methods. One per line. Each method can be fully specified or use a wildcard. i.e. 'MyNamespace.MyClass.MyTestMethod' or '\*.MyTestMethod'. [-method]

Include Classes

MyNamespace.MyClass

Only run any of the given test classes. One class per line. Each class should be fully specified. i.e. 'MyNamespace.MyClass. [-class]

Include Namespaces

MyNamespace.MySubNamespace

Only run any of the given test namespaces. One per line. i.e. 'MyNamespace.MySubNamespace. [-namespace]

Validate

Save

Cancel

Help

### Include Traits

Only run tests with any of the matching name/value traits. One per line. [-trait]

### Include Methods

Only run any of the given test methods. One per line. Each method can be fully specified or use a wildcard. i.e. 'MyNamespace.MyClass.MyTestMethod' or '\*.MyTestMethod'. [-method]

### Include Classes

Only run any of the given test classes. One class per line. Each class should be fully specified. i.e. 'MyNamespace.MyClass'. [-class]

### Include Namespaces

Only run any of the given test namespaces. One per line. i.e. 'MyNamespace.MySubNamespace'. [-namespace]

## Exclude

**XUnit Action**

XUnit

Include

Exclude

Options

Environment

Comments

Exclude Traits

Do not run tests with any of the matching name/value traits. One per line. [-notrait]

Exclude Methods

MyNamespace.MyClass.MyTestMethod

Do not run any of the given test method. One per line. Each method can be fully specified or use a wildcard. i.e. 'MyNamespace.MyClass.MyTestMethod' or '\*.MyTestMethod'. [-nomethod]

Exclude Classes

MyNamespace.MyClass

Do not run any of the given test classes. One class per line. Each class should be fully specified i.e. 'MyNamespace.MyClass'. [-noclass]

Exclude Namespaces

MyNamespace.MySubNamespace

Do not run any of the given test namespaces. One per line. 'MyNamespace.MySubNamespace'. [-nonamespace]

Validate

Save

Cancel

Help

### Exclude Traits

Do not run tests with any of the matching name/value traits. One per line. [-notraits]

### Exclude Methods

Do not run any of the given test method. One per line. Each method can be fully specified or use a wildcard. i.e. 'MyNamespace.MyClass.MyTestMethod' or '\*.MyTestMethod'. [-nomethod]

### Exclude Classes

Do not run any of the given test classes. One class per line. Each class should be fully specified i.e. 'MyNamespace.MyClass'. [-noclass]

### Exclude Namespaces

Do not run any of the given test namespaces. One per line. 'MyNamespace.MySubNamespace'. [-nonamespace]

## Options

**XUnit Action**

XUnitIncludeExcludeOptionsEnvironmentComments

Required Field

☐ Convert skipped tests into failures. [-failskips]

☐ Stop on first test failure. [-stoponfail]

Level of Parallelism

Default

Logging Level

Normal

☐ Enable diagnostic messages for all test assemblies. [-diagnostics]

Additional Command

Line Arguments

☐ Fail action if any tests fail

☒ Fail action if any tests error

Timeout (in seconds)

0

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

Validate

SaveCancelHelp

### Convert skipped tests into failures

If this is ticked, skipped tests are converted into failures. [-failskips]

### Stop on first test failure

If this is ticked, it stops on first test failure. [-stoponfail]

### Level of Parallelism

The level of parallelism. [-parallel]

### Logging level

The amount of information detail to display in the build log. [-quiet | -verbose]

### Enable diagnostic messages for all test assemblies.

Not visible if the 'Logging Level' field is set to Quiet'.

If this is ticked, it enables diagnostic messages for all test assemblies. [-diagnostics]

### Fail action if any tests fail

Tick this to cause the build to fail if any tests fail.

### Fail action if any tests error

Tick this to cause the build to fail if an error occurred while running any test.

### 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

**XUnit Action**

XUnit

Include

Exclude

Options

Environment

Comments

☐

Convert skipped tests into failures. [-failskips]

☐

Stop on first test failure. [-stoponfail]

Level of Parallelism

Default

▼

Logging Level

Normal

▼

☐

Enable diagnostic messages for all test assemblies. [-diagnostics]

Additional Command Line Arguments

☐

Fail action if any tests fail

☒

Fail action if any tests error

Timeout (in seconds)

0

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

Validate

Save

Cancel

Help

### 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.