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)

[PDF](http://docs.google.com/userguide/userguide.pdf)

* [User Manual Home](http://docs.google.com/userguide/userguide.html)
* [Release Notes](http://docs.google.com/release-notes.html)
* [Installing Gradle](http://docs.google.com/userguide/installation.html)
* [Tutorials](https://guides.gradle.org/)

### Reference

* [Groovy DSL Reference](http://docs.google.com/dsl/)
* [Gradle API Javadoc](http://docs.google.com/javadoc/)
* [Core Plugins](http://docs.google.com/userguide/plugin_reference.html)
* [Gradle & Third-party Tools](http://docs.google.com/userguide/third_party_integration.html)

### Getting Started

* [Creating New Gradle Builds](https://guides.gradle.org/creating-new-gradle-builds/)
* [Creating Build Scans](https://guides.gradle.org/creating-build-scans/)
* [Migrating From Maven](https://guides.gradle.org/migrating-from-maven/)

### Running Gradle Builds

* [Command-Line Interface](http://docs.google.com/userguide/command_line_interface.html)
* [Customizing Execution](#gjdgxs)
  + [Configuring the Build Environment](http://docs.google.com/userguide/build_environment.html)
  + [Configuring the Gradle Daemon](http://docs.google.com/userguide/gradle_daemon.html)
  + [Initialization Scripts](http://docs.google.com/userguide/init_scripts.html)
* [Directory Layout](http://docs.google.com/userguide/directory_layout.html)
* [Executing Multi-Project Builds](http://docs.google.com/userguide/intro_multi_project_builds.html)
* [Gradle Wrapper](http://docs.google.com/userguide/gradle_wrapper.html)
* [Troubleshooting](http://docs.google.com/userguide/troubleshooting.html)
* [Using Build Scans](https://docs.gradle.com/build-scan-plugin)
* [Enabling and Configuring the Build Cache](http://docs.google.com/userguide/build_cache.html)
* [Integrating Separate Gradle Builds (Composite Builds)](http://docs.google.com/userguide/composite_builds.html)

### Authoring Gradle Builds

* [Fundamentals](#30j0zll)
  + [Introducing the Basics of Build Scripts](http://docs.google.com/userguide/tutorial_using_tasks.html)
  + [Working with Tasks](http://docs.google.com/userguide/more_about_tasks.html)
  + [Learning More About Build Scripts](http://docs.google.com/userguide/writing_build_scripts.html)
  + [Working with Files](http://docs.google.com/userguide/working_with_files.html)
  + [Creating Custom Task Types](http://docs.google.com/userguide/custom_tasks.html)
  + [Using Gradle Plugins](http://docs.google.com/userguide/plugins.html)
  + [The Standard Gradle Plugins](http://docs.google.com/userguide/standard_plugins.html)
  + [Understanding the Build Lifecycle](http://docs.google.com/userguide/build_lifecycle.html)
  + [Working with Logging](http://docs.google.com/userguide/logging.html)
  + [Configuring Multi-Project Builds](http://docs.google.com/userguide/multi_project_builds.html)
* [Best Practices](#1fob9te)
  + [Authoring Maintainable Build Scripts](http://docs.google.com/userguide/authoring_maintainable_build_scripts.html)
  + [Organizing Gradle Projects](http://docs.google.com/userguide/organizing_gradle_projects.html)
  + [Optimizing Build Performance](https://guides.gradle.org/performance/)
  + [Using the Build Cache](https://guides.gradle.org/using-build-cache/)
* [Dependency Management](#3znysh7)
  + [Introduction to Dependency Management](http://docs.google.com/userguide/introduction_dependency_management.html)
  + [Dependency Management Terminology](http://docs.google.com/userguide/dependency_management_terminology.html)
  + [Dependency Types](http://docs.google.com/userguide/dependency_types.html)
  + [Repository Types](http://docs.google.com/userguide/repository_types.html)
  + [Declaring Dependencies](http://docs.google.com/userguide/declaring_dependencies.html)
  + [Declaring Repositories](http://docs.google.com/userguide/declaring_repositories.html)
  + [Inspecting Dependencies](http://docs.google.com/userguide/inspecting_dependencies.html)
  + [Managing Dependency Configurations](http://docs.google.com/userguide/managing_dependency_configurations.html)
  + [Managing Transitive Dependencies](http://docs.google.com/userguide/managing_transitive_dependencies.html)
  + [Dependency Locking](http://docs.google.com/userguide/dependency_locking.html)
  + [Troubleshooting Dependency Resolution](http://docs.google.com/userguide/troubleshooting_dependency_resolution.html)
  + [Customizing Dependency Resolution Behavior](http://docs.google.com/userguide/customizing_dependency_resolution_behavior.html)
  + [Dependency Cache Internals](http://docs.google.com/userguide/dependency_cache.html)
  + [Working with Dependencies](http://docs.google.com/userguide/working_with_dependencies.html)
* [Publishing Artifacts](http://docs.google.com/userguide/artifact_management.html)
* [C++ Projects](#2et92p0)
  + [Building Native Software](http://docs.google.com/userguide/native_software.html)
  + [Software Model Concepts](http://docs.google.com/userguide/software_model_concepts.html)
  + [Rule-based Model Configuration](http://docs.google.com/userguide/software_model.html)
  + [Implementing Model Rules in a Plugin](http://docs.google.com/userguide/rule_source.html)
  + [Extending the Software Model](http://docs.google.com/userguide/software_model_extend.html)
* [Java Projects](#tyjcwt)
  + [Building Java & JVM projects](http://docs.google.com/userguide/building_java_projects.html)
  + [Testing Java & JVM projects](http://docs.google.com/userguide/java_testing.html)
* [Advanced Techniques](#3dy6vkm)
  + [Configuring Tasks Lazily](http://docs.google.com/userguide/lazy_configuration.html)
  + [Developing Parallel Tasks](https://guides.gradle.org/using-the-worker-api/)
  + [Testing Your Build with TestKit](http://docs.google.com/userguide/test_kit.html)
  + [Using Ant from Gradle](http://docs.google.com/userguide/ant.html)
* [Sample Gradle builds](#1t3h5sf)
  + [Groovy DSL Samples](https://github.com/gradle/gradle/tree/master/subprojects/docs/src/samples)
  + [Kotlin DSL Samples](https://github.com/gradle/kotlin-dsl/tree/master/samples)

### Extending Gradle

* [Writing Custom Plugins](http://docs.google.com/userguide/custom_plugins.html)
* [Plugin Development Guides](https://gradle.org/guides/?q=Plugin+Development)

[Edit this page](https://github.com/gradle/gradle/edit/master/subprojects/docs/src/docs/userguide/)

# Directory Layout

Contents

[Gradle user home directory](#4d34og8)

[Project root directory](#2s8eyo1)

Gradle uses two main directories to perform and manage its work: the [Gradle user home directory](#4d34og8) and the [Project root directory](#2s8eyo1). The following two sections describe what is stored in each of them and how transient files and directories are cleaned up.

[Gradle user home directory](#4d34og8)

The Gradle user home directory ($USER\_HOME/.gradle by default) is used to store global configuration properties and initialization scripts as well as caches and log files. It is roughly structured as follows:

├── caches (1)  
│   ├── 4.8 (2)  
│   ├── 4.9 (2)  
│   ├── ⋮  
│   ├── jars-3 (3)  
│   └── modules-2 (3)  
├── daemon (4)  
│   ├── ⋮  
│   ├── 4.8  
│   └── 4.9  
├── init.d (5)  
│   └── my-setup.gradle  
├── wrapper  
│ └── dists (6)  
│ ├── ⋮  
│ ├── gradle-4.8-bin  
│ ├── gradle-4.9-all  
│ └── gradle-4.9-bin  
└── gradle.properties (7)

1. Global cache directory (for everything that’s not project-specific)
2. Version-specific caches (e.g. to support incremental builds)
3. Shared caches (e.g. for artifacts of dependencies)
4. Registry and logs of the [Gradle Daemon](http://docs.google.com/gradle_daemon.html#gradle_daemon)
5. Global [initialization scripts](http://docs.google.com/init_scripts.html#init_scripts)
6. Distributions downloaded by the [Gradle Wrapper](http://docs.google.com/gradle_wrapper.html#gradle_wrapper)
7. Global [Gradle configuration properties](http://docs.google.com/build_environment.html#sec:gradle_configuration_properties)

[Cleanup of caches and distributions](#17dp8vu)

From version 4.10 onwards, Gradle automatically cleans its user home directory. The cleanup runs in the background when the Gradle daemon is stopped or shuts down. If using --no-daemon, it runs in the foreground after the build session with a visual progress indicator.

The following cleanup strategies are applied periodically (at most every 24 hours):

* Version-specific caches in caches/<gradle-version>/ are checked for whether they are still in use. If not, directories for release versions are deleted after 30 days of inactivity, snapshot versions after 7 days of inactivity.
* Shared caches in caches/ (e.g. jars-\*) are checked for whether they are still in use. If there’s no Gradle version that still uses them, they are deleted.
* Files in shared caches used by the current Gradle version in caches/ (e.g. jars-3 or modules-2) are checked for when they were last accessed. Depending on whether the file can be recreated locally or would have to be downloaded from a remote repository again, it will be deleted after 7 or 30 days of not being accessed, respectively.
* Gradle distributions in wrapper/dists/ are checked for whether they are still in use, i.e. whether there’s a corresponding version-specific cache directory. Unused distributions are deleted.

[Project root directory](#2s8eyo1)

The project root directory contains all source files that are part of your project. In addition, it contains files and directories that are generated by Gradle such as .gradle and build. While the former are usually checked in to source control, the latter are transient files used by Gradle to support features like incremental builds. Overall, the anatomy of a typical project root directory looks roughly as follows:

├── .gradle (1)  
│   ├── 4.8 (2)  
│   ├── 4.9 (2)  
│   └── ⋮  
├── build (3)  
├── gradle  
│   └── wrapper (4)  
├── build.gradle *or* build.gradle.kts (5)  
├── gradle.properties (6)  
├── gradlew (7)  
├── gradlew.bat (7)  
└── settings.gradle *or* settings.gradle.kts (8)

1. Project-specific cache directory generated by Gradle
2. Version-specific caches (e.g. to support incremental builds)
3. The build directory of this project into which Gradle generates all build artifacts.
4. Contains the JAR file and configuration of the [Gradle Wrapper](http://docs.google.com/gradle_wrapper.html#gradle_wrapper)
5. The project’s Gradle build script
6. Project-specific [Gradle configuration properties](http://docs.google.com/build_environment.html#sec:gradle_configuration_properties)
7. Scripts for executing builds using the [Gradle Wrapper](http://docs.google.com/gradle_wrapper.html#gradle_wrapper)
8. The project’s [settings file](http://docs.google.com/build_lifecycle.html#sec:settings_file)

[Project cache cleanup](#3rdcrjn)

From version 4.10 onwards, Gradle automatically cleans the project-specific cache directory. After building the project, version-specific cache directories in .gradle/<gradle-version>/ are checked periodically (at most every 24 hours) for whether they are still in use. They are deleted if they haven’t been used for 7 days.

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/)