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)
  + [Methods](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#N16CE0)

### 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)

ConfigurationContainer

**Table of Contents**

[Examples](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#N16CC0)[Properties](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#N16CDB)[Methods](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#N16CE0)[Script blocks](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#N16D86)[Method details](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#N16D8B)

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

A ConfigurationContainer is responsible for declaring and managing configurations. See also [Configuration](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html).

You can obtain a ConfigurationContainer instance by calling [Project.getConfigurations()](http://docs.google.com/dsl/org.gradle.api.Project.html#org.gradle.api.Project:configurations), or using the configurations property in your build script.

The configurations in a container are accessible as read-only properties of the container, using the name of the configuration as the property name. For example:

configurations.create('myConfiguration')  
configurations.myConfiguration.transitive = false

A dynamic method is added for each configuration which takes a configuration closure. This is equivalent to calling [ConfigurationContainer.getByName(java.lang.String, groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:getByName(java.lang.String,%20groovy.lang.Closure)). For example:

configurations.create('myConfiguration')  
configurations.myConfiguration {  
 transitive = false  
}

Examples

An example showing how to refer to a given configuration by name in order to get hold of all dependencies (e.g. jars, but only)

apply plugin: 'java' //so that I can use 'compile' configuration  
  
//copying all dependencies attached to 'compile' into a specific folder  
task copyAllDependencies(type: Copy) {  
 //referring to the 'compile' configuration  
 from configurations.compile  
 into 'allLibs'  
}

An example showing how to declare and configure configurations

apply plugin: 'java' //so that I can use 'compile', 'testCompile' configurations  
  
configurations {  
 //adding a configuration:  
 myConfiguration  
  
 //adding a configuration that extends existing configuration:  
 //(testCompile was added by the java plugin)  
 myIntegrationTestsCompile.extendsFrom(testCompile)  
  
 //configuring existing configurations not to put transitive dependencies on the compile classpath  
 //this way you can avoid issues with implicit dependencies to transitive libraries  
 compile.transitive = false  
 testCompile.transitive = false  
}

Examples on configuring the *resolution strategy* - see docs for [ResolutionStrategy](http://docs.google.com/dsl/org.gradle.api.artifacts.ResolutionStrategy.html) Please see the [Managing Dependency Configurations User Guide chapter](https://docs.gradle.org/current/userguide/managing_dependency_configurations.html) for more information.

Properties

No properties

Methods

| Method | Description |
| --- | --- |
| [create](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:create(java.lang.String))(name) | Creates a new item with the given name, adding it to this container. |
| [create](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:create(java.lang.String,%20groovy.lang.Closure))(name, configureClosure) | Creates a new item with the given name, adding it to this container, then configuring it with the given closure. |
| [create](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:create(java.lang.String,%20org.gradle.api.Action))(name, configureAction) | Creates a new item with the given name, adding it to this container, then configuring it with the given action. |
| [detachedConfiguration](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:detachedConfiguration(org.gradle.api.artifacts.Dependency%5B%5D))(dependencies) | Creates a configuration, but does not add it to this container. |
| [getAt](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:getAt(java.lang.String))(name) | Locates an object by name, failing if there is no such task. This method is identical to [NamedDomainObjectCollection.getByName(java.lang.String)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectCollection.html#org.gradle.api.NamedDomainObjectCollection:getByName(java.lang.String)). You can call this method in your build script by using the groovy [] operator. |
| [getByName](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:getByName(java.lang.String))(name) | Locates an object by name, failing if there is no such object. |
| [getByName](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:getByName(java.lang.String,%20groovy.lang.Closure))(name, configureClosure) | Locates an object by name, failing if there is no such object. The given configure closure is executed against the object before it is returned from this method. The object is passed to the closure as its delegate. |
| [getByName](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:getByName(java.lang.String,%20org.gradle.api.Action))(name, configureAction) | Locates an object by name, failing if there is no such object. The given configure action is executed against the object before it is returned from this method. |
| [maybeCreate](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:maybeCreate(java.lang.String))(name) | Looks for an item with the given name, creating and adding it to this container if it does not exist. |
| [named](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:named(java.lang.String))(name) | Incubating  Locates a object by name, without triggering its creation or configuration, failing if there is no such object. |
| [register](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:register(java.lang.String))(name) | Incubating  Defines a new object, which will be created when it is required. A object is 'required' when the object is located using query methods such as [NamedDomainObjectCollection.getByName(java.lang.String)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectCollection.html#org.gradle.api.NamedDomainObjectCollection:getByName(java.lang.String)) or when [Provider.get()](http://docs.google.com/javadoc/org/gradle/api/provider/Provider.html#get--) is called on the return value of this method. |
| [register](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html#org.gradle.api.artifacts.ConfigurationContainer:register(java.lang.String,%20org.gradle.api.Action))(name, configurationAction) | Incubating  Defines a new object, which will be created and configured when it is required. An object is 'required' when the object is located using query methods such as [NamedDomainObjectCollection.getByName(java.lang.String)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectCollection.html#org.gradle.api.NamedDomainObjectCollection:getByName(java.lang.String)) or when [Provider.get()](http://docs.google.com/javadoc/org/gradle/api/provider/Provider.html#get--) is called on the return value of this method. |

Script blocks

No script blocks

Method details

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

Creates a new item with the given name, adding it to this container.

T create([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name, [**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) configureClosure)

Creates a new item with the given name, adding it to this container, then configuring it with the given closure.

T create([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name, [**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super T> configureAction)

Creates a new item with the given name, adding it to this container, then configuring it with the given action.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) detachedConfiguration([**Dependency**](http://docs.google.com/javadoc/org/gradle/api/artifacts/Dependency.html)... dependencies)

Creates a configuration, but does not add it to this container.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) getAt([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name)

Locates an object by name, failing if there is no such task. This method is identical to [NamedDomainObjectCollection.getByName(java.lang.String)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectCollection.html#org.gradle.api.NamedDomainObjectCollection:getByName(java.lang.String)). You can call this method in your build script by using the groovy [] operator.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) getByName([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name)

Locates an object by name, failing if there is no such object.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) getByName([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name, [**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) configureClosure)

Locates an object by name, failing if there is no such object. The given configure closure is executed against the object before it is returned from this method. The object is passed to the closure as its delegate.

[**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html) getByName([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name, [**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**Configuration**](http://docs.google.com/dsl/org.gradle.api.artifacts.Configuration.html)> configureAction)

Locates an object by name, failing if there is no such object. The given configure action is executed against the object before it is returned from this method.

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

Looks for an item with the given name, creating and adding it to this container if it does not exist.

[**NamedDomainObjectProvider**](http://docs.google.com/javadoc/org/gradle/api/NamedDomainObjectProvider.html)<T> named([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name)

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

Locates a object by name, without triggering its creation or configuration, failing if there is no such object.

[**NamedDomainObjectProvider**](http://docs.google.com/javadoc/org/gradle/api/NamedDomainObjectProvider.html)<T> register([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name)

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

Defines a new object, which will be created when it is required. A object is 'required' when the object is located using query methods such as [NamedDomainObjectCollection.getByName(java.lang.String)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectCollection.html#org.gradle.api.NamedDomainObjectCollection:getByName(java.lang.String)) or when [Provider.get()](http://docs.google.com/javadoc/org/gradle/api/provider/Provider.html#get--) is called on the return value of this method.

It is generally more efficient to use this method instead of [NamedDomainObjectContainer.create(java.lang.String)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectContainer.html#org.gradle.api.NamedDomainObjectContainer:create(java.lang.String)), as that methods will eagerly create he object, regardless of whether that object is required for the current build or not. This method, on the other hand, will defer creation until required.

[**NamedDomainObjectProvider**](http://docs.google.com/javadoc/org/gradle/api/NamedDomainObjectProvider.html)<T> register([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name, [**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super T> configurationAction)

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

Defines a new object, which will be created and configured when it is required. An object is 'required' when the object is located using query methods such as [NamedDomainObjectCollection.getByName(java.lang.String)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectCollection.html#org.gradle.api.NamedDomainObjectCollection:getByName(java.lang.String)) or when [Provider.get()](http://docs.google.com/javadoc/org/gradle/api/provider/Provider.html#get--) is called on the return value of this method.

It is generally more efficient to use this method instead of [NamedDomainObjectContainer.create(java.lang.String, org.gradle.api.Action)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectContainer.html#org.gradle.api.NamedDomainObjectContainer:create(java.lang.String,%20org.gradle.api.Action)) or [NamedDomainObjectContainer.create(java.lang.String)](http://docs.google.com/dsl/org.gradle.api.NamedDomainObjectContainer.html#org.gradle.api.NamedDomainObjectContainer:create(java.lang.String)), as those methods will eagerly create and configure the object, regardless of whether that object is required for the current build or not. This method, on the other hand, will defer creation and configuration until required.

**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/)