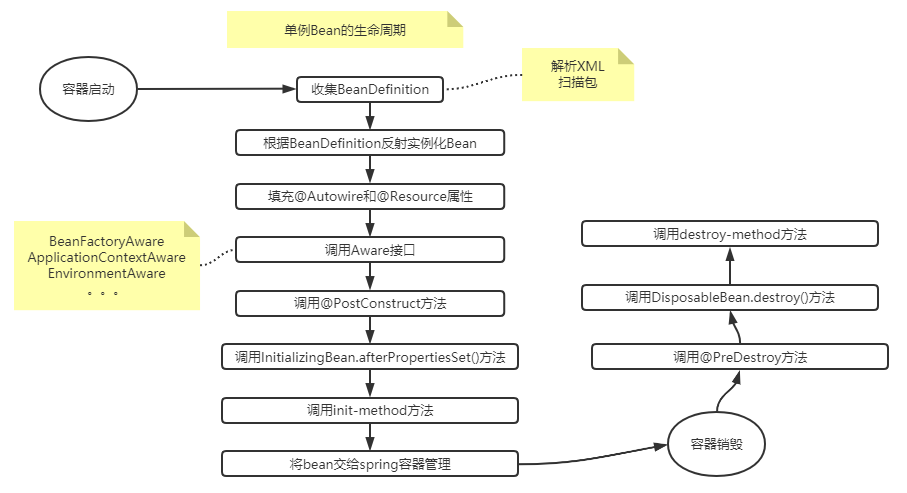
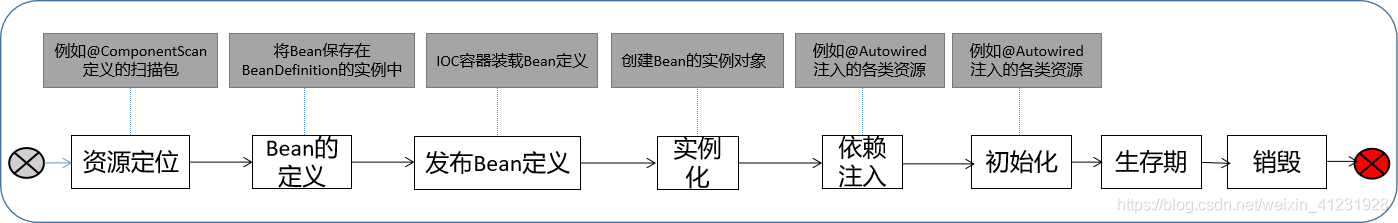
# Bean总结

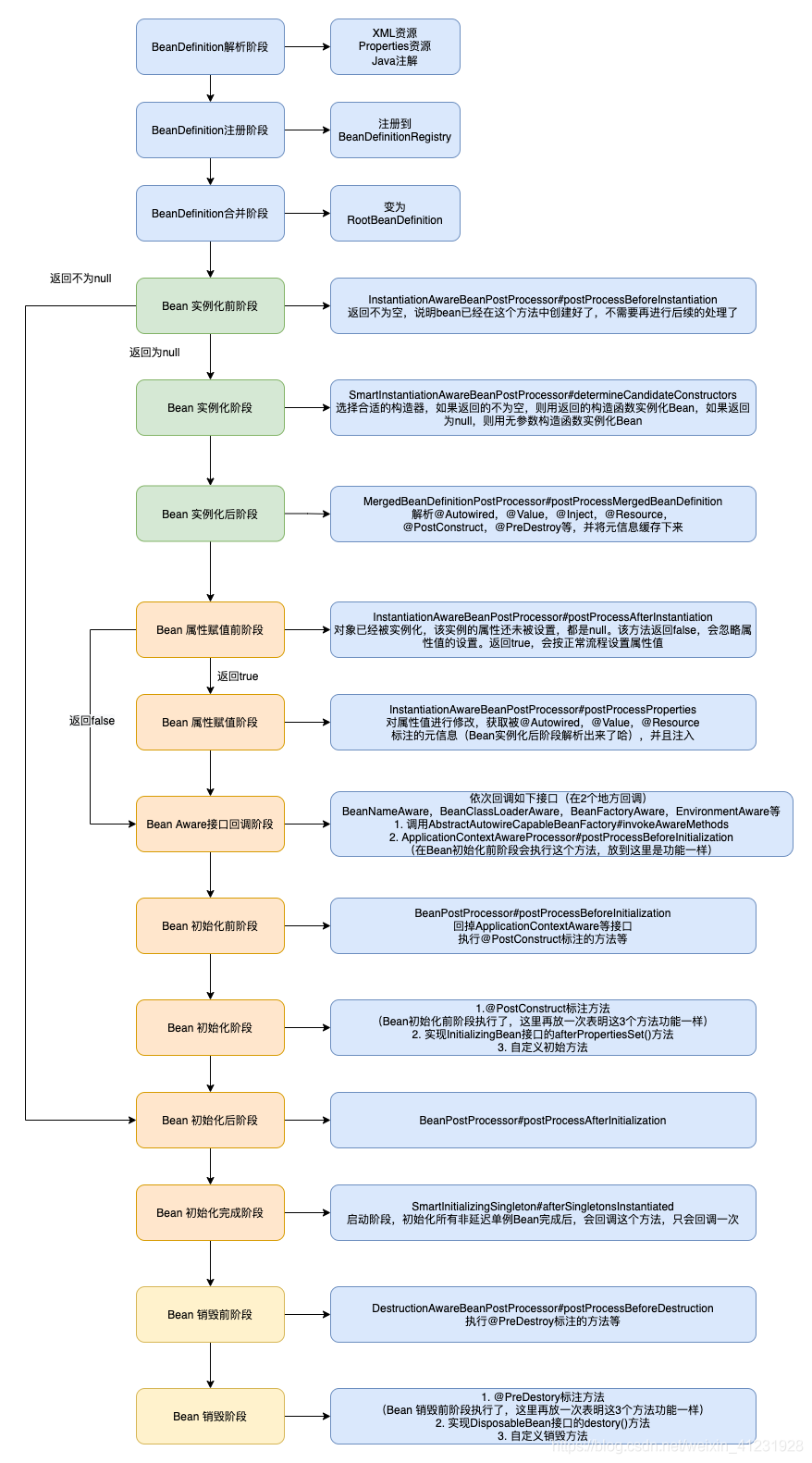


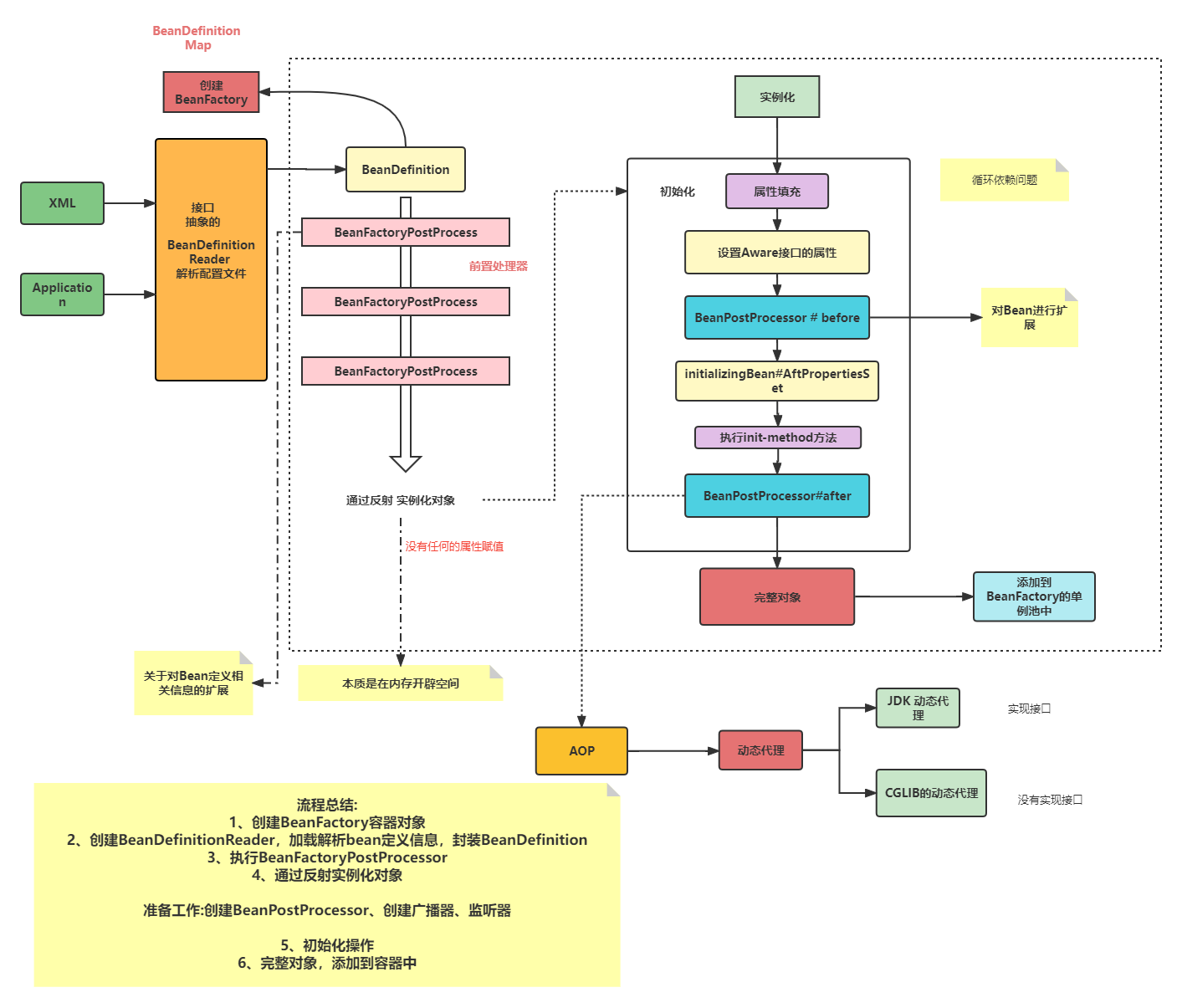
## **bean**[生命周期](https://so.csdn.net/so/search?q=%E7%94%9F%E5%91%BD%E5%91%A8%E6%9C%9F&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/a910247/article/details/_blank)

其定义为：****从对象的创建到销毁的过程。****而Spring中的一个Bean从开始到结束经历很多过程，但总体可以分为

****六个阶段：Bean定义、实例化、属性赋值、初始化、生存期、销毁****。







## spring初始化

首先需要说一样，JVM中的Bean的初始化和实例化还是不一样的，这里先记录spring中的初始化，其他的后面更新。

在spring中的初始化(initialization)之前要经过这两步骤

　　1)new 一个Bean这个 时候Bean里面的属性是空的

　　2)给Bean的属性赋值

　　3)将创建的bean放到Bean工厂（BeanFactory）中

在spring中实力化(instantiation)指的是，从容器中得到一个Bean。

总结： spring中的初始化就是创建完对象，付完值往容器中添加对象的过程，实例化就是从容器中获取对象的过程getBean()的过程。

## 小结

\* 在Spring容器创建完成时，会自动调用单实例bean的构造方法，对单实例bean进行了实例化操作。

\* \* 多实例的bean在容器关闭的时候是不进行销毁的，也就是说你每次获取时，IOC容器帮你创建出对象交还给你，

\* 至于要什么时候销毁这是你自己的事，Spring容器压根就不会再管理这些多实例的bean了。

\* \* 单实例：在容器启动的时候创建对象；

\* 多实例：在每次获取的时候创建对象；

\* \* 初始化：

\* 对象创建好，并赋值好，调用初始化方法；

\* \* 销毁：

\* 单实例：容器关闭的时候；

\* 多实例：容器不会管理Bean，只是负责创建，容器不会调用销毁方法；

\* \* 1、指定初始化和销毁的方法；

\* 2、通过Bean 实现InitializingBean和DisposableBean

\* InitializingBean(定义初始化逻辑)

\* DisposableBean（定义销毁逻辑）