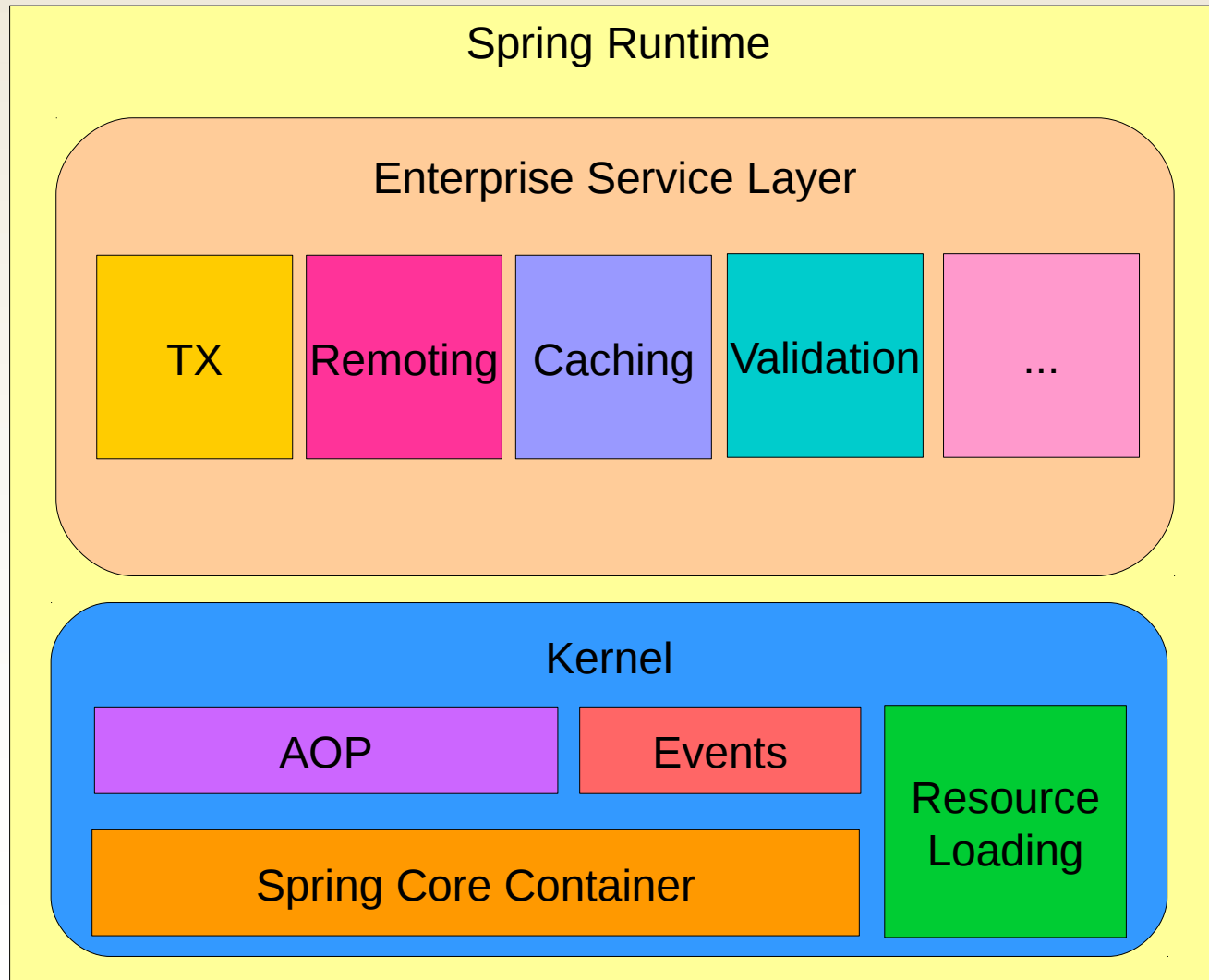


Spring ApplicationContext Bootstrap Süreci



Spring Runtime Mimarisi



Spring Bootstrap Süreci

- İlk adımda **ApplicationContext'in yaratılması** gerekir
 - Konfigürasyon metadata'nın yüklenerek ApplicationContext nesnesinin oluşturulması gerekir
 - ApplicationContext programatik (standalone) veya deklaratif (web) biçimde yaratılabilir
 - Konfigürasyon metadata farklı formatlarda (xml, java, annotation) olabilir

Spring Bootstrap Süreci

- Daha sonraki adımda **uygulamanın bileşen konfigürasyonu** tespit edilir
- Bu aşamada konfigürasyon metadata'daki bean tanımları, runtime'da **BeanDefinition** nesnelere dönüştürülür
- BeanDefinition nesneleri **bileşenlerin özellikleri, hangi bağımlılıklara ihtiyaç duyduğu** gibi bilgileri içerir

Spring Bootstrap Süreci

- Bir sonraki adım **bileşenlerin** (bean instance'ların) **yaratılmasıdır**
- Bu aşama **iki fazda** gerçekleşir
- Önce bazı **özel altyapısal bean'ler** yaratılır (BeanFactoryPostProcessor)
- Bu özel bean'lar diğer **BeanDefinition'ların üzerinde özelleştirmeler/değişiklikler** yapabilirler

Spring Bootstrap Süreci

- Daha sonra ise **başka bazı altyapısal bean'ler** (BeanPostProcessor) yaratılır
- Bu özel bean'ler ise yaratılacak olan **uygulama bean instance'larının üzerinde özelleştirme veya değişiklik yapacaklardır**
- Bu aşamada artık uygulamanın **normal bean instance'ları** yaratılmaya başlanır

Spring Bootstrap Süreci

- Her bean yaratılıp **bağımlılıkları enjekte** edilir
- Varsa **init metotları çağrılmadan önce BeanPostProcessor** tarafından özelleştirmeye tabi tutulabilir
- Ardından **init metotları çağrılır**
- Sonra **yine BeanPostProcessor** tarafından özelleştirilebilir
- Bu aşamanın ardından artık bean **kullanıma hazır** hale gelmiştir

İletişim

- **Harezmi** Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- <http://www.java-egitimleri.com>
- info@java-egitimleri.com

