4.10.1

* Docs  
  [User Manual](http://docs.google.com/userguide/userguide.html)  
  [Guides and Tutorials](https://guides.gradle.org)  
  [DSL Reference](http://docs.google.com/dsl/)  
  [Javadoc](http://docs.google.com/javadoc/)  
  [Release Notes](http://docs.google.com/release-notes.html)
* [Forums](https://discuss.gradle.org/)
* [Training](https://gradle.org/training/)
* [Try Gradle Enterprise](https://gradle.com/enterprise)
* [User Manual Home](http://docs.google.com/userguide/userguide.html)
* [DSL Reference Home](http://docs.google.com/dsl/)
* [Release Notes](http://docs.google.com/release-notes.html)
  + [Properties](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N12D36)
  + [Methods](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N12D9C)
  + [Script blocks](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N12E00)

### Build script blocks

* [allprojects { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:allprojects(groovy.lang.Closure))
* [artifacts { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:artifacts(groovy.lang.Closure))
* [buildscript { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:buildscript(groovy.lang.Closure))
* [configurations { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:configurations(groovy.lang.Closure))
* [dependencies { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:dependencies(groovy.lang.Closure))
* [repositories { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:repositories(groovy.lang.Closure))
* [sourceSets { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:sourceSets(groovy.lang.Closure))
* [subprojects { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:subprojects(groovy.lang.Closure))
* [publishing { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:publishing(groovy.lang.Closure))

### Core types

* [Project](http://docs.google.com/org.gradle.api.Project.html)
* [Task](http://docs.google.com/org.gradle.api.Task.html)
* [Gradle](http://docs.google.com/org.gradle.api.invocation.Gradle.html)
* [Settings](http://docs.google.com/org.gradle.api.initialization.Settings.html)
* [IncludedBuild](http://docs.google.com/org.gradle.api.initialization.IncludedBuild.html)
* [Script](http://docs.google.com/org.gradle.api.Script.html)
* [JavaToolChain](http://docs.google.com/org.gradle.jvm.toolchain.JavaToolChain.html)
* [SourceSet](http://docs.google.com/org.gradle.api.tasks.SourceSet.html)
* [SourceSetOutput](http://docs.google.com/org.gradle.api.tasks.SourceSetOutput.html)
* [SourceDirectorySet](http://docs.google.com/org.gradle.api.file.SourceDirectorySet.html)
* [IncrementalTaskInputs](http://docs.google.com/org.gradle.api.tasks.incremental.IncrementalTaskInputs.html)
* [Configuration](http://docs.google.com/org.gradle.api.artifacts.Configuration.html)
* [ResolutionStrategy](http://docs.google.com/org.gradle.api.artifacts.ResolutionStrategy.html)
* [ArtifactResolutionQuery](http://docs.google.com/org.gradle.api.artifacts.query.ArtifactResolutionQuery.html)
* [ComponentSelection](http://docs.google.com/org.gradle.api.artifacts.ComponentSelection.html)
* [ComponentSelectionRules](http://docs.google.com/org.gradle.api.artifacts.ComponentSelectionRules.html)
* [ExtensionAware](http://docs.google.com/org.gradle.api.plugins.ExtensionAware.html)
* [ExtraPropertiesExtension](http://docs.google.com/org.gradle.api.plugins.ExtraPropertiesExtension.html)
* [PluginDependenciesSpec](http://docs.google.com/org.gradle.plugin.use.PluginDependenciesSpec.html)
* [PluginDependencySpec](http://docs.google.com/org.gradle.plugin.use.PluginDependencySpec.html)
* [PluginManagementSpec](http://docs.google.com/org.gradle.plugin.management.PluginManagementSpec.html)
* [ResourceHandler](http://docs.google.com/org.gradle.api.resources.ResourceHandler.html)
* [TextResourceFactory](http://docs.google.com/org.gradle.api.resources.TextResourceFactory.html)

### Publishing types

* [PublishingExtension](http://docs.google.com/org.gradle.api.publish.PublishingExtension.html)
* [IvyPublication](http://docs.google.com/org.gradle.api.publish.ivy.IvyPublication.html)
* [IvyArtifact](http://docs.google.com/org.gradle.api.publish.ivy.IvyArtifact.html)
* [IvyArtifactSet](http://docs.google.com/org.gradle.api.publish.ivy.IvyArtifactSet.html)
* [IvyModuleDescriptorSpec](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorSpec.html)
* [IvyModuleDescriptorAuthor](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorAuthor.html)
* [IvyModuleDescriptorLicense](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorLicense.html)
* [IvyModuleDescriptorDescription](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorDescription.html)
* [MavenPublication](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html)
* [MavenArtifact](http://docs.google.com/org.gradle.api.publish.maven.MavenArtifact.html)
* [MavenArtifactSet](http://docs.google.com/org.gradle.api.publish.maven.MavenArtifactSet.html)
* [MavenPom](http://docs.google.com/org.gradle.api.publish.maven.MavenPom.html)
* [MavenPomCiManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomCiManagement.html)
* [MavenPomContributor](http://docs.google.com/org.gradle.api.publish.maven.MavenPomContributor.html)
* [MavenPomContributorSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomContributorSpec.html)
* [MavenPomDeveloper](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDeveloper.html)
* [MavenPomDeveloperSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDeveloperSpec.html)
* [MavenPomDistributionManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDistributionManagement.html)
* [MavenPomIssueManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomIssueManagement.html)
* [MavenPomLicense](http://docs.google.com/org.gradle.api.publish.maven.MavenPomLicense.html)
* [MavenPomLicenseSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomLicenseSpec.html)
* [MavenPomMailingList](http://docs.google.com/org.gradle.api.publish.maven.MavenPomMailingList.html)
* [MavenPomMailingListSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomMailingListSpec.html)
* [MavenPomOrganization](http://docs.google.com/org.gradle.api.publish.maven.MavenPomOrganization.html)
* [MavenPomRelocation](http://docs.google.com/org.gradle.api.publish.maven.MavenPomRelocation.html)
* [MavenPomScm](http://docs.google.com/org.gradle.api.publish.maven.MavenPomScm.html)

### Container types

* [TaskContainer](http://docs.google.com/org.gradle.api.tasks.TaskContainer.html)
* [ConfigurationContainer](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html)
* [RepositoryHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.RepositoryHandler.html)
* [DependencyHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html)
* [ArtifactHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.ArtifactHandler.html)

### Build Cache types

* [BuildCacheConfiguration](http://docs.google.com/org.gradle.caching.configuration.BuildCacheConfiguration.html)
* [DirectoryBuildCache](http://docs.google.com/org.gradle.caching.local.DirectoryBuildCache.html)
* [HttpBuildCache](http://docs.google.com/org.gradle.caching.http.HttpBuildCache.html)

### Input Normalization types

* [InputNormalizationHandler](http://docs.google.com/org.gradle.normalization.InputNormalizationHandler.html)
* [InputNormalization](http://docs.google.com/org.gradle.normalization.InputNormalization.html)
* [RuntimeClasspathNormalization](http://docs.google.com/org.gradle.normalization.RuntimeClasspathNormalization.html)

### Help Task types

* [TaskReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.TaskReportTask.html)
* [ProjectReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.ProjectReportTask.html)
* [DependencyReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.DependencyReportTask.html)
* [DependencyInsightReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.DependencyInsightReportTask.html)
* [PropertyReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.PropertyReportTask.html)
* [ComponentReport](http://docs.google.com/org.gradle.api.reporting.components.ComponentReport.html)
* [DependentComponentsReport](http://docs.google.com/org.gradle.api.reporting.dependents.DependentComponentsReport.html)
* [ModelReport](http://docs.google.com/org.gradle.api.reporting.model.ModelReport.html)

### Task types

* [AntlrTask](http://docs.google.com/org.gradle.api.plugins.antlr.AntlrTask.html)
* [BuildEnvironmentReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.BuildEnvironmentReportTask.html)
* [Checkstyle](http://docs.google.com/org.gradle.api.plugins.quality.Checkstyle.html)
* [CodeNarc](http://docs.google.com/org.gradle.api.plugins.quality.CodeNarc.html)
* [CompareGradleBuilds](http://docs.google.com/org.gradle.api.plugins.buildcomparison.gradle.CompareGradleBuilds.html)
* [Copy](http://docs.google.com/org.gradle.api.tasks.Copy.html)
* [CreateStartScripts](http://docs.google.com/org.gradle.jvm.application.tasks.CreateStartScripts.html)
* [Delete](http://docs.google.com/org.gradle.api.tasks.Delete.html)
* [Ear](http://docs.google.com/org.gradle.plugins.ear.Ear.html)
* [Exec](http://docs.google.com/org.gradle.api.tasks.Exec.html)
* [FindBugs](http://docs.google.com/org.gradle.api.plugins.quality.FindBugs.html)
* [GenerateIvyDescriptor](http://docs.google.com/org.gradle.api.publish.ivy.tasks.GenerateIvyDescriptor.html)
* [GenerateMavenPom](http://docs.google.com/org.gradle.api.publish.maven.tasks.GenerateMavenPom.html)
* [GenerateBuildDashboard](http://docs.google.com/org.gradle.api.reporting.GenerateBuildDashboard.html)
* [GradleBuild](http://docs.google.com/org.gradle.api.tasks.GradleBuild.html)
* [GroovyCompile](http://docs.google.com/org.gradle.api.tasks.compile.GroovyCompile.html)
* [Groovydoc](http://docs.google.com/org.gradle.api.tasks.javadoc.Groovydoc.html)
* [HtmlDependencyReportTask](http://docs.google.com/org.gradle.api.reporting.dependencies.HtmlDependencyReportTask.html)
* [JacocoReport](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoReport.html)
* [JacocoMerge](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoMerge.html)
* [JacocoCoverageVerification](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoCoverageVerification.html)
* [Jar](http://docs.google.com/org.gradle.api.tasks.bundling.Jar.html)
* [JavaCompile](http://docs.google.com/org.gradle.api.tasks.compile.JavaCompile.html)
* [Javadoc](http://docs.google.com/org.gradle.api.tasks.javadoc.Javadoc.html)
* [JavaExec](http://docs.google.com/org.gradle.api.tasks.JavaExec.html)
* [JDepend](http://docs.google.com/org.gradle.api.plugins.quality.JDepend.html)
* [Pmd](http://docs.google.com/org.gradle.api.plugins.quality.Pmd.html)
* [PublishToIvyRepository](http://docs.google.com/org.gradle.api.publish.ivy.tasks.PublishToIvyRepository.html)
* [PublishToMavenRepository](http://docs.google.com/org.gradle.api.publish.maven.tasks.PublishToMavenRepository.html)
* [ScalaCompile](http://docs.google.com/org.gradle.api.tasks.scala.ScalaCompile.html)
* [ScalaDoc](http://docs.google.com/org.gradle.api.tasks.scala.ScalaDoc.html)
* [InitBuild](http://docs.google.com/org.gradle.buildinit.tasks.InitBuild.html)
* [Sign](http://docs.google.com/org.gradle.plugins.signing.Sign.html)
* [Sync](http://docs.google.com/org.gradle.api.tasks.Sync.html)
* [Tar](http://docs.google.com/org.gradle.api.tasks.bundling.Tar.html)
* [AbstractTestTask](http://docs.google.com/org.gradle.api.tasks.testing.AbstractTestTask.html)
* [Test](http://docs.google.com/org.gradle.api.tasks.testing.Test.html)
* [TestReport](http://docs.google.com/org.gradle.api.tasks.testing.TestReport.html)
* [Upload](http://docs.google.com/org.gradle.api.tasks.Upload.html)
* [War](http://docs.google.com/org.gradle.api.tasks.bundling.War.html)
* [Wrapper](http://docs.google.com/org.gradle.api.tasks.wrapper.Wrapper.html)
* [WriteProperties](http://docs.google.com/org.gradle.api.tasks.WriteProperties.html)
* [Zip](http://docs.google.com/org.gradle.api.tasks.bundling.Zip.html)

### Reporting types

* [CustomizableHtmlReport](http://docs.google.com/org.gradle.api.reporting.CustomizableHtmlReport.html)
* [SingleFileReport](http://docs.google.com/org.gradle.api.reporting.SingleFileReport.html)
* [DirectoryReport](http://docs.google.com/org.gradle.api.reporting.DirectoryReport.html)
* [FindBugsXmlReport](http://docs.google.com/org.gradle.api.plugins.quality.FindBugsXmlReport.html)
* [Report](http://docs.google.com/org.gradle.api.reporting.Report.html)
* [Reporting](http://docs.google.com/org.gradle.api.reporting.Reporting.html)
* [ReportContainer](http://docs.google.com/org.gradle.api.reporting.ReportContainer.html)
* [ReportingExtension](http://docs.google.com/org.gradle.api.reporting.ReportingExtension.html)

### Eclipse/IDEA model types

* [EclipseModel](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseModel.html)
* [EclipseProject](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)
* [EclipseClasspath](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseClasspath.html)
* [EclipseJdt](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseJdt.html)
* [EclipseWtp](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtp.html)
* [EclipseWtpComponent](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtpComponent.html)
* [EclipseWtpFacet](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtpFacet.html)
* [IdeaModel](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaModel.html)
* [IdeaProject](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaProject.html)
* [IdeaModule](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaModule.html)
* [IdeaWorkspace](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaWorkspace.html)
* [XmlFileContentMerger](http://docs.google.com/org.gradle.plugins.ide.api.XmlFileContentMerger.html)
* [FileContentMerger](http://docs.google.com/org.gradle.plugins.ide.api.FileContentMerger.html)

### Eclipse/IDEA task types

* [GenerateEclipseProject](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseProject.html)
* [GenerateEclipseClasspath](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseClasspath.html)
* [GenerateEclipseJdt](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseJdt.html)
* [GenerateEclipseWtpComponent](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseWtpComponent.html)
* [GenerateEclipseWtpFacet](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseWtpFacet.html)
* [GenerateIdeaModule](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaModule.html)
* [GenerateIdeaProject](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaProject.html)
* [GenerateIdeaWorkspace](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaWorkspace.html)

### Native software types

* [PrebuiltLibrary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltLibrary.html)
* [PrebuiltSharedLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltSharedLibraryBinary.html)
* [PrebuiltStaticLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltStaticLibraryBinary.html)
* [NativeComponentSpec](http://docs.google.com/org.gradle.nativeplatform.NativeComponentSpec.html)
* [NativeExecutableSpec](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableSpec.html)
* [NativeLibrarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeLibrarySpec.html)
* [NativeTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.NativeTestSuiteSpec.html)
* [CUnitTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.cunit.CUnitTestSuiteSpec.html)
* [GoogleTestTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.googletest.GoogleTestTestSuiteSpec.html)
* [NativeBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeBinarySpec.html)
* [NativeExecutableBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableBinarySpec.html)
* [NativeLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeLibraryBinarySpec.html)
* [SharedLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.SharedLibraryBinarySpec.html)
* [StaticLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.StaticLibraryBinarySpec.html)
* [NativeTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.NativeTestSuiteBinarySpec.html)
* [CUnitTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.cunit.CUnitTestSuiteBinarySpec.html)
* [GoogleTestTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.googletest.GoogleTestTestSuiteBinarySpec.html)
* [NativePlatform](http://docs.google.com/org.gradle.nativeplatform.platform.NativePlatform.html)
* [BuildType](http://docs.google.com/org.gradle.nativeplatform.BuildType.html)
* [Flavor](http://docs.google.com/org.gradle.nativeplatform.Flavor.html)
* [Gcc](http://docs.google.com/org.gradle.nativeplatform.toolchain.Gcc.html)
* [Clang](http://docs.google.com/org.gradle.nativeplatform.toolchain.Clang.html)
* [VisualCpp](http://docs.google.com/org.gradle.nativeplatform.toolchain.VisualCpp.html)
* [AssemblerSourceSet](http://docs.google.com/org.gradle.language.assembler.AssemblerSourceSet.html)
* [CSourceSet](http://docs.google.com/org.gradle.language.c.CSourceSet.html)
* [CppSourceSet](http://docs.google.com/org.gradle.language.cpp.CppSourceSet.html)
* [ObjectiveCSourceSet](http://docs.google.com/org.gradle.language.objectivec.ObjectiveCSourceSet.html)
* [ObjectiveCppSourceSet](http://docs.google.com/org.gradle.language.objectivecpp.ObjectiveCppSourceSet.html)
* [WindowsResourceSet](http://docs.google.com/org.gradle.language.rc.WindowsResourceSet.html)
* [VisualStudioProject](http://docs.google.com/org.gradle.ide.visualstudio.VisualStudioProject.html)
* [VisualStudioSolution](http://docs.google.com/org.gradle.ide.visualstudio.VisualStudioSolution.html)
* [NativeExecutable](http://docs.google.com/org.gradle.nativeplatform.NativeExecutable.html)
* [NativeLibrary](http://docs.google.com/org.gradle.nativeplatform.NativeLibrary.html)
* [NativeBinary](http://docs.google.com/org.gradle.nativeplatform.NativeBinary.html)
* [NativeExecutableBinary](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableBinary.html)
* [SharedLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.SharedLibraryBinary.html)
* [StaticLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.StaticLibraryBinary.html)

### Native component task types

* [CppCompile](http://docs.google.com/org.gradle.language.cpp.tasks.CppCompile.html)
* [CCompile](http://docs.google.com/org.gradle.language.c.tasks.CCompile.html)
* [Assemble](http://docs.google.com/org.gradle.language.assembler.tasks.Assemble.html)
* [ObjectiveCCompile](http://docs.google.com/org.gradle.language.objectivec.tasks.ObjectiveCCompile.html)
* [ObjectiveCppCompile](http://docs.google.com/org.gradle.language.objectivecpp.tasks.ObjectiveCppCompile.html)
* [WindowsResourceCompile](http://docs.google.com/org.gradle.language.rc.tasks.WindowsResourceCompile.html)
* [LinkExecutable](http://docs.google.com/org.gradle.nativeplatform.tasks.LinkExecutable.html)
* [LinkSharedLibrary](http://docs.google.com/org.gradle.nativeplatform.tasks.LinkSharedLibrary.html)
* [CreateStaticLibrary](http://docs.google.com/org.gradle.nativeplatform.tasks.CreateStaticLibrary.html)
* [InstallExecutable](http://docs.google.com/org.gradle.nativeplatform.tasks.InstallExecutable.html)
* [RunTestExecutable](http://docs.google.com/org.gradle.nativeplatform.test.tasks.RunTestExecutable.html)

EclipseProject

**Table of Contents**

[Properties](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N12D36)[Methods](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N12D9C)[Script blocks](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N12E00)[Property details](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N12E15)[Method details](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N12F46)[Script block details](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#N1303C)

| API Documentation: | [EclipseProject](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/EclipseProject.html) |
| --- | --- |

Enables fine-tuning project details (.project file) of the Eclipse plugin

Example of use with a blend of all possible properties. Bear in mind that usually you don't have configure eclipse project directly because Gradle configures it for free!

apply plugin: 'java'  
apply plugin: 'eclipse'  
  
eclipse {  
 project {  
 //if you don't like the name Gradle has chosen  
 name = 'someBetterName'  
  
 //if you want to specify the Eclipse project's comment  
 comment = 'Very interesting top secret project'  
  
 //if you want to append some extra referenced projects in a declarative fashion:  
 referencedProjects 'someProject', 'someOtherProject'  
 //if you want to assign referenced projects  
 referencedProjects = ['someProject'] as Set  
  
 //if you want to append some extra natures in a declarative fashion:  
 natures 'some.extra.eclipse.nature', 'some.another.interesting.nature'  
 //if you want to assign natures in a groovy fashion:  
 natures = ['some.extra.eclipse.nature', 'some.another.interesting.nature']  
  
 //if you want to append some extra build command:  
 buildCommand 'buildThisLovelyProject'  
 //if you want to append a build command with parameters:  
 buildCommand 'buildItWithTheArguments', argumentOne: "I'm first", argumentTwo: "I'm second"  
  
 //if you want to create an extra link in the eclipse project,  
 //by location uri:  
 linkedResource name: 'someLinkByLocationUri', type: 'someLinkType', locationUri: 'file://someUri'  
 //by location:  
 linkedResource name: 'someLinkByLocation', type: 'someLinkType', location: '/some/location'  
  
 //if you don't want any node\_modules folder to appear in Eclipse, you can filter it out:  
 resourceFilter {  
 appliesTo = 'FOLDERS'  
 type = 'EXCLUDE\_ALL'  
 matcher {  
 id = 'org.eclipse.ui.ide.multiFilter'  
 arguments = '1.0-name-matches-false-false-node\_modules'  
 }  
 }  
 }  
}

For tackling edge cases users can perform advanced configuration on resulting XML file. It is also possible to affect the way eclipse plugin merges the existing configuration via beforeMerged and whenMerged closures.

beforeMerged and whenMerged closures receive [Project](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/Project.html) object

Examples of advanced configuration:

apply plugin: 'java'  
apply plugin: 'eclipse'  
  
eclipse {  
 project {  
  
 file {  
 //if you want to mess with the resulting XML in whatever way you fancy  
 withXml {  
 def node = it.asNode()  
 node.appendNode('xml', 'is what I love')  
 }  
  
 //closure executed after .project content is loaded from existing file  
 //but before gradle build information is merged  
 beforeMerged { project ->  
 //if you want skip merging natures... (a very abstract example)  
 project.natures.clear()  
 }  
  
 //closure executed after .project content is loaded from existing file  
 //and after gradle build information is merged  
 whenMerged { project ->  
 //you can tinker with the Project here  
 }  
 }  
 }  
}

Properties

| Property | Description |
| --- | --- |
| [buildCommands](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:buildCommands) | The build commands to be added to this Eclipse project. |
| [comment](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:comment) | A comment used for the eclipse project. By default it will be configured to *project.description* |
| [file](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:file) | See [EclipseProject.file(org.gradle.api.Action)](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:file(org.gradle.api.Action)) |
| [linkedResources](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:linkedResources) | The linked resources to be added to this Eclipse project. |
| [name](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:name) | Configures eclipse project name. It is *optional* because the task should configure it correctly for you. By default it will try to use the *project.name* or prefix it with a part of a *project.path* to make sure the moduleName is unique in the scope of a multi-module build. The 'uniqueness' of a module name is required for correct import into Eclipse and the task will make sure the name is unique. |
| [natures](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:natures) | The natures to be added to this Eclipse project. |
| [referencedProjects](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:referencedProjects) | The referenced projects of this Eclipse project (\*not\*: java build path project references). |
| [resourceFilters](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:resourceFilters) | Incubating  The resource filters of the eclipse project. |

Methods

| Method | Description |
| --- | --- |
| [buildCommand](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:buildCommand(java.lang.String))(buildCommand) | Adds a build command to the eclipse project. |
| [buildCommand](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:buildCommand(java.util.Map,%20java.lang.String))(args, buildCommand) | Adds a build command with arguments to the eclipse project. |
| [file](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:file(org.gradle.api.Action))(action) | Enables advanced configuration like tinkering with the output XML or affecting the way existing .project content is merged with gradle build information. For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html) |
| [linkedResource](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:linkedResource(java.util.Map))(args) | Adds a resource link (aka 'source link') to the eclipse project. |
| [natures](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:natures(java.lang.String%5B%5D))(natures) | Appends natures entries to the eclipse project. |
| [referencedProjects](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:referencedProjects(java.lang.String%5B%5D))(referencedProjects) | The referenced projects of this Eclipse project (\*not\*: java build path project references). |
| [resourceFilter](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:resourceFilter(groovy.lang.Closure))(configureClosure) | Incubating  Adds a resource filter to the eclipse project. |
| [resourceFilter](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:resourceFilter(org.gradle.api.Action))(configureAction) | Incubating  Adds a resource filter to the eclipse project. |

Script blocks

| Block | Description |
| --- | --- |
| [file](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:file(groovy.lang.Closure)) | Enables advanced configuration like tinkering with the output XML or affecting the way existing .project content is merged with gradle build information |

Property details

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**BuildCommand**](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/BuildCommand.html)> buildCommands

The build commands to be added to this Eclipse project.

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

Default with eclipse and java plugins:Java builder, plus Scala and Web builders as appropriate.

[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) comment

A comment used for the eclipse project. By default it will be configured to *project.description*

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

Default with eclipse and java plugins:project.description

[**XmlFileContentMerger**](http://docs.google.com/dsl/org.gradle.plugins.ide.api.XmlFileContentMerger.html) file (read-only)

See [EclipseProject.file(org.gradle.api.Action)](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:file(org.gradle.api.Action))

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**Link**](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/Link.html)> linkedResources

The linked resources to be added to this Eclipse project.

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

Default with eclipse and java plugins:[]

[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name

Configures eclipse project name. It is *optional* because the task should configure it correctly for you. By default it will try to use the *project.name* or prefix it with a part of a *project.path* to make sure the moduleName is unique in the scope of a multi-module build. The 'uniqueness' of a module name is required for correct import into Eclipse and the task will make sure the name is unique.

The logic that makes sure project names are unique is available *since* 1.0-milestone-2

If your project has problems with unique names it is recommended to always run gradle eclipse from the root, e.g. for all subprojects, including generation of .classpath. If you run the generation of the eclipse project only for a single subproject then you may have different results because the unique names are calculated based on eclipse projects that are involved in the specific build run.

If you update the project names then make sure you run gradle eclipse from the root, e.g. for all subprojects. The reason is that there may be subprojects that depend on the subproject with amended eclipse project name. So you want them to be generated as well because the project dependencies in .classpath need to refer to the amended project name. Basically, for non-trivial projects it is recommended to always run gradle eclipse from the root.

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

Default with eclipse and java plugins:*${project.name}* (sometimes prefixed with parts of *${project.path}* to guarantee uniqeness)

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> natures

The natures to be added to this Eclipse project.

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

Default with eclipse and java plugins:Java nature, plus Groovy, Scala and Web natures as appropriate.

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> referencedProjects

The referenced projects of this Eclipse project (\*not\*: java build path project references).

Referencing projects does not mean adding a build path dependencies between them! If you need to configure a build path dependency use Gradle's dependencies section or eclipse.classpath.whenMerged { classpath -> ... to manipulate the classpath entries

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

Default with eclipse and java plugins:[]

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**ResourceFilter**](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/ResourceFilter.html)> resourceFilters (read-only)

Note: This property is [incubating](http://docs.google.com/userguide/feature_lifecycle.html) and may change in a future version of Gradle.

The resource filters of the eclipse project.

Default with eclipse and java plugins:[]

Method details

void buildCommand([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) buildCommand)

Adds a build command to the eclipse project.

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

void buildCommand([**Map**](http://download.oracle.com/javase/8/docs/api/java/util/Map.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html), [**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> args, [**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) buildCommand)

Adds a build command with arguments to the eclipse project.

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

void file([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**XmlFileContentMerger**](http://docs.google.com/dsl/org.gradle.plugins.ide.api.XmlFileContentMerger.html)> action)

Enables advanced configuration like tinkering with the output XML or affecting the way existing .project content is merged with gradle build information. For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

void linkedResource([**Map**](http://download.oracle.com/javase/8/docs/api/java/util/Map.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html), [**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> args)

Adds a resource link (aka 'source link') to the eclipse project.

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

void natures([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)... natures)

Appends natures entries to the eclipse project.

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

void referencedProjects([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)... referencedProjects)

The referenced projects of this Eclipse project (\*not\*: java build path project references).

Referencing projects does not mean adding a build path dependencies between them! If you need to configure a build path dependency use Gradle's dependencies section or eclipse.classpath.whenMerged { classpath -> ... to manipulate the classpath entries

[**ResourceFilter**](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/ResourceFilter.html) resourceFilter([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) configureClosure)

Note: This method is [incubating](http://docs.google.com/userguide/feature_lifecycle.html) and may change in a future version of Gradle.

Adds a resource filter to the eclipse project.

For examples, see docs for [ResourceFilter](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/ResourceFilter.html)

[**ResourceFilter**](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/ResourceFilter.html) resourceFilter([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**ResourceFilter**](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/ResourceFilter.html)> configureAction)

Note: This method is [incubating](http://docs.google.com/userguide/feature_lifecycle.html) and may change in a future version of Gradle.

Adds a resource filter to the eclipse project.

For examples, see docs for [ResourceFilter](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/ResourceFilter.html)

Script block details

file { }

Enables advanced configuration like tinkering with the output XML or affecting the way existing .project content is merged with gradle build information

The object passed to whenMerged{} and beforeMerged{} closures is of type [Project](http://docs.google.com/javadoc/org/gradle/plugins/ide/eclipse/model/Project.html)

For example see docs for [EclipseProject](http://docs.google.com/dsl/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)

Delegates to:[XmlFileContentMerger](http://docs.google.com/dsl/org.gradle.plugins.ide.api.XmlFileContentMerger.html) from [file](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html#org.gradle.plugins.ide.eclipse.model.EclipseProject:file)

**Docs**

* [User Manual](http://docs.google.com/userguide/userguide.html)
* [DSL Reference](http://docs.google.com/dsl/)
* [Release Notes](http://docs.google.com/release-notes.html)
* [Javadoc](http://docs.google.com/javadoc/)

**News**

* [Blog](https://blog.gradle.org/)
* [Newsletter](https://newsletter.gradle.com/)
* [Twitter](https://twitter.com/gradle)

**Products**

* [Build Scans](https://gradle.com/build-scans)
* [Build Cache](https://gradle.com/build-cache)
* [Enterprise Docs](https://gradle.com/enterprise/resources)

**Get Help**

* [Forums](https://discuss.gradle.org/c/help-discuss)
* [GitHub](https://github.com/gradle/)
* [Training](https://gradle.org/training/)
* [Services](https://gradle.org/services/)

Subscribe for important Gradle updates and news

Subscribe

By entering your email, you agree to our [Terms](https://gradle.org/terms/) and [Privacy Policy](https://gradle.org/privacy/), including receipt of emails. You can unsubscribe at any time.

© [Gradle Inc.](https://gradle.com)2018 All rights reserved.

[Careers](https://gradle.com/careers) | [Privacy](https://gradle.org/privacy) | [Terms of Service](https://gradle.org/terms) | [Contact](https://gradle.org/contact/)