

SpringBoot的配置类和常见条件注解

01、分析

在项目中，有时会遇到我们的@Configuration、@Bean、Service等等的bean组件需要依条件按需加载的情况。那么Spring Boot怎么做的呢？它为此定义了许多有趣的条件，当我们将它们运用到我们的bean上时，就可以实现动态的加载控制了。

SpringBoot 定义了许多条件化注解，可以将它们用到配置类上，以说明生效条件。

条件化注解	生效条件
@ConditionalOnBean	配置了特定的 Bean。
@ConditionalOnMissingBean	没有配置特定的 Bean。你容器没有我加载，如果有就不加载了
@ConditionalOnClass	Classpath 里有指定的类。
@ConditionalOnMissingClass	Classpath 里没有指定的类。
@ConditionalOnExpression	给定的 SpringExpressionLanguage(SpEL) 表达式计算结果为 true。
@ConditionalOnJava	Java 的版本匹配特定值或者一个范围值。
@ConditionalOnJndi	JNDI 参数必须存在。
@ConditionalOnProperty	指定的属性有一个明确的值。
@ConditionalOnResource	Classpath 里存在指定的资源。
@ConditionalOnWebApplication	是一个 Web 应用程序。
@ConditionalOnNotWebApplication	不是一个 Web 应用程序。

比如 SpringBootWebSecurityConfiguration.java 的源码为：

```

1  @Configuration
2  @ConditionalOnClass(WebSecurityConfigurerAdapter.class)
3  @ConditionalOnMissingBean(WebSecurityConfigurerAdapter.class)
4  @ConditionalOnWebApplication(type = Type.SERVLET)
5  public class SpringBootWebSecurityConfiguration {
6
7      @Configuration
8      @Order(SecurityProperties.BASIC_AUTH_ORDER)
9      static class DefaultConfigurerAdapter extends
WebSecurityConfigurerAdapter {
10
11      }
12
13 }

```

SpringBootWebSecurityConfiguration 配置了三个条件化注解，也就是说 SpringBootWebSecurityConfiguration 必须在这些条件都满足的情况下，才能生效：

1. Classpath 里有 WebSecurityConfigurerAdapter.class。
2. 没有配置 WebSecurityConfigurerAdapter Bean。
3. 是一个 Web 应用程序。

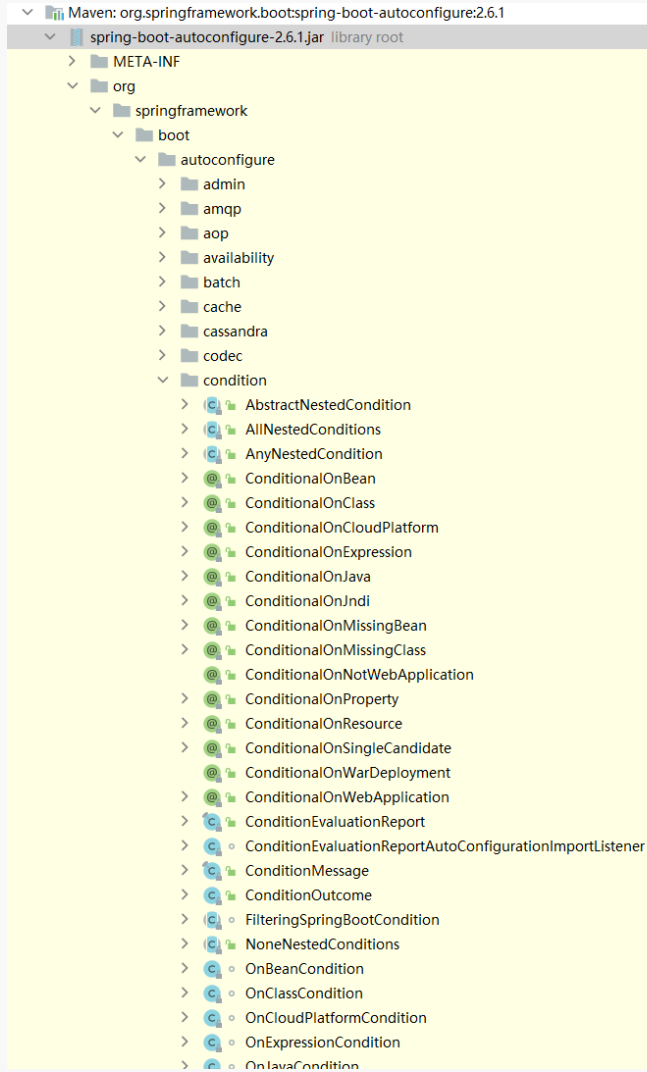
其中 ConditionalOnWebApplication 注解存在这些类型：

参数	说明
SERVLET	基于 servlet 的 web 应用程序。
REACTIVE	基于 reactive 的 web 应用程序。
ANY	上述两种都可以。

基于 reactive 的 web 应用程序是 Spring 5 新增的模块，名为 spring-webflux。它可以用少量线程来处理 request 和 response io 操作，这些线程统称为 Loop 线程。那些业务中阻塞操作，会提交给响应式框架的 work 线程进行处理，而那些不阻塞的操作则在 Loop 线程中进行处理。通过这种方式，可以大大提高 Loop 线程的利用率。

02、条件注解

所谓的条件注解就是应用程序的配置类的配置项，在满足某些特定条件后才会被自动启动和加载到IOC容器中。所有的条件注解都位于：**spring-boot-autoconfigure-2.6.1.jar**的**condition**包中，如下：



```
Maven: org.springframework.boot:spring-boot-autoconfigure:2.6.1
├── META-INF
└── org
    └── springframework
        └── boot
            └── autoconfigure
                ├── admin
                ├── amqp
                ├── aop
                ├── availability
                ├── batch
                ├── cache
                ├── cassandra
                ├── codec
                └── condition
                    ├── AbstractNestedCondition
                    ├── AllNestedConditions
                    ├── AnyNestedCondition
                    ├── ConditionalOnBean
                    ├── ConditionalOnClass
                    ├── ConditionalOnCloudPlatform
                    ├── ConditionalOnExpression
                    ├── ConditionalOnJava
                    ├── ConditionalOnJndi
                    ├── ConditionalOnMissingBean
                    ├── ConditionalOnMissingClass
                    ├── ConditionalOnNotWebApplication
                    ├── ConditionalOnProperty
                    ├── ConditionalOnResource
                    ├── ConditionalOnSingleCandidate
                    ├── ConditionalOnWarDeployment
                    ├── ConditionalOnWebApplication
                    ├── ConditionEvaluationReport
                    ├── ConditionEvaluationReportAutoConfigurationImportListener
                    ├── ConditionMessage
                    ├── ConditionOutcome
                    ├── FilteringSpringBootCondition
                    ├── NoneNestedConditions
                    ├── OnBeanCondition
                    ├── OnClassCondition
                    ├── OnCloudPlatformCondition
                    ├── OnExpressionCondition
                    └── OnJavaCondition
```