

Advanced Bean Configuration



Bryan Hansen

DIRECTOR OF SOFTWARE DEVELOPMENT

@bh5k

Advanced Bean Configuration

BeanPostProcessor

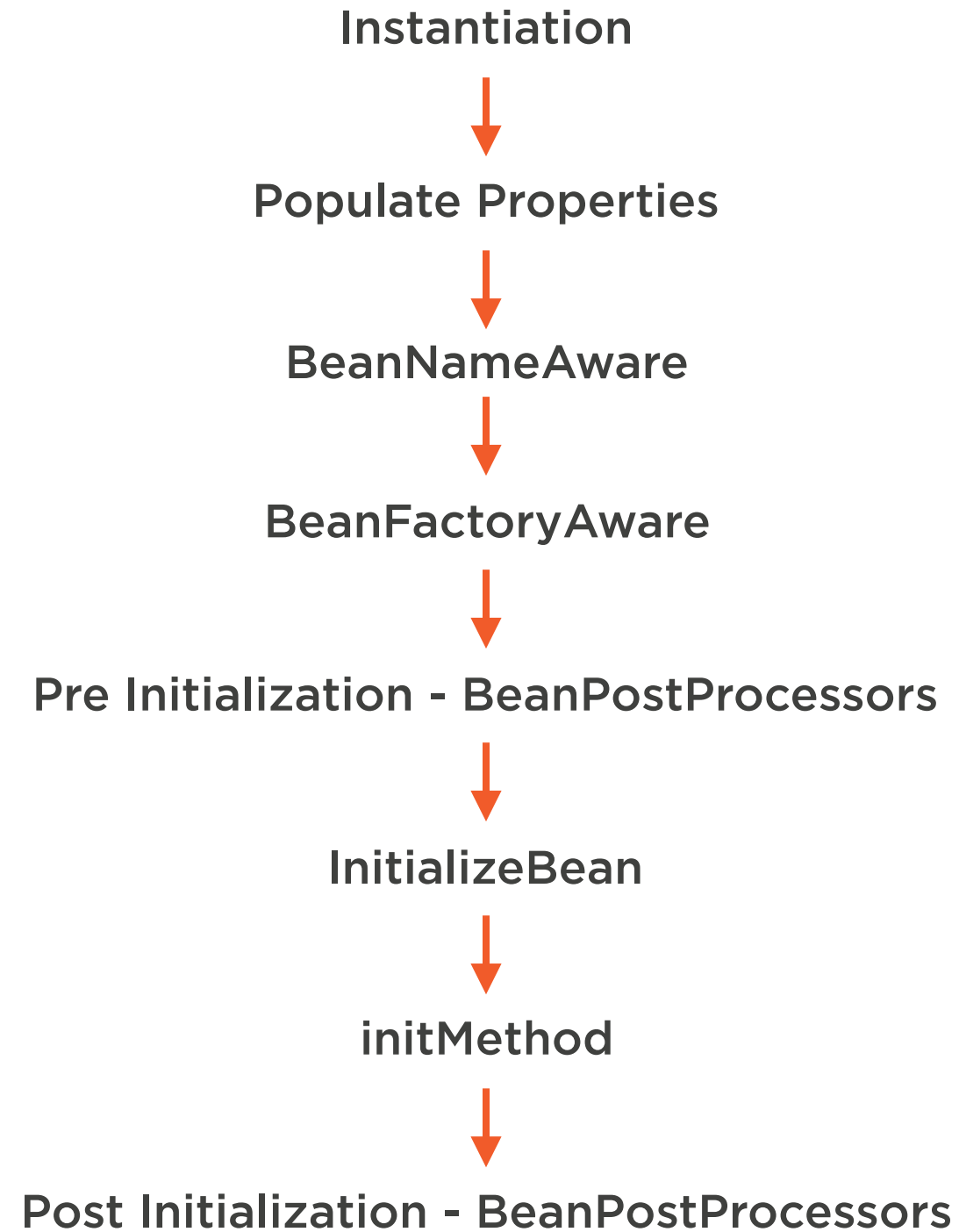
FactoryBean

SpEL

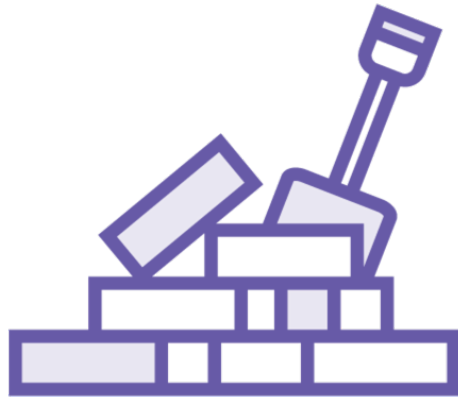
Proxies

Profiles

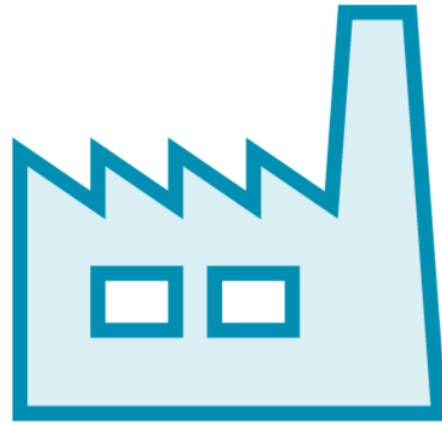
Bean Lifecycle



FactoryBean



**Builds on
initMethod concept**



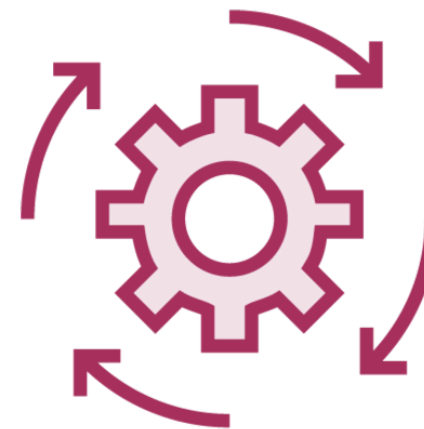
**Factory
Method Pattern**



Legacy Code



**Contract without
Constructor**



Static Methods

```
@Value("#{ T(java.lang.Math).random() * 100.0 }")  
private double seedNum;
```

SpEL

Manipulate Object Graph

Evaluate at Runtime

Configuration

Spring AOP Proxies

```
ProxyFactory factory = new ProxyFactory(new SimplePojo());  
factory.addInterface(Pojo.class);  
factory.addAdvice(new RetryAdvice());  
factory.setExposeProxy(true);
```

```
Pojo pojo = (Pojo) factory.getProxy();
```

```
// this is a method call on the proxy!  
pojo.foo();
```

Spring Utilizes Proxies

When?

@Transactional

```
@Repository("speakerRepository")
@Profile("dev")
public class HibernateSpeakerRepositoryImpl implements
    SpeakerRepository
```

Bean Profiles

Adapt Environments

Runtime Configuration

Summary

Bean Lifecycle

FactoryBean

SpEL

Proxies

Profiles

Next