

06 Spring 5.x源代码编译并导入到eclipse或者ide

更新时间：2020-05-28 11:36:49



“时间像海绵里的水，只要你愿意挤，总还是有的。——鲁迅”

子贡问为仁。子曰：“工欲善其事，必先利其器。居是邦也，事其大夫之贤者，友其士之仁者。”

——孔子（春秋）《论语·卫灵公》

背景

想要玩转 **Spring**，就需要可以随意修改 **Spring** 的源码，想要随意修改 **Spring** 的源码，就需要将 **Spring** 的源代码导入到我们常用的编译器如 **IDEA** 或者 **Eclipse** 中，**Eclipse** 导入的方式和 **IDEA** 相似，我们仅以导入 **Eclipse** 为例介绍 **Spring** 源码构建及导入的过程，**IDEA** 的类似，就不一一赘述了。

步骤

第一步，**jdk** 的安装

作为 **Java** 开发者恐怕不需要在此叙说，本文选择 **jdk8** 版本。

第二步，**Github** 下载最新 **master** 的代码

一种 **clone** 方式：

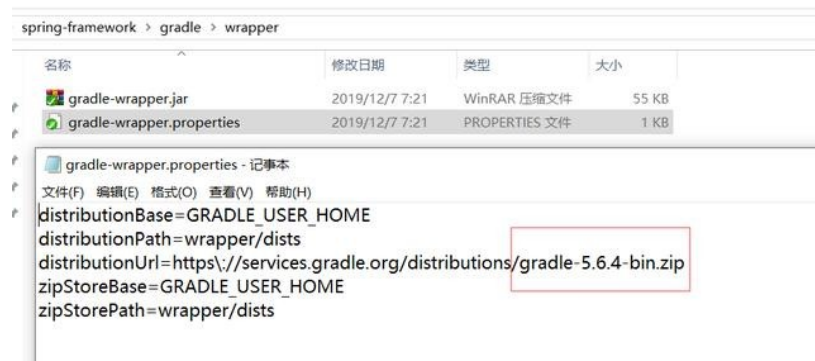
```
git clone https://github.com/spring-projects/spring-framework.git
```

另外一种方式:

http 下载 <https://github.com/spring-projects/spring-framework>

第三步, 选择按照对应版本的 **Gradle** (很重要)

在 **Spring-framework** 中定义好了 **master** 构建的 **Gradle** 版本:



第四步, 下载 **Gradle**

本次应该下载对应的是 **gradle-5.6.4-bin.zip**, 去[官网](https://services.gradle.org/distributions/)下载。

gradle-6.0-all.zip	08-Nov-2019 19:25 +0100	134.65M
gradle-6.0-all.zip.sha256	08-Nov-2019 19:25 +0100	64.00B
gradle-6.0-rc-3-wrapper.jar.sha256	05-Nov-2019 19:12 +0100	64.00B
gradle-6.0-rc-3-src.zip	05-Nov-2019 19:12 +0100	37.28M
gradle-6.0-rc-3-src.zip.sha256	05-Nov-2019 19:12 +0100	64.00B
gradle-6.0-rc-3-bin.zip	05-Nov-2019 19:12 +0100	90.72M
gradle-6.0-rc-3-bin.zip.sha256	05-Nov-2019 19:12 +0100	64.00B
gradle-6.0-rc-3-all.zip	05-Nov-2019 19:11 +0100	134.85M
gradle-6.0-rc-3-all.zip.sha256	05-Nov-2019 19:12 +0100	64.00B
gradle-5.6.4-wrapper.jar.sha256	01-Nov-2019 21:55 +0100	64.00B
gradle-5.6.4-src.zip	01-Nov-2019 21:55 +0100	37.34M
gradle-5.6.4-src.zip.sha256	01-Nov-2019 21:55 +0100	64.00B
gradle-5.6.4-bin.zip	01-Nov-2019 21:55 +0100	89.55M
gradle-5.6.4-bin.zip.sha256	01-Nov-2019 21:55 +0100	64.00B
gradle-5.6.4-all.zip	01-Nov-2019 21:55 +0100	133.32M
gradle-5.6.4-all.zip.sha256	01-Nov-2019 21:55 +0100	64.00B

第五步, 解压缩到本地目录

第六步, 配置 **GRADLE_HOME** 的环境变量

第七步, 将 **%GRADLE_HOME%/bin** 配置到 **PATH** 中

第八步, 验证

配置完, 验证一下:

```

C:\Users\S6137>gradle -v

Welcome to Gradle 5.6.4!

Here are the highlights of this release:
- Incremental Groovy compilation
- Groovy compile avoidance
- Test fixtures for Java projects
- Manage plugin versions via settings script

For more details see https://docs.gradle.org/5.6.4/release-notes.html

-----
Gradle 5.6.4
-----

Build time: 2019-11-01 20:42:00 UTC
Revision: dd870424f9bd8e195d614dc14bb140f43c22da98

Kotlin: 1.3.41
Groovy: 2.5.4
Ant: Apache Ant(TM) version 1.9.14 compiled on March 12 2019
JVM: 1.8.0_91 (Oracle Corporation 25.91-b15)
OS: Windows 10 10.0 amd64

C:\Users\S6137>

```

第九步，国内配置阿里云依赖库

在 gradle home 下的 init.d 目录下加入一个名叫 init.gradle 的文件，然后把阿里云的配置加进去，内容如下：

```

buildscript {
    repositories {
        maven{ url 'https://maven.aliyun.com/repository/public'}
    }
    dependencies {
        classpath 'org.asciidoctor:asciidoctorj-pdf:1.5.0-alpha.16'
        classpath 'io.spring.asciidoctor:spring-asciidoctor-extensions:0.1.3.RELEASE'
    }
}

```

第十步，转成 **Eclipse** 工程

在这里注意，如果网速不好的话，转成 **Eclipse** 工程时间比较长。

进入命令行，到相应的 Spring-framework 下载目录的根目录，执行 Gradle 命令：

gradle clean eclipse

如下图所示：

```

C:\work\spring-framework>gradle clean eclipse
Starting a Gradle Daemon (subsequent builds will be faster)

BUILD SUCCESSFUL in 2h 10m 31s
169 actionable tasks: 144 executed, 25 up-to-date
C:\work\spring-framework>

```

第十一步，添加依赖项

gradle objenesisRepackJar

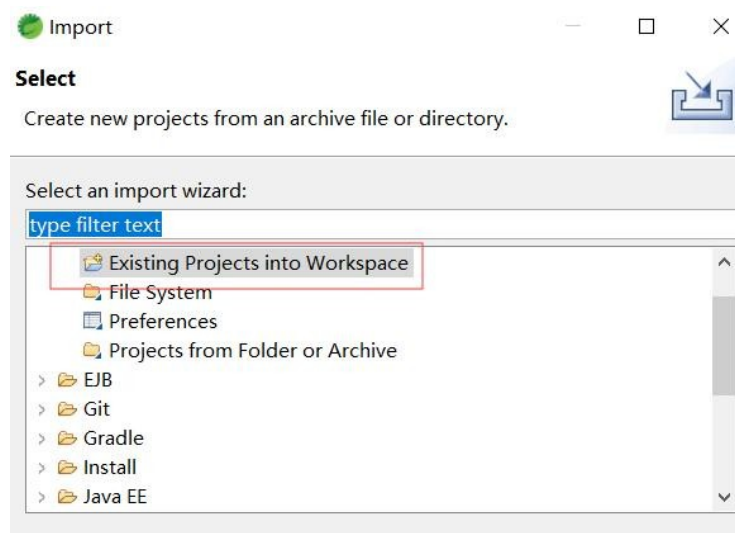
gradle cglibRepackJar

```
C:\work\spring-framework>gradle objenesisRepackJar

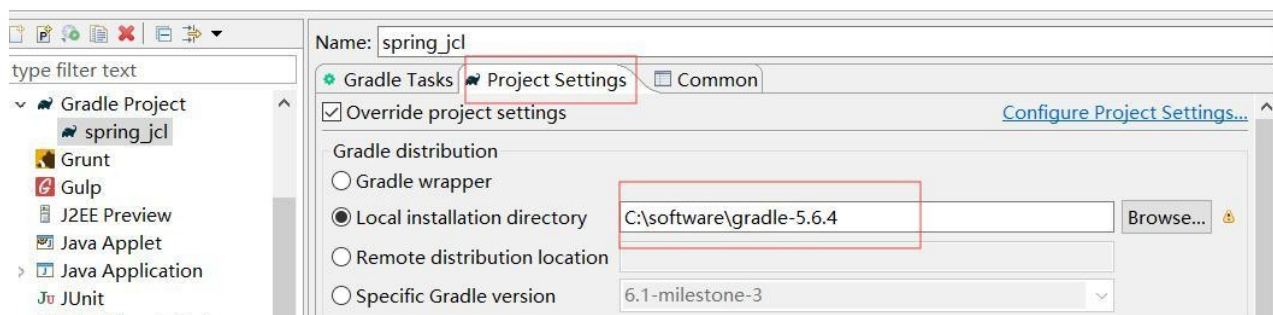
BUILD SUCCESSFUL in 1m 4s
1 actionable task: 1 executed
C:\work\spring-framework>gradle cglibRepackJar

BUILD SUCCESSFUL in 1m 34s
1 actionable task: 1 executed
C:\work\spring-framework>
```

第十二步，将要调试的子项目导入到 **Eclipse**

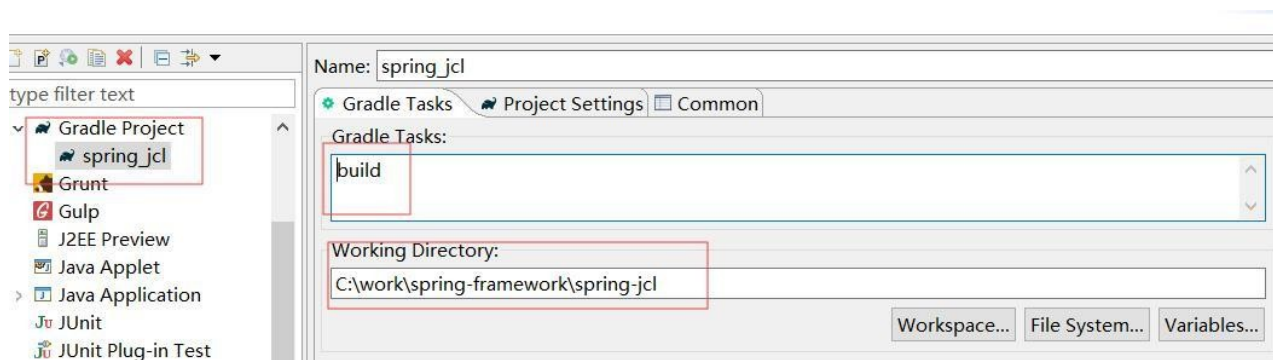


第十三步，**Eclipse** 配置 **Gradle**



第十四步，运行 **Gradle** 子项目

Run --> run config --> new Gradle task:build



Spring-core 依赖于 jcl，其他的可以看工程依赖。

注意：有些项目需要 **Groovy** 和 **kotlin** 插件支持，请自行安装。

总结

源代码的阅读最忌是“动口不动手”，这里讲的不动手是指只是拿着源代码一遍遍的看或者机械的 **Debug**，而不是去修改源代码，不断的验证自己的想法。只有不断的尝试着自己对框架的了解，只有不断的验证框架的设计和自己的设计是否吻合，才能学到框架的精髓，否则不过是竹篮打水一场空罢了，浪费了精力而无所收获。

想要验证自己的想法或者深究框架本身的原理，就需要将程序运行的每一步都可以 **Debug**，每一步都可以修改，这就需要将源码编译，并将工程导入的我们常用的编辑器如 **IDEA**, **Eclipse**, **STS** 等作为工程跑起来，我们在编译后的源码工程上 **Debug** 或者修改源码，使之按照我们自己的思路来运行。

}



05 Spring 5.x特性、设计理念及架构

07 别整虚的！揭开Spring IoC、DI的神秘面纱

