**Dubbo源码阅读笔记**

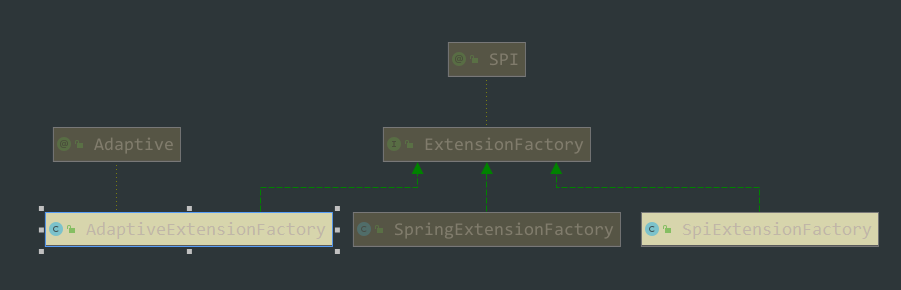
1. **ConcurrentMap<Class<?>, ExtensionLoader<?>> EXTENSION\_LOADERS**

**存放扩展loader，接口类型作为key.**

1. **objectFactory:获取扩展类的工厂对象**
2. **volatile Class<?> cachedAdaptiveClass：存放adaptive的类(已经compile过的class)**
3. **Map<String, Activate> cachedActivates：**
4. **Holder<Map<String, Class<?>>> cachedClasses： 存放的是，扩展文件中接口和实现类的对应关系.**

## ExtensionFactory

根据类别分为AdaptiveExtensionFactory和\*\*ExtensionFactory。其中\*\*ExtensionFactory代表的就是spi和spring两种实现(也就是两个工厂)



## @Adaptive

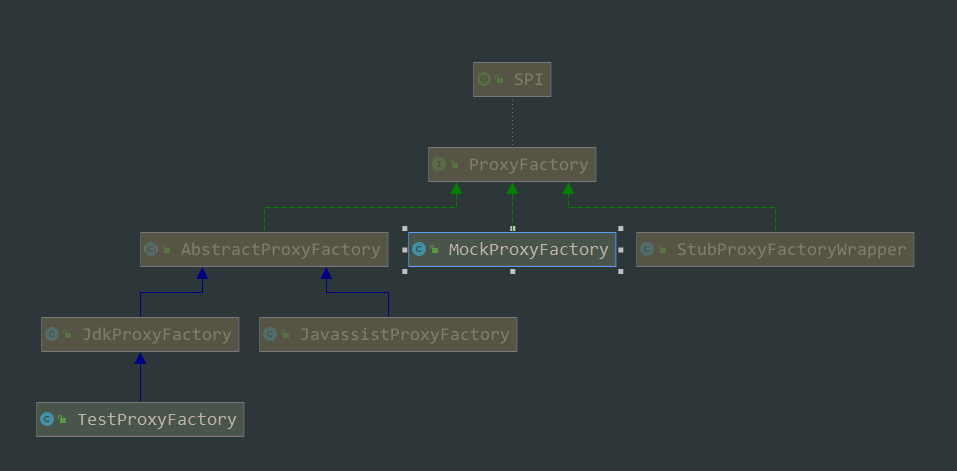
1. 该注解在类上

再类上，表示改类为接口的适配类

1. 该注解在方法上

表示该方法是适配方法(也就是实际调用子类的方法)

## ProxyFactory（默认实现为javassit）



**afterPropertiesSet**:bean被初始化结束之后（属性也都注入结束）触发。在这里主要是将registry\provider等属性赋值。

Wrapper：在dubbo中Wrapper是有妙用的，在getExtension-🡪>>createExtension中会先获取到wrapper。

服务暴露过程：

ServiceBean（onApplicationEvent）

-----🡪>>>