

从零开始造Spring

刘欣

微信: onlyliuxin97

微信公众号: 码农翻身(coderising)



思路

- 1. 读取XML文件

```
<context:component-scan base-  
package="org.litespring.service.v4,org.litespring.dao.v4">  
</context:component-scan>
```

- 2. 对指定的package 进行扫描(scan), 找到那些标记为
@Component 的类, **创建BeanDefinition**

- **2.1 把一个package下面的class 变成resource**

- 2.2 使用ASM读取Resource中的注解

- 2.3 创建BeanDefinition

- 3. 通过BeanDefinition创建Bean实例, 根据注解来注入

先来说说ASM

- ASM
 - 一个可以操作Java 字节码的框架
 - 可以读取/修改class中的字节码 （目前我们只关注读取）
 - 工作模式： Visitor
 - Class Reader
 - Class Visitor （使用ASM的用户提供）
 - Class Writer （暂不考虑）

ASM的Visitor模式

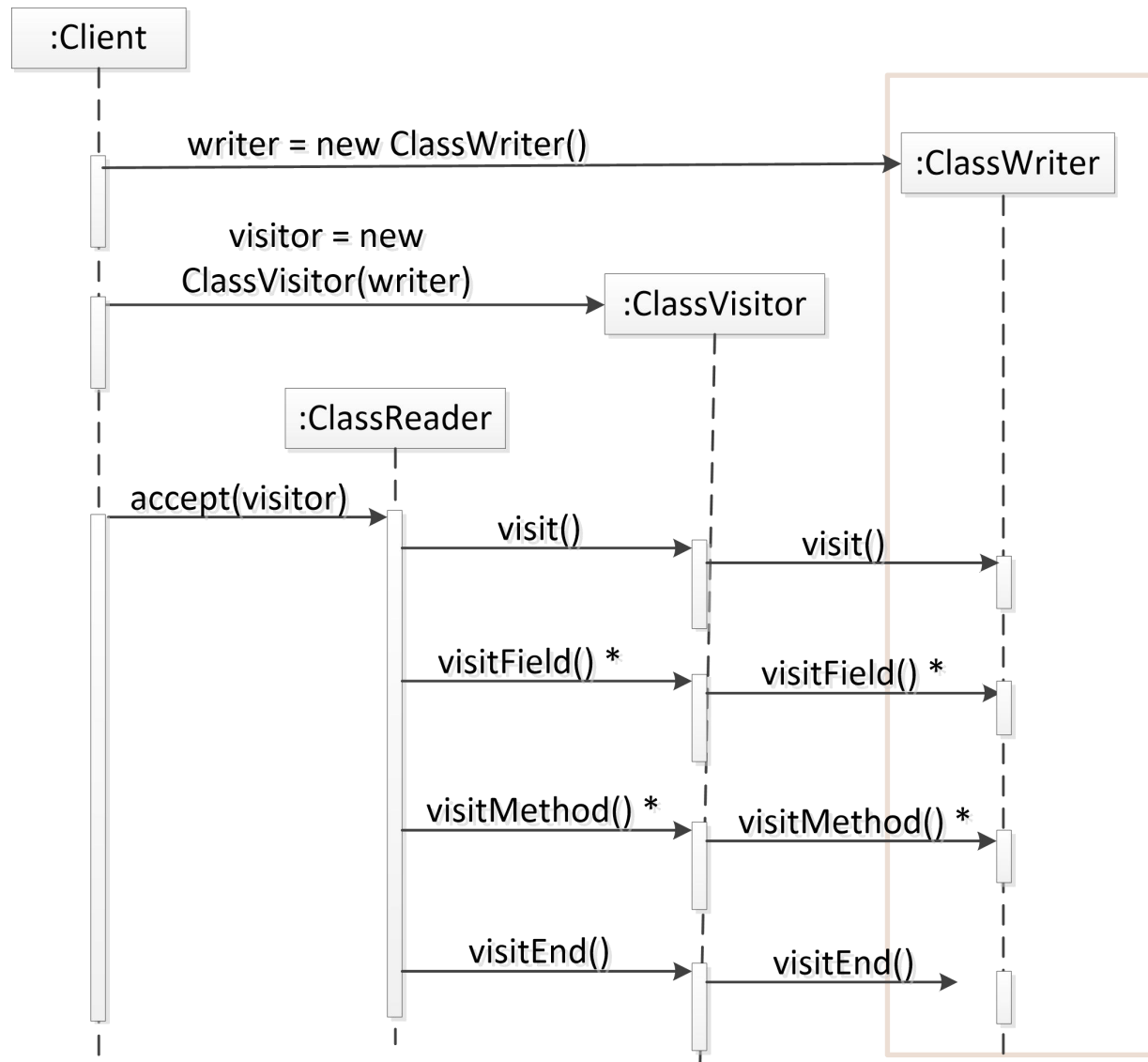
ClassReader : 我现在要解析一个复杂结构的类文件了啊, 每当我解析好一点东西, 我都会通知你来处理。

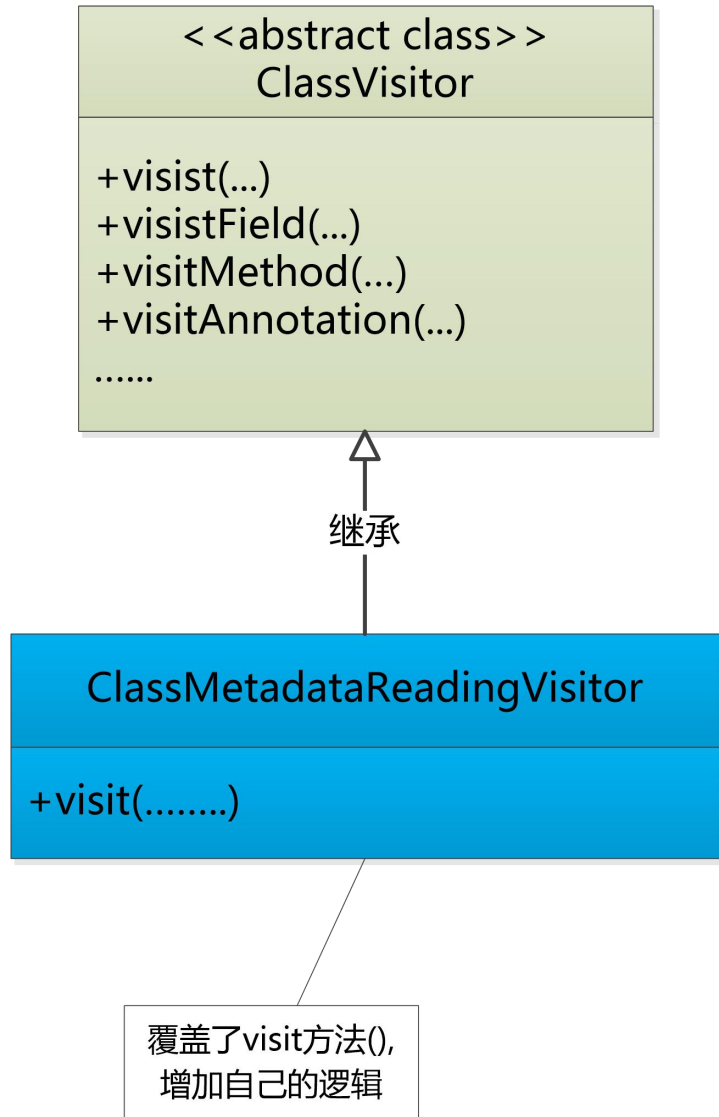
ClassVisistor : 你怎么通知我?

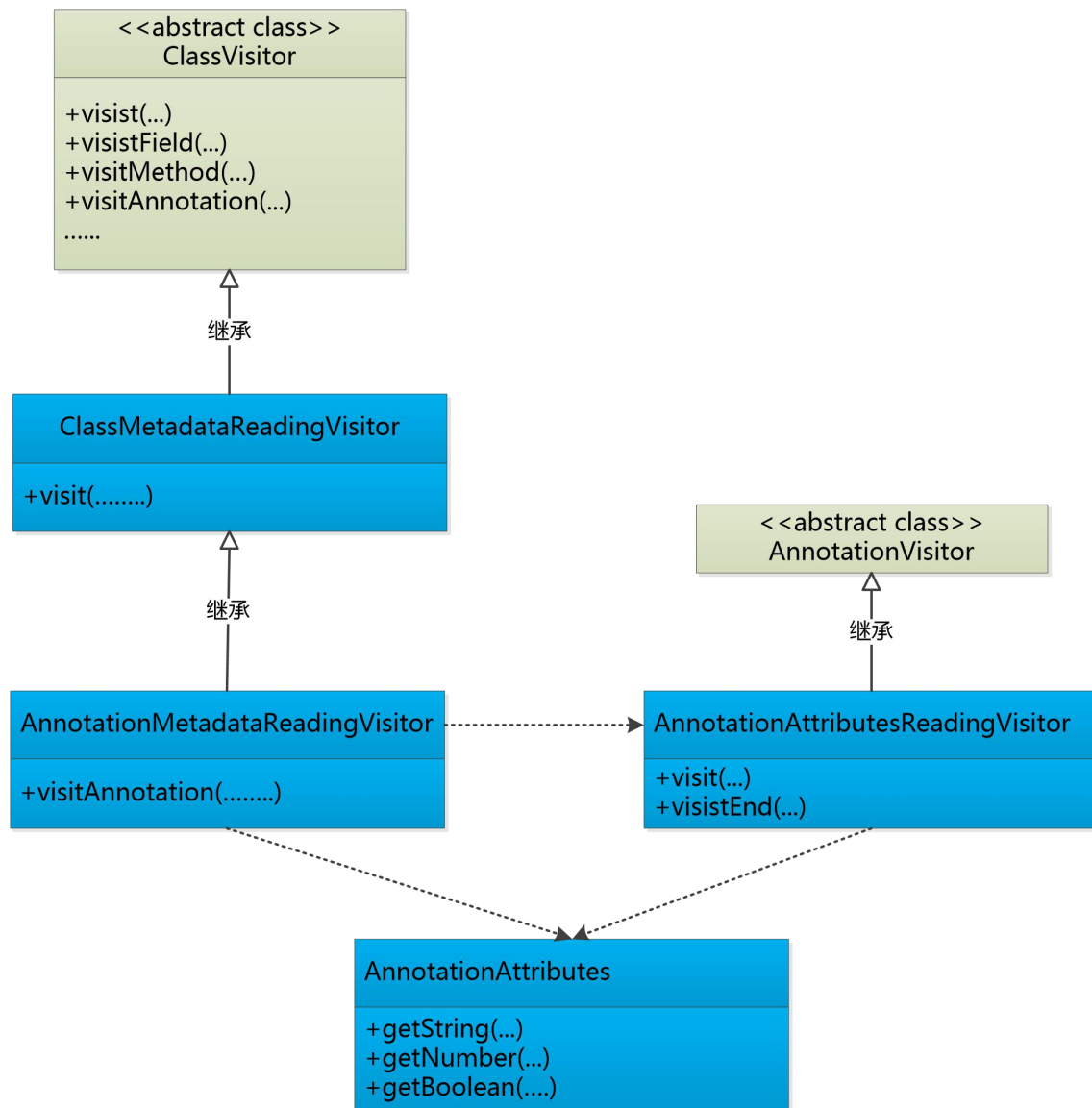
ClassReader : 当然是回调你的方法啊, 比如说当我开始解析一个方法时, 我就会回调你的visitMethod(), 把相关数据给你发过去, 你就可以处理了

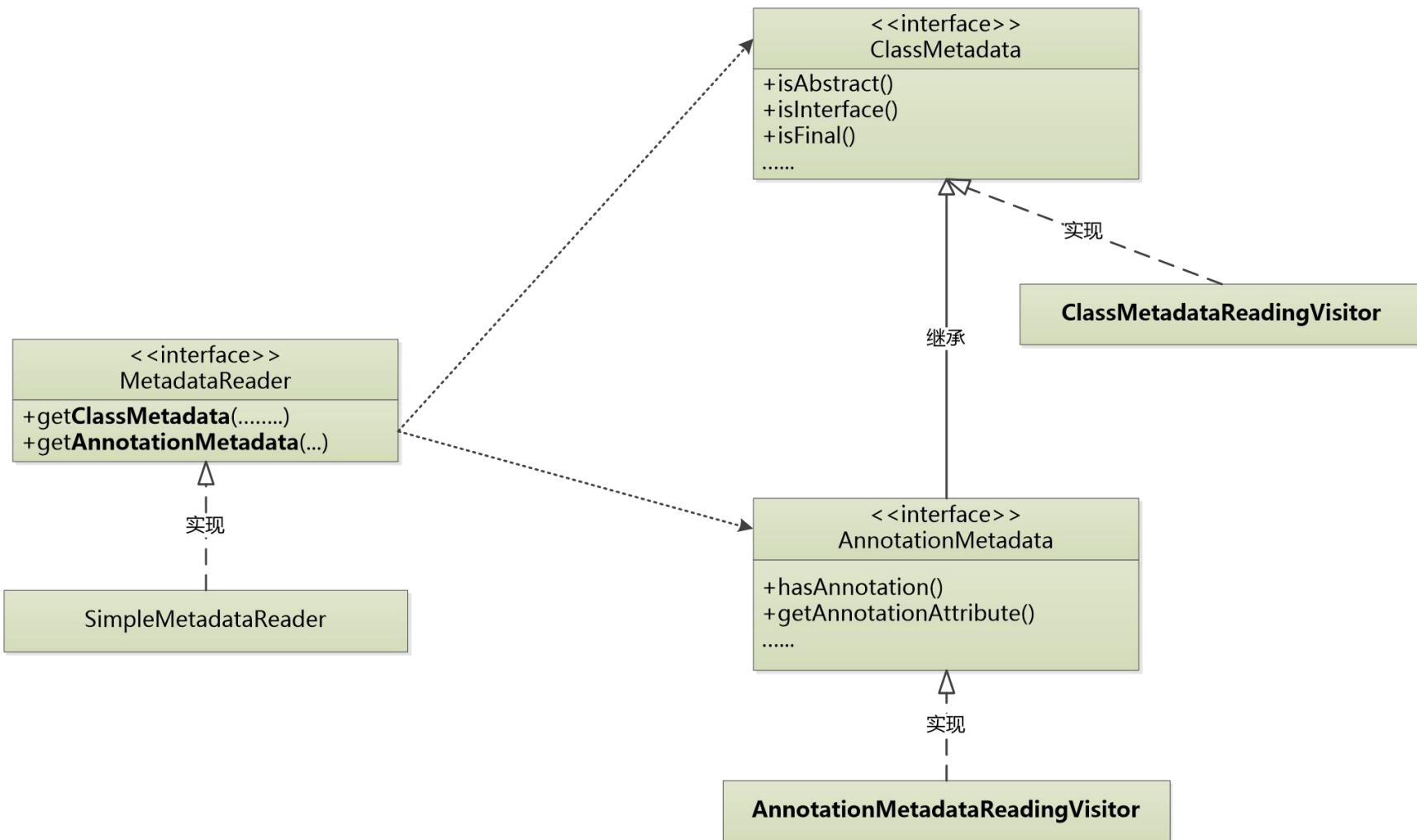
ClassVisitor : 明白! 但是当我处理完了, 比如修改了字节码, 怎么写回去呢?

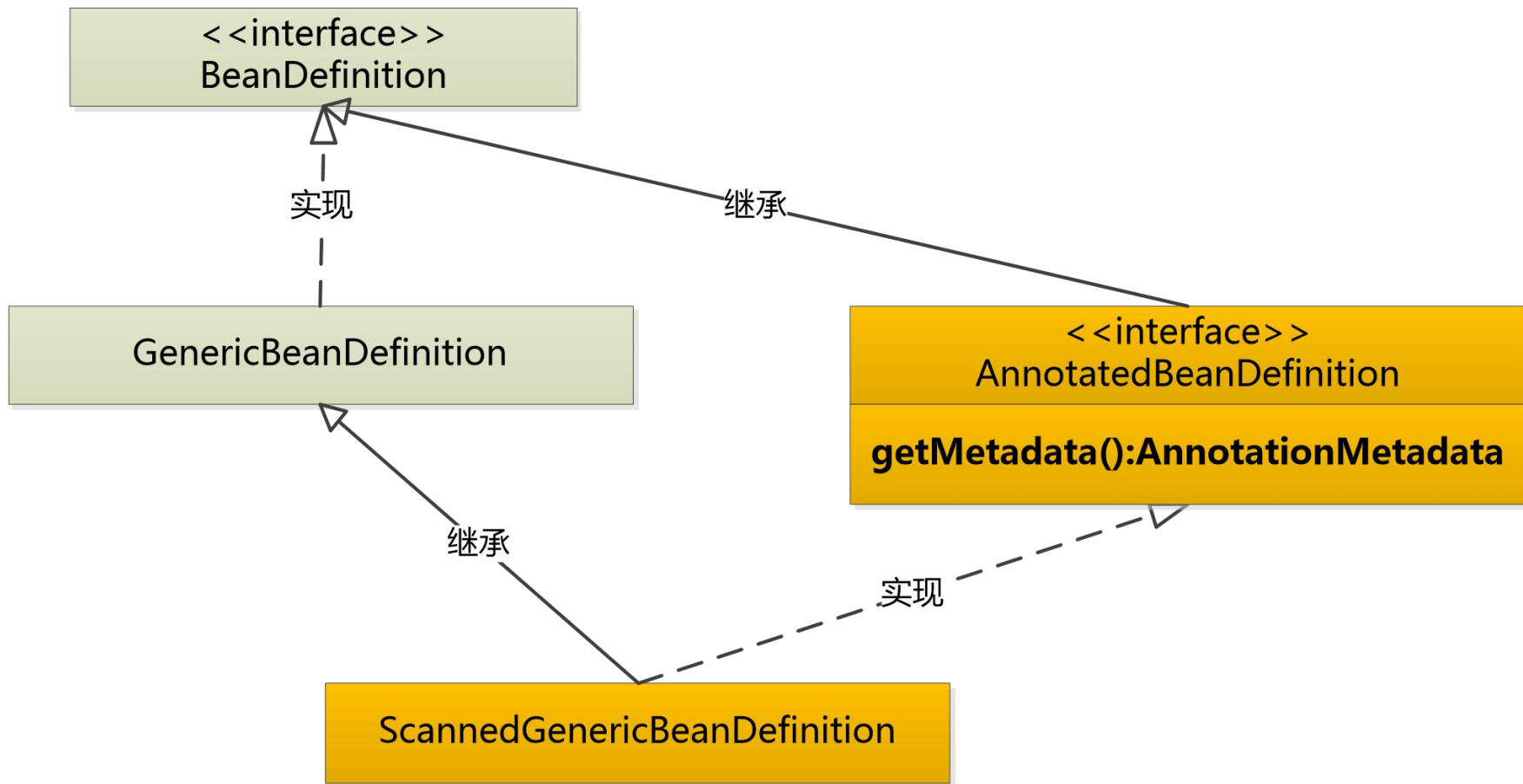
ClassReader : 你可以创建一个ClassWriter 对象, 每次处理完以后, 再调用同样的方法, 例如visitMethod(), 这样ClassWriter 就知道你的修改了, 它负责写回去











ClassPathBeanDefinitionScanner

- 1. 给定一个package的名称列表, 例如
org.litespring.service.v4,org.litespring.dao.v4"
- 2. 对指定的package 进行扫描(scan), 找到那些标记为
@Component 的类, 创建
ScannedGenericBeanDefinition, 并且注册到
BeanFactory中。

