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.publish.maven.MavenPublication.html#N1B913)
  + [Methods](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#N1B94C)

### 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)

MavenPublication

**Table of Contents**

[Example of publishing a Java module with a source artifact and a customized POM](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#N1B90D)[Properties](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#N1B913)[Methods](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#N1B94C)[Script blocks](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#N1B9B0)[Property details](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#N1B9B5)[Method details](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#N1B9FA)

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

A MavenPublication is the representation/configuration of how Gradle should publish something in Maven format. You directly add a named Maven Publication the project's publishing.publications container by providing [MavenPublication](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html) as the type.

publishing {  
 publications {  
 myPublicationName(MavenPublication) {  
 // Configure the publication here  
 }  
 }  
}

The default Maven POM identifying attributes are mapped as follows:

* groupId - project.group
* artifactId - project.name
* version - project.version

For certain common use cases, it's often sufficient to specify the component to publish, and nothing more ([MavenPublication.from(org.gradle.api.component.SoftwareComponent)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:from(org.gradle.api.component.SoftwareComponent)). The published component is used to determine which artifacts to publish, and which dependencies should be listed in the generated POM file.

To add additional artifacts to the set published, use the [MavenPublication.artifact(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifact(java.lang.Object)) and [MavenPublication.artifact(java.lang.Object, org.gradle.api.Action)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifact(java.lang.Object,%20org.gradle.api.Action)) methods. You can also completely replace the set of published artifacts using [MavenPublication.setArtifacts(java.lang.Iterable)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:setArtifacts(java.lang.Iterable)). Together, these methods give you full control over what artifacts will be published.

To customize the metadata published in the generated POM, set properties, e.g. [MavenPom.getDescription()](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPom.html#org.gradle.api.publish.maven.MavenPom:description), on the POM returned via the [MavenPublication.getPom()](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:pom) method or directly by an action (or closure) passed into [MavenPublication.pom(org.gradle.api.Action)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:pom(org.gradle.api.Action)). As a last resort, it is possible to modify the generated POM using the [MavenPom.withXml(org.gradle.api.Action)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPom.html#org.gradle.api.publish.maven.MavenPom:withXml(org.gradle.api.Action)) method.

Example of publishing a Java module with a source artifact and a customized POM

apply plugin: "java"  
apply plugin: "maven-publish"  
  
task sourceJar(type: Jar) {  
 from sourceSets.main.allJava  
 classifier "sources"  
}  
  
publishing {  
 publications {  
 myPublication(MavenPublication) {  
 from components.java  
 artifact sourceJar  
 pom {  
 name = "Demo"  
 description = "A demonstration of Maven POM customization"  
 url = "http://www.example.com/project"  
 licenses {  
 license {  
 name = "The Apache License, Version 2.0"  
 url = "http://www.apache.org/licenses/LICENSE-2.0.txt"  
 }  
 }  
 developers {  
 developer {  
 id = "johnd"  
 name = "John Doe"  
 email = "john.doe@example.com"  
 }  
 }  
 scm {  
 connection = "scm:svn:http://subversion.example.com/svn/project/trunk/"  
 developerConnection = "scm:svn:https://subversion.example.com/svn/project/trunk/"  
 url = "http://subversion.example.com/svn/project/trunk/"  
 }  
 }  
 }  
 }  
}

Properties

| Property | Description |
| --- | --- |
| [artifactId](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifactId) | The artifactId for this publication. |
| [artifacts](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifacts) | The complete set of artifacts for this publication. |
| [groupId](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:groupId) | The groupId for this publication. |
| [pom](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:pom) | The POM that will be published. |
| [version](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:version) | The version for this publication. |

Methods

| Method | Description |
| --- | --- |
| [artifact](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifact(java.lang.Object))(source) | Creates a custom [MavenArtifact](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html) to be included in the publication. The artifact method can take a variety of input: |
| [artifact](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifact(java.lang.Object,%20org.gradle.api.Action))(source, config) | Creates an [MavenArtifact](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html) to be included in the publication, which is configured by the associated action. The first parameter is used to create a custom artifact and add it to the publication, as per [MavenPublication.artifact(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifact(java.lang.Object)). The created [MavenArtifact](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html) is then configured using the supplied action, which can override the extension or classifier of the artifact. This method also accepts the configure action as a closure argument, by type coercion. |
| [from](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:from(org.gradle.api.component.SoftwareComponent))(component) | Provides the software component that should be published. |
| [pom](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:pom(org.gradle.api.Action))(configure) | Configures the POM that will be published. The supplied action will be executed against the [MavenPublication.getPom()](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:pom) result. This method also accepts a closure argument, by type coercion. |
| [setArtifacts](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:setArtifacts(java.lang.Iterable))(sources) | Clears any previously added artifacts from [MavenPublication.getArtifacts()](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifacts) and creates artifacts from the specified sources. Each supplied source is interpreted as per [MavenPublication.artifact(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifact(java.lang.Object)). For example, to exclude the dependencies declared by a component and instead use a custom set of artifacts: |

Script blocks

No script blocks

Property details

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

The artifactId for this publication.

[**MavenArtifactSet**](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifactSet.html) artifacts

The complete set of artifacts for this publication.

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

The groupId for this publication.

[**MavenPom**](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPom.html) pom (read-only)

The POM that will be published.

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

The version for this publication.

Method details

[**MavenArtifact**](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html) artifact([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) source)

Creates a custom [MavenArtifact](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html) to be included in the publication. The artifact method can take a variety of input:

* A [PublishArtifact](http://docs.google.com/javadoc/org/gradle/api/artifacts/PublishArtifact.html) instance. Extension and classifier values are taken from the wrapped instance.
* An [AbstractArchiveTask](http://docs.google.com/dsl/org.gradle.api.tasks.bundling.AbstractArchiveTask.html) instance. Extension and classifier values are taken from the wrapped instance.
* Anything that can be resolved to a [File](http://download.oracle.com/javase/8/docs/api/java/io/File.html) via the [Project.file(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.Project.html#org.gradle.api.Project:file(java.lang.Object)) method. Extension and classifier values are interpolated from the file name.
* A [Map](http://download.oracle.com/javase/8/docs/api/java/util/Map.html) that contains a 'source' entry that can be resolved as any of the other input types, including file. This map can contain a 'classifier' and an 'extension' entry to further configure the constructed artifact.

The following example demonstrates the addition of various custom artifacts.

apply plugin: "maven-publish"  
  
task sourceJar(type: Jar) {  
 classifier "sources"  
}  
  
publishing {  
 publications {  
 maven(MavenPublication) {  
 artifact sourceJar // Publish the output of the sourceJar task  
 artifact 'my-file-name.jar' // Publish a file created outside of the build  
 artifact source: sourceJar, classifier: 'src', extension: 'zip'  
 }  
 }  
}

[**MavenArtifact**](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html) artifact([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) source, [**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**MavenArtifact**](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html)> config)

Creates an [MavenArtifact](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html) to be included in the publication, which is configured by the associated action. The first parameter is used to create a custom artifact and add it to the publication, as per [MavenPublication.artifact(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifact(java.lang.Object)). The created [MavenArtifact](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenArtifact.html) is then configured using the supplied action, which can override the extension or classifier of the artifact. This method also accepts the configure action as a closure argument, by type coercion.

apply plugin: "maven-publish"  
  
task sourceJar(type: Jar) {  
 classifier "sources"  
}  
  
publishing {  
 publications {  
 maven(MavenPublication) {  
 artifact(sourceJar) {  
 // These values will be used instead of the values from the task. The task values will not be updated.  
 classifier "src"  
 extension "zip"  
 }  
 artifact("my-docs-file.htm") {  
 classifier "documentation"  
 extension "html"  
 }  
 }  
 }  
}

void from([**SoftwareComponent**](http://docs.google.com/javadoc/org/gradle/api/component/SoftwareComponent.html) component)

Provides the software component that should be published.

* Any artifacts declared by the component will be included in the publication.
* The dependencies declared by the component will be included in the published meta-data.

Currently 2 types of component are supported: 'components.java' (added by the JavaPlugin) and 'components.web' (added by the WarPlugin). For any individual MavenPublication, only a single component can be provided in this way. The following example demonstrates how to publish the 'java' component to a Maven repository.

apply plugin: "java"  
apply plugin: "maven-publish"  
  
publishing {  
 publications {  
 maven(MavenPublication) {  
 from components.java  
 }  
 }  
}

void pom([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**MavenPom**](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPom.html)> configure)

Configures the POM that will be published. The supplied action will be executed against the [MavenPublication.getPom()](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:pom) result. This method also accepts a closure argument, by type coercion.

void setArtifacts([**Iterable**](http://download.oracle.com/javase/8/docs/api/java/lang/Iterable.html)<?> sources)

Clears any previously added artifacts from [MavenPublication.getArtifacts()](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifacts) and creates artifacts from the specified sources. Each supplied source is interpreted as per [MavenPublication.artifact(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.publish.maven.MavenPublication.html#org.gradle.api.publish.maven.MavenPublication:artifact(java.lang.Object)). For example, to exclude the dependencies declared by a component and instead use a custom set of artifacts:

apply plugin: "java"  
apply plugin: "maven-publish"  
  
task sourceJar(type: Jar) {  
 classifier "sources"  
}  
  
publishing {  
 publications {  
 maven(MavenPublication) {  
 from components.java  
 artifacts = ["my-custom-jar.jar", sourceJar]  
 }  
 }  
}

**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/)