

# Web Uygulamalarında ApplicationContext Konfigürasyonu

# Spring ve Web Uygulamaları

- Web uygulamalarında Spring Container genellikle **web.xml'de deklaratif biçimde** tanımlanarak yaratılır
- Spring **herhangi bir Web UI teknolojisi** ile birlikte kullanılabilir
- Web uygulamalarındaki Spring Container'ın tipi **WebApplicationContext'tir**
- **WebApplicationContext** **ApplicationContext'i extend** eder

# Web Uygulamalarında ApplicationContext Oluşturma

- Web ortamı için **ilave kabiliyetler** sunar
- `WebApplicationContext`  
**ContextLoaderListener** ile bootstrap sırasında yaratılır
- Yaratılan `WebApplicationContext` daha sonra **ServletContext** üzerinden erişilebilir
- Örneğin bir Servlet içerisinde `ApplicationContext` **bean lookup** ile istenilen bean alınıp kullanılabilir

# Web Uygulamalarında ApplicationContext Oluşturma

web.xml

```
<web-app>
  <context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>
      classpath*: /appcontext/beans-*.xml
    </param-value>
  </context-param>
```

- Default: /WEB-INF/applicationContext.xml
- Eğer parametre mevcut ise birden fazla dosya path'i belirtilebilir
- Path'ler virgül, noktalı virgül veya boşluk ile ayrılabilir
- Ant-style path pattern'ları da desteklenir

```
<listener>
  <listener-class>
    org.springframework.web.context.ContextLoaderListener
  </listener-class>
</listener>
</web-app>
```

# Web Uygulamalarında Java Tabanlı Container Konfigürasyonu

- **ContextLoaderListener** default olarak **WebApplicationContext** sınıfından bir Container oluşturur, bu da XML konfigürasyon dosyaları ile çalışır
- Java tabanlı konfigürasyon sınıflarının yüklenmesi isteniyorsa **contextClass** isimli bir context-param ile **AnnotationConfigWebApplicationContext**'i kullanması söylenmelidir
- **contextConfigLocation** context-param'a ise bundan böyle değer olarak FQN şeklinde konfigürasyon sınıfları verilebilir

# Web Uygulamalarında Java Tabanlı Container Konfigürasyonu

```
<web-app>
```

```
  <context-param>
```

```
    <param-name>contextClass</param-name>
```

```
    <param-value>
```

```
org.springframework.web.context.support.AnnotationConfigWebApplicationContext
```

```
  </param-value>
```

```
  </context-param>
```

```
  <context-param>
```

```
    <param-name>contextConfigLocation</param-name>
```

```
    <param-value>
```

```
      com.javaegitimleri.petclinic.config.AppContextConfig
```

```
    </param-value>
```

```
  </context-param>
```

```
  <listener>
```

```
    <listener-class>
```

```
org.springframework.web.context.ContextLoaderListener
```

```
    </listener-class>
```

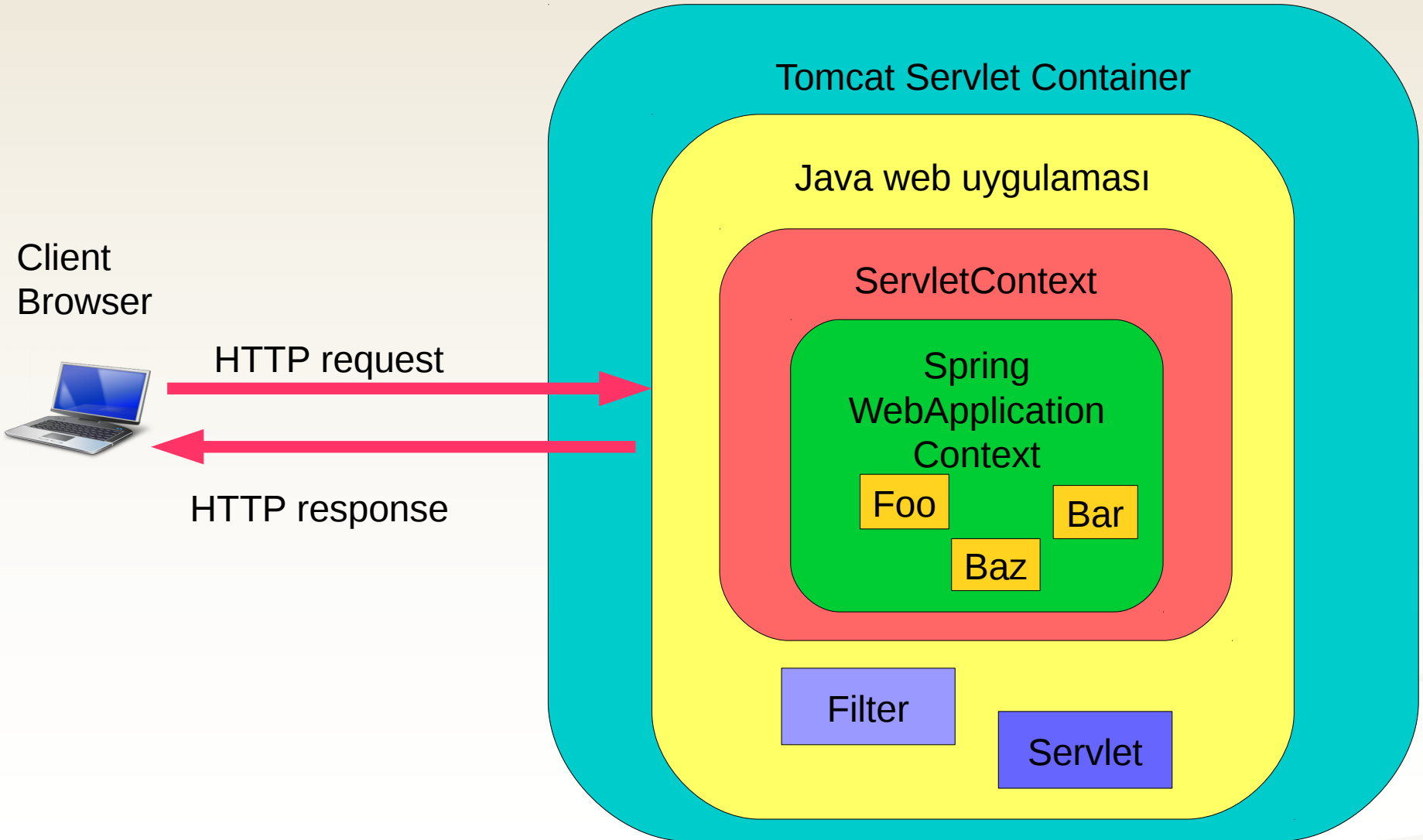
```
  </listener>
```

```
</web-app>
```

# WebApplicationContext'e Nasıl Erişilir?

- ServletContext'e bind edilen WebApplicationContext'e erişmek için Spring bir **utility sınıf** sunar
- **WebApplicationContextUtils** sınıfı ile Servlet vb nesneler içerisinde WebApplicationContext'e erişilebilir
  - `getRequiredWebApplicationContext(servletContext)`
  - `getWebApplicationContext(servletContext)`

# ServletContext & WebApplicationContext İlişkisi





# İletişim

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

