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.api.artifacts.dsl.DependencyHandler.html#N16FE3)
  + [Methods](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#N17005)

### 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)

DependencyHandler

**Table of Contents**

[Advanced dependency configuration](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#N16F0A)[Dependency notations](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#N16F32)[Properties](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#N16FE3)[Methods](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#N17005)[Script blocks](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#N1709F)[Property details](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#N170A4)[Method details](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#N170D1)

| API Documentation: | [DependencyHandler](http://docs.google.com/javadoc/org/gradle/api/artifacts/dsl/DependencyHandler.html) |
| --- | --- |

A DependencyHandler is used to declare dependencies. Dependencies are grouped into configurations (see [Configuration](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html)).

To declare a specific dependency for a configuration you can use the following syntax:

dependencies {  
 configurationName dependencyNotation1, dependencyNotation2, ...  
}

Example shows a basic way of declaring dependencies.

apply plugin: 'java'  
//so that we can use 'compile', 'testCompile' for dependencies  
  
dependencies {  
 //for dependencies found in artifact repositories you can use  
 //the group:name:version notation  
 compile 'commons-lang:commons-lang:2.6'  
 testCompile 'org.mockito:mockito:1.9.0-rc1'  
  
 //map-style notation:  
 compile group: 'com.google.code.guice', name: 'guice', version: '1.0'  
  
 //declaring arbitrary files as dependencies  
 compile files('hibernate.jar', 'libs/spring.jar')  
  
 //putting all jars from 'libs' onto compile classpath  
 compile fileTree('libs')  
}

Advanced dependency configuration

To do some advanced configuration on a dependency when it is declared, you can additionally pass a configuration closure:

dependencies {  
 configurationName(dependencyNotation){  
 configStatement1  
 configStatement2  
 }  
}

Examples of advanced dependency declaration including:

* Forcing certain dependency version in case of the conflict.
* Excluding certain dependencies by name, group or both. More details about per-dependency exclusions can be found in docs for [ModuleDependency.exclude(java.util.Map)](http://docs.google.com/javadoc/org/gradle/api/artifacts/ModuleDependency.html#exclude-java.util.Map-).
* Avoiding transitive dependencies for certain dependency.

apply plugin: 'java' //so that I can declare 'compile' dependencies  
  
dependencies {  
 compile('org.hibernate:hibernate:3.1') {  
 //in case of versions conflict '3.1' version of hibernate wins:  
 force = true  
  
 //excluding a particular transitive dependency:  
 exclude module: 'cglib' //by artifact name  
 exclude group: 'org.jmock' //by group  
 exclude group: 'org.unwanted', module: 'iAmBuggy' //by both name and group  
  
 //disabling all transitive dependencies of this dependency  
 transitive = false  
 }  
}

More examples of advanced configuration, useful when dependency module has multiple artifacts:

* Declaring dependency to a specific configuration of the module.
* Explicit specification of the artifact. See also [ModuleDependency.artifact(groovy.lang.Closure)](http://docs.google.com/javadoc/org/gradle/api/artifacts/ModuleDependency.html#artifact-groovy.lang.Closure-).

apply plugin: 'java' //so that I can declare 'compile' dependencies  
  
dependencies {  
 //configuring dependency to specific configuration of the module  
 compile configuration: 'someConf', group: 'org.someOrg', name: 'someModule', version: '1.0'  
  
 //configuring dependency on 'someLib' module  
 compile(group: 'org.myorg', name: 'someLib', version:'1.0') {  
 //explicitly adding the dependency artifact:  
 artifact {  
 //useful when some artifact properties unconventional  
 name = 'someArtifact' //artifact name different than module name  
 extension = 'someExt'  
 type = 'someType'  
 classifier = 'someClassifier'  
 }  
 }  
}

Dependency notations

There are several supported dependency notations. These are described below. For each dependency declared this way, a [Dependency](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) object is created. You can use this object to query or further configure the dependency.

You can also always add instances of [Dependency](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) directly:

*configurationName* <instance>

External dependencies

There are two notations supported for declaring a dependency on an external module. One is a string notation formatted this way:

*configurationName* "*group*:*name*:*version*:*classifier*@*extension*"

The other is a map notation:

*configurationName* group: *group*, name: *name*, version: *version*, classifier: *classifier*, ext: *extension*

In both notations, all properties, except name, are optional.

External dependencies are represented by a [ExternalModuleDependency](http://docs.google.com/javadoc/org/gradle/api/artifacts/ExternalModuleDependency.html).

apply plugin: 'java'  
//so that we can use 'compile', 'testCompile' for dependencies  
  
dependencies {  
 //for dependencies found in artifact repositories you can use  
 //the string notation, e.g. group:name:version  
 compile 'commons-lang:commons-lang:2.6'  
 testCompile 'org.mockito:mockito:1.9.0-rc1'  
  
 //map notation:  
 compile group: 'com.google.code.guice', name: 'guice', version: '1.0'  
}

Project dependencies

To add a project dependency, you use the following notation:

*configurationName* project(':someProject')

The notation project(':projectA') is similar to the syntax you use when configuring a projectA in a multi-module gradle project.

By default, when you declare dependency to projectA, you actually declare dependency to the 'default' configuration of the projectA. If you need to depend on a specific configuration of projectA, use map notation for projects:

*configurationName* project(path: ':projectA', configuration: 'someOtherConfiguration')

Project dependencies are represented using a [ProjectDependency](http://docs.google.com/javadoc/org/gradle/api/artifacts/ProjectDependency.html).

File dependencies

You can also add a dependency using a [FileCollection](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html):

*configurationName* files('a file')

apply plugin: 'java'  
//so that we can use 'compile', 'testCompile' for dependencies  
  
dependencies {  
 //declaring arbitrary files as dependencies  
 compile files('hibernate.jar', 'libs/spring.jar')  
  
 //putting all jars from 'libs' onto compile classpath  
 compile fileTree('libs')  
}

File dependencies are represented using a [SelfResolvingDependency](http://docs.google.com/javadoc/org/gradle/api/artifacts/SelfResolvingDependency.html).

Dependencies to other configurations

You can add a dependency using a [Configuration](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html).

When the configuration is from the same project as the target configuration, the target configuration is changed to extend from the provided configuration.

When the configuration is from a different project, a project dependency is added.

Gradle distribution specific dependencies

It is possible to depend on certain Gradle APIs or libraries that Gradle ships with. It is particularly useful for Gradle plugin development. Example:

//Our Gradle plugin is written in groovy  
apply plugin: 'groovy'  
//now we can use the 'compile' configuration for declaring dependencies  
  
dependencies {  
 //we will use the Groovy version that ships with Gradle:  
 compile localGroovy()  
  
 //our plugin requires Gradle API interfaces and classes to compile:  
 compile gradleApi()  
  
 //we will use the Gradle test-kit to test build logic:  
 testCompile gradleTestKit()  
}

Client module dependencies

To add a client module to a configuration you can use the notation:

configurationName module(moduleNotation) {  
 module dependencies  
}

The module notation is the same as the dependency notations described above, except that the classifier property is not available. Client modules are represented using a [ClientModule](http://docs.google.com/javadoc/org/gradle/api/artifacts/ClientModule.html).

Properties

| Property | Description |
| --- | --- |
| [components](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:components) | Incubating  The component metadata handler for this project. The returned handler can be used for adding rules that modify the metadata of depended-on software components. |
| [modules](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:modules) | Incubating  The component module metadata handler for this project. The returned handler can be used for adding rules that modify the metadata of depended-on software components. |

Methods

| Method | Description |
| --- | --- |
| [add](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:add(java.lang.String,%20java.lang.Object))(configurationName, dependencyNotation) | Adds a dependency to the given configuration. |
| [add](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:add(java.lang.String,%20java.lang.Object,%20groovy.lang.Closure))(configurationName, dependencyNotation, configureClosure) | Adds a dependency to the given configuration, and configures the dependency using the given closure. |
| [components](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:components(org.gradle.api.Action))(configureAction) | Incubating  Configures component metadata for this project. |
| [create](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:create(java.lang.Object))(dependencyNotation) | Creates a dependency without adding it to a configuration. |
| [create](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:create(java.lang.Object,%20groovy.lang.Closure))(dependencyNotation, configureClosure) | Creates a dependency without adding it to a configuration, and configures the dependency using the given closure. |
| [createArtifactResolutionQuery](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:createArtifactResolutionQuery())() | Incubating  Creates an artifact resolution query. |
| [gradleApi](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:gradleApi())() | Creates a dependency on the API of the current version of Gradle. |
| [gradleTestKit](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:gradleTestKit())() | Incubating  Creates a dependency on the [Gradle test-kit](https://docs.gradle.org/current/userguide/test_kit.html) API. |
| [localGroovy](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:localGroovy())() | Creates a dependency on the Groovy that is distributed with the current version of Gradle. |
| [module](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:module(java.lang.Object))(notation) | Creates a dependency on a client module. |
| [module](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:module(java.lang.Object,%20groovy.lang.Closure))(notation, configureClosure) | Creates a dependency on a client module. The dependency is configured using the given closure before it is returned. |
| [modules](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:modules(org.gradle.api.Action))(configureAction) | Incubating  Configures module metadata for this project. |
| [project](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html#org.gradle.api.artifacts.dsl.DependencyHandler:project(java.util.Map))(notation) | Creates a dependency on a project. |

Script blocks

No script blocks

Property details

[**ComponentMetadataHandler**](http://docs.google.com/dsl/org.gradle.api.artifacts.dsl.ComponentMetadataHandler.html) components (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 component metadata handler for this project. The returned handler can be used for adding rules that modify the metadata of depended-on software components.

[**ComponentModuleMetadataHandler**](http://docs.google.com/dsl/org.gradle.api.artifacts.dsl.ComponentModuleMetadataHandler.html) modules (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 component module metadata handler for this project. The returned handler can be used for adding rules that modify the metadata of depended-on software components.

Method details

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) add([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) configurationName, [**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) dependencyNotation)

Adds a dependency to the given configuration.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) add([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) configurationName, [**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) dependencyNotation, [**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) configureClosure)

Adds a dependency to the given configuration, and configures the dependency using the given closure.

void components([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**ComponentMetadataHandler**](http://docs.google.com/dsl/org.gradle.api.artifacts.dsl.ComponentMetadataHandler.html)> configureAction)

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

Configures component metadata for this project.

This method executes the given action against the [ComponentMetadataHandler](http://docs.google.com/dsl/org.gradle.api.artifacts.dsl.ComponentMetadataHandler.html) for this project.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) create([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) dependencyNotation)

Creates a dependency without adding it to a configuration.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) create([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) dependencyNotation, [**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) configureClosure)

Creates a dependency without adding it to a configuration, and configures the dependency using the given closure.

[**ArtifactResolutionQuery**](http://docs.google.com/dsl/org.gradle.api.artifacts.query.ArtifactResolutionQuery.html) createArtifactResolutionQuery()

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

Creates an artifact resolution query.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) gradleApi()

Creates a dependency on the API of the current version of Gradle.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) gradleTestKit()

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

Creates a dependency on the [Gradle test-kit](https://docs.gradle.org/current/userguide/test_kit.html) API.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) localGroovy()

Creates a dependency on the Groovy that is distributed with the current version of Gradle.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) module([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) notation)

Creates a dependency on a client module.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) module([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) notation, [**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) configureClosure)

Creates a dependency on a client module. The dependency is configured using the given closure before it is returned.

void modules([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**ComponentModuleMetadataHandler**](http://docs.google.com/dsl/org.gradle.api.artifacts.dsl.ComponentModuleMetadataHandler.html)> configureAction)

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

Configures module metadata for this project.

This method executes the given action against the [ComponentModuleMetadataHandler](http://docs.google.com/dsl/org.gradle.api.artifacts.dsl.ComponentModuleMetadataHandler.html) for this project.

[**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html) project([**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), ?> notation)

Creates a dependency on a project.

**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/)