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.Configuration.html#N1A290)
  + [Methods](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#N1A341)

### 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)

Configuration

**Table of Contents**

[Properties](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#N1A290)[Methods](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#N1A341)[Script blocks](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#N1A43B)[Property details](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#N1A440)[Method details](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#N1A568)

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

A Configuration represents a group of artifacts and their dependencies. Find more information about declaring dependencies to a configuration or about managing configurations in docs for [ConfigurationContainer](http://docs.google.com/dsl/org.gradle.api.artifacts.ConfigurationContainer.html)

Configuration is an instance of a [FileCollection](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) that contains all dependencies (see also [Configuration.getAllDependencies()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:allDependencies)) but not artifacts. If you want to refer to the artifacts declared in this configuration please use [Configuration.getArtifacts()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:artifacts) or [Configuration.getAllArtifacts()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:allArtifacts). Read more about declaring artifacts in the configuration in docs for [ArtifactHandler](http://docs.google.com/dsl/org.gradle.api.artifacts.dsl.ArtifactHandler.html) Please see the [Managing Dependency Configurations User Guide chapter](https://docs.gradle.org/current/userguide/managing_dependency_configurations.html) for more information.

Properties

| Property | Description |
| --- | --- |
| [allArtifacts](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:allArtifacts) | The artifacts of this configuration including the artifacts of extended configurations. |
| [allDependencies](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:allDependencies) | Gets the complete set of declared dependencies including those contributed by superconfigurations. |
| [artifacts](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:artifacts) | The artifacts of this configuration excluding the artifacts of extended configurations. |
| [buildDependencies](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:buildDependencies) | Returns a TaskDependency object containing all required dependencies to build the local dependencies (e.g. project dependencies) belonging to this configuration or to one of its super configurations. |
| [dependencies](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:dependencies) | Gets the set of declared dependencies directly contained in this configuration (ignoring superconfigurations). |
| [description](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:description) | The description for this configuration. |
| [excludeRules](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:excludeRules) | The exclude rules applied for resolving any dependency of this configuration. |
| [extendsFrom](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:extendsFrom) | The names of the configurations which this configuration extends from. The artifacts of the super configurations are also available in this configuration. |
| [hierarchy](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:hierarchy) | Gets a ordered set including this configuration and all superconfigurations recursively. |
| [incoming](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:incoming) | The incoming dependencies of this configuration. |
| [name](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:name) | The name of this configuration. |
| [resolutionStrategy](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:resolutionStrategy) | The resolution strategy used by this configuration. The resolution strategy provides extra details on how to resolve this configuration. See docs for [ResolutionStrategy](http://docs.google.com/dsl/org.gradle.api.artifacts.ResolutionStrategy.html) for more info and examples. |
| [resolvedConfiguration](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:resolvedConfiguration) | Resolves this configuration. This locates and downloads the files which make up this configuration, and returns a [ResolvedConfiguration](http://docs.google.com/javadoc/org/gradle/api/artifacts/ResolvedConfiguration.html) that may be used to determine information about the resolve (including errors). |
| [state](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:state) | A Configuration represents a group of artifacts and their dependencies. Find more information about declaring dependencies to a configuration or about managing configurations in docs for [ConfigurationContainer](http://docs.google.com/dsl/org.gradle.api.artifacts.ConfigurationContainer.html) |
| [transitive](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:transitive) | The transitivity of this configuration. A transitive configuration contains the transitive closure of its direct dependencies, and all their dependencies. An intransitive configuration contains only the direct dependencies. The default value is true. |
| [visible](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:visible) | Returns true if this is a visible configuration. A visible configuration is usable outside the project it belongs to. The default value is true. |

Methods

| Method | Description |
| --- | --- |
| [copy](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copy())() | Creates a copy of this configuration that only contains the dependencies directly in this configuration (without contributions from superconfigurations). The new configuration will be in the UNRESOLVED state, but will retain all other attributes of this configuration except superconfigurations. [Configuration.getHierarchy()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:hierarchy) for the copy will not include any superconfigurations. |
| [copy](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copy(groovy.lang.Closure))(dependencySpec) | Takes a closure which gets coerced into a [Spec](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html). Behaves otherwise in the same way as [Configuration.copy(org.gradle.api.specs.Spec)](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copy(org.gradle.api.specs.Spec)) |
| [copy](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copy(org.gradle.api.specs.Spec))(dependencySpec) | Creates a copy of this configuration ignoring superconfigurations (see [Configuration.copy()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copy()) but filtering the dependencies using the specified dependency spec. |
| [copyRecursive](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copyRecursive())() | Creates a copy of this configuration that contains the dependencies directly in this configuration and those derived from superconfigurations. The new configuration will be in the UNRESOLVED state, but will retain all other attributes of this configuration except superconfigurations. [Configuration.getHierarchy()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:hierarchy) for the copy will not include any superconfigurations. |
| [copyRecursive](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copyRecursive(groovy.lang.Closure))(dependencySpec) | Takes a closure which gets coerced into a [Spec](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html). Behaves otherwise in the same way as [Configuration.copyRecursive(org.gradle.api.specs.Spec)](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copyRecursive(org.gradle.api.specs.Spec)) |
| [copyRecursive](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copyRecursive(org.gradle.api.specs.Spec))(dependencySpec) | Creates a copy of this configuration with dependencies from superconfigurations (see [Configuration.copyRecursive()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copyRecursive())) but filtering the dependencies using the dependencySpec. |
| [defaultDependencies](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:defaultDependencies(org.gradle.api.Action))(action) | Incubating  Execute the given action if the configuration has no defined dependencies when it first participates in dependency resolution. A Configuration will participate in dependency resolution when: |
| [exclude](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:exclude(java.util.Map))(excludeProperties) | Adds an exclude rule to exclude transitive dependencies for all dependencies of this configuration. You can also add exclude rules per-dependency. See [ModuleDependency.exclude(java.util.Map)](http://docs.google.com/javadoc/org/gradle/api/artifacts/ModuleDependency.html#exclude-java.util.Map-). |
| [extendsFrom](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:extendsFrom(org.gradle.api.artifacts.Configuration%5B%5D))(superConfigs) | Adds the given configurations to the set of configuration which this configuration extends from. |
| [fileCollection](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:fileCollection(groovy.lang.Closure))(dependencySpecClosure) | Takes a closure which gets coerced into a [Spec](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html). Behaves otherwise in the same way as [Configuration.fileCollection(org.gradle.api.specs.Spec)](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:fileCollection(org.gradle.api.specs.Spec)). |
| [fileCollection](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:fileCollection(org.gradle.api.artifacts.Dependency%5B%5D))(dependencies) | Resolves this configuration lazily. The resolve happens when the elements of the returned [FileCollection](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) get accessed the first time. This locates and downloads the files which make up this configuration. Only the resulting set of files belonging to specified dependencies is contained in the FileCollection. |
| [fileCollection](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:fileCollection(org.gradle.api.specs.Spec))(dependencySpec) | Resolves this configuration lazily. The resolve happens when the elements of the returned [FileCollection](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) get accessed the first time. This locates and downloads the files which make up this configuration. Only the resulting set of files belonging to the subset of dependencies specified by the dependencySpec is contained in the FileCollection. |
| [files](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:files(groovy.lang.Closure))(dependencySpecClosure) | Takes a closure which gets coerced into a [Spec](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html). Behaves otherwise in the same way as [Configuration.files(org.gradle.api.specs.Spec)](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:files(org.gradle.api.specs.Spec)). |
| [files](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:files(org.gradle.api.artifacts.Dependency%5B%5D))(dependencies) | Resolves this configuration. This locates and downloads the files which make up this configuration. But only the resulting set of files belonging to the specified dependencies is returned. |
| [files](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:files(org.gradle.api.specs.Spec))(dependencySpec) | Resolves this configuration. This locates and downloads the files which make up this configuration. But only the resulting set of files belonging to the subset of dependencies specified by the dependencySpec is returned. |
| [getTaskDependencyFromProjectDependency](http://docs.google.com/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:getTaskDependencyFromProjectDependency(boolean,%20java.lang.String))(useDependedOn, taskName) | Returns a TaskDependency object containing dependencies on all tasks with the specified name from project dependencies related to this configuration or one of its super configurations. These other projects may be projects this configuration depends on or projects with a similarly named configuration that depend on this one based on the useDependOn argument. |

Script blocks

No script blocks

Property details

[**PublishArtifactSet**](http://docs.google.com/javadoc/org/gradle/api/artifacts/PublishArtifactSet.html) allArtifacts (read-only)

The artifacts of this configuration including the artifacts of extended configurations.

[**DependencySet**](http://docs.google.com/javadoc/org/gradle/api/artifacts/DependencySet.html) allDependencies (read-only)

Gets the complete set of declared dependencies including those contributed by superconfigurations.

This method does not resolve the configuration. Therefore, the return value does not include transitive dependencies.

[**PublishArtifactSet**](http://docs.google.com/javadoc/org/gradle/api/artifacts/PublishArtifactSet.html) artifacts (read-only)

The artifacts of this configuration excluding the artifacts of extended configurations.

[**TaskDependency**](http://docs.google.com/javadoc/org/gradle/api/tasks/TaskDependency.html) buildDependencies (read-only)

Returns a TaskDependency object containing all required dependencies to build the local dependencies (e.g. project dependencies) belonging to this configuration or to one of its super configurations.

[**DependencySet**](http://docs.google.com/javadoc/org/gradle/api/artifacts/DependencySet.html) dependencies (read-only)

Gets the set of declared dependencies directly contained in this configuration (ignoring superconfigurations).

This method does not resolve the configuration. Therefore, the return value does not include transitive dependencies.

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

The description for this configuration.

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**ExcludeRule**](http://docs.google.com/javadoc/org/gradle/api/artifacts/ExcludeRule.html)> excludeRules (read-only)

The exclude rules applied for resolving any dependency of this configuration.

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html)> extendsFrom

The names of the configurations which this configuration extends from. The artifacts of the super configurations are also available in this configuration.

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html)> hierarchy (read-only)

Gets a ordered set including this configuration and all superconfigurations recursively.

[**ResolvableDependencies**](http://docs.google.com/javadoc/org/gradle/api/artifacts/ResolvableDependencies.html) incoming (read-only)

The incoming dependencies of this configuration.

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

The name of this configuration.

[**ResolutionStrategy**](http://docs.google.com/dsl/org.gradle.api.artifacts.ResolutionStrategy.html) resolutionStrategy (read-only)

The resolution strategy used by this configuration. The resolution strategy provides extra details on how to resolve this configuration. See docs for [ResolutionStrategy](http://docs.google.com/dsl/org.gradle.api.artifacts.ResolutionStrategy.html) for more info and examples.

[**ResolvedConfiguration**](http://docs.google.com/javadoc/org/gradle/api/artifacts/ResolvedConfiguration.html) resolvedConfiguration (read-only)

Resolves this configuration. This locates and downloads the files which make up this configuration, and returns a [ResolvedConfiguration](http://docs.google.com/javadoc/org/gradle/api/artifacts/ResolvedConfiguration.html) that may be used to determine information about the resolve (including errors).

[**State**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Configuration.State.html) state (read-only)

A Configuration represents a group of artifacts and their dependencies. Find more information about declaring dependencies to a configuration or about managing configurations in docs for [ConfigurationContainer](http://docs.google.com/dsl/org.gradle.api.artifacts.ConfigurationContainer.html)

Configuration is an instance of a [FileCollection](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) that contains all dependencies (see also [Configuration.getAllDependencies()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:allDependencies)) but not artifacts. If you want to refer to the artifacts declared in this configuration please use [Configuration.getArtifacts()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:artifacts) or [Configuration.getAllArtifacts()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:allArtifacts). Read more about declaring artifacts in the configuration in docs for [ArtifactHandler](http://docs.google.com/dsl/org.gradle.api.artifacts.dsl.ArtifactHandler.html) Please see the [Managing Dependency Configurations User Guide chapter](https://docs.gradle.org/current/userguide/managing_dependency_configurations.html) for more information.

boolean transitive

The transitivity of this configuration. A transitive configuration contains the transitive closure of its direct dependencies, and all their dependencies. An intransitive configuration contains only the direct dependencies. The default value is true.

boolean visible

Returns true if this is a visible configuration. A visible configuration is usable outside the project it belongs to. The default value is true.

Method details

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) copy()

Creates a copy of this configuration that only contains the dependencies directly in this configuration (without contributions from superconfigurations). The new configuration will be in the UNRESOLVED state, but will retain all other attributes of this configuration except superconfigurations. [Configuration.getHierarchy()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:hierarchy) for the copy will not include any superconfigurations.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) copy([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) dependencySpec)

Takes a closure which gets coerced into a [Spec](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html). Behaves otherwise in the same way as [Configuration.copy(org.gradle.api.specs.Spec)](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copy(org.gradle.api.specs.Spec))

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) copy([**Spec**](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html)<? super [**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html)> dependencySpec)

Creates a copy of this configuration ignoring superconfigurations (see [Configuration.copy()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copy()) but filtering the dependencies using the specified dependency spec.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) copyRecursive()

Creates a copy of this configuration that contains the dependencies directly in this configuration and those derived from superconfigurations. The new configuration will be in the UNRESOLVED state, but will retain all other attributes of this configuration except superconfigurations. [Configuration.getHierarchy()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:hierarchy) for the copy will not include any superconfigurations.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) copyRecursive([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) dependencySpec)

Takes a closure which gets coerced into a [Spec](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html). Behaves otherwise in the same way as [Configuration.copyRecursive(org.gradle.api.specs.Spec)](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copyRecursive(org.gradle.api.specs.Spec))

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) copyRecursive([**Spec**](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html)<? super [**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html)> dependencySpec)

Creates a copy of this configuration with dependencies from superconfigurations (see [Configuration.copyRecursive()](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:copyRecursive())) but filtering the dependencies using the dependencySpec.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) defaultDependencies([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**DependencySet**](http://docs.google.com/javadoc/org/gradle/api/artifacts/DependencySet.html)> action)

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

Execute the given action if the configuration has no defined dependencies when it first participates in dependency resolution. A Configuration will participate in dependency resolution when:

* The [Configuration](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) itself is resolved
* Another [Configuration](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) that extends this one is resolved
* Another [Configuration](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) that references this one as a project dependency is resolved

This method is useful for specifying default dependencies for a configuration:

configurations { conf }  
configurations['conf'].defaultDependencies { dependencies ->  
 dependencies.add(owner.project.dependencies.create("org.gradle:my-util:1.0"))  
}

A Configuration is considered empty even if it extends another, non-empty Configuration. If multiple actions are supplied, each action will be executed until the set of dependencies is no longer empty. Remaining actions will be ignored.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) exclude([**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)> excludeProperties)

Adds an exclude rule to exclude transitive dependencies for all dependencies of this configuration. You can also add exclude rules per-dependency. See [ModuleDependency.exclude(java.util.Map)](http://docs.google.com/javadoc/org/gradle/api/artifacts/ModuleDependency.html#exclude-java.util.Map-).

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) extendsFrom([**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html)... superConfigs)

Adds the given configurations to the set of configuration which this configuration extends from.

[**FileCollection**](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) fileCollection([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) dependencySpecClosure)

Takes a closure which gets coerced into a [Spec](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html). Behaves otherwise in the same way as [Configuration.fileCollection(org.gradle.api.specs.Spec)](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:fileCollection(org.gradle.api.specs.Spec)).

[**FileCollection**](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) fileCollection([**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html)... dependencies)

Resolves this configuration lazily. The resolve happens when the elements of the returned [FileCollection](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) get accessed the first time. This locates and downloads the files which make up this configuration. Only the resulting set of files belonging to specified dependencies is contained in the FileCollection.

[**FileCollection**](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) fileCollection([**Spec**](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html)<? super [**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html)> dependencySpec)

Resolves this configuration lazily. The resolve happens when the elements of the returned [FileCollection](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) get accessed the first time. This locates and downloads the files which make up this configuration. Only the resulting set of files belonging to the subset of dependencies specified by the dependencySpec is contained in the FileCollection.

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**File**](http://download.oracle.com/javase/8/docs/api/java/io/File.html)> files([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) dependencySpecClosure)

Takes a closure which gets coerced into a [Spec](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html). Behaves otherwise in the same way as [Configuration.files(org.gradle.api.specs.Spec)](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html#org.gradle.api.artifacts.Configuration:files(org.gradle.api.specs.Spec)).

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**File**](http://download.oracle.com/javase/8/docs/api/java/io/File.html)> files([**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html)... dependencies)

Resolves this configuration. This locates and downloads the files which make up this configuration. But only the resulting set of files belonging to the specified dependencies is returned.

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**File**](http://download.oracle.com/javase/8/docs/api/java/io/File.html)> files([**Spec**](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html)<? super [**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html)> dependencySpec)

Resolves this configuration. This locates and downloads the files which make up this configuration. But only the resulting set of files belonging to the subset of dependencies specified by the dependencySpec is returned.

[**TaskDependency**](http://docs.google.com/javadoc/org/gradle/api/tasks/TaskDependency.html) getTaskDependencyFromProjectDependency(boolean useDependedOn, [**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) taskName)

Returns a TaskDependency object containing dependencies on all tasks with the specified name from project dependencies related to this configuration or one of its super configurations. These other projects may be projects this configuration depends on or projects with a similarly named configuration that depend on this one based on the useDependOn argument.

**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/)