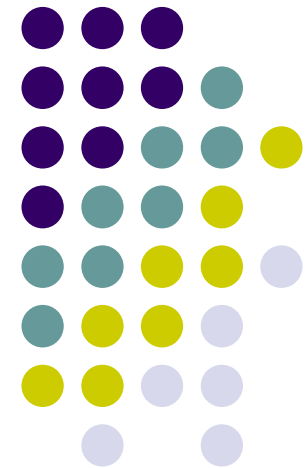


# JSF ve JBoss Seam

---

Melih Sakarya

[www.mergecons.com](http://www.mergecons.com)



# Java' da Web Mimarileri

- Servlet
- JSP
- Struts
- ...
- JSF



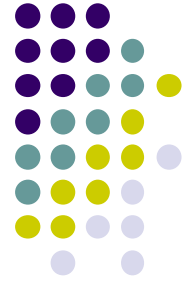
# JSF



- Açılımı Java Server Faces
- MVC (Model-View-Controller)
- JSR Standartı (JSR-127, JSR 252)
- Component Oriented
- Event Driven
- Farklı JSF implementasyonları
- 3. parti JSF bileşenleri (PrimeFaces, Ice Faces, RichFaces)

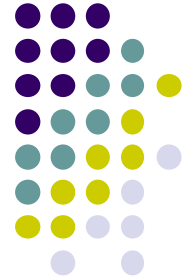
# JSF' in ortaya çıkışı

- 2001 ortaları
  - Amy Fowler
- 2002 yılında
  - Ed Burns
  - Craig McClanahan

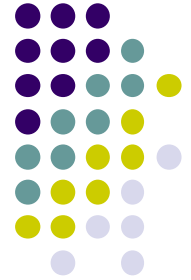


# Specification

- JSR127 – JSF 1.1
- JSR252 – JSF 1.2



# Gerçeklemeler (Implementation)



- MyFaces (Apache)
- Sun RI (Reference Implementation)
- ...



# Neden JSF

- Standart olması.
- RAD desteđi.
- Bileşen mimarisinin kolaylığı.
- Farklı grupların bileşen desteđi.
- Gelişmiş EL (Expression Language desteđi).
- Karmaşık uygulamaları daha kolay geliştirme.

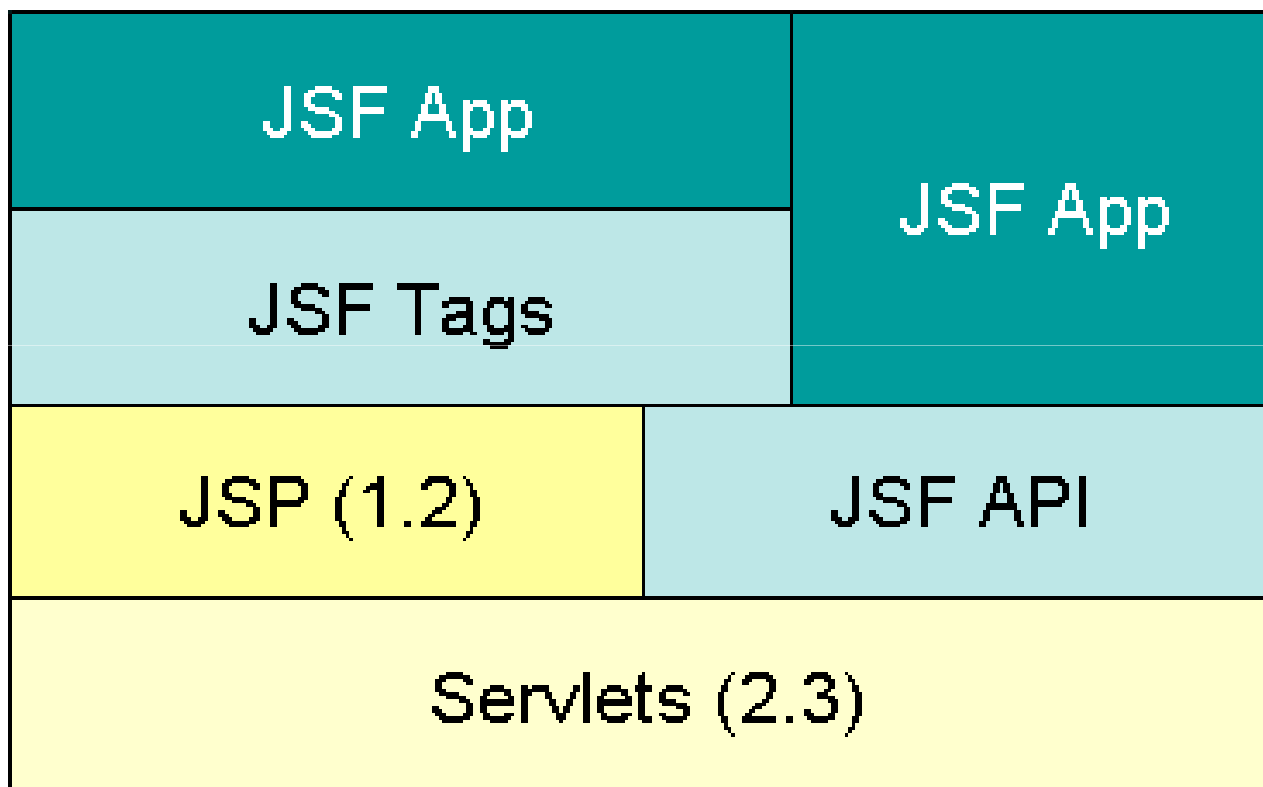
# JSF Yaklaşımı



- Bileşen yönelimli
- Olay(event) tabanlı
- Bileşenlere olay(event) tanımlama
- Bileşenleri Java objelerine erişim (EL desteği ile)

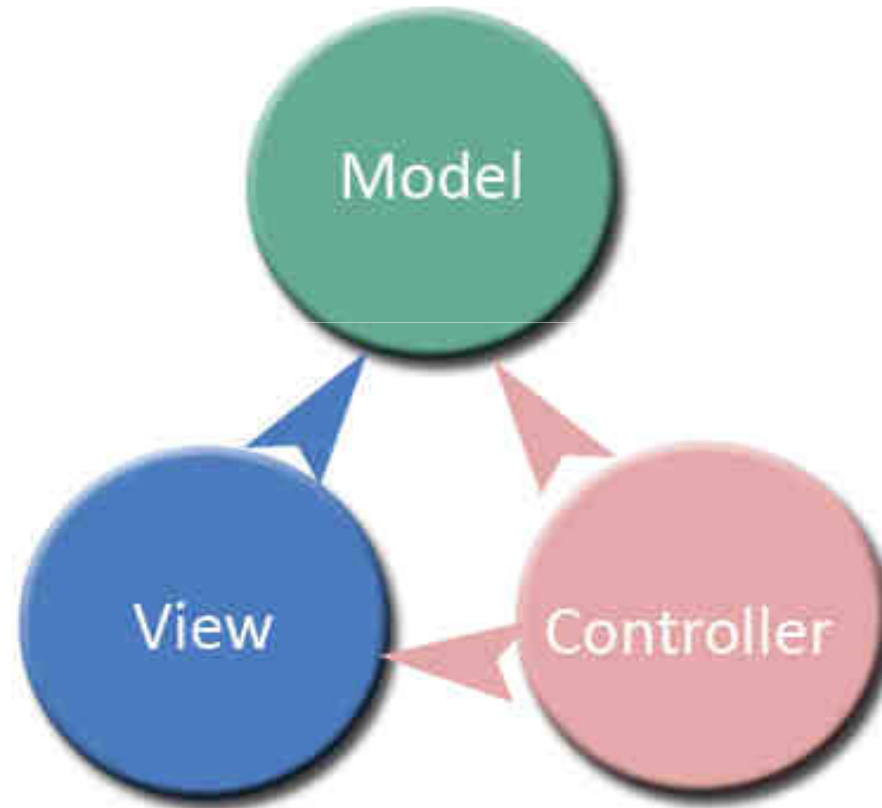


# JSF Mimarisi

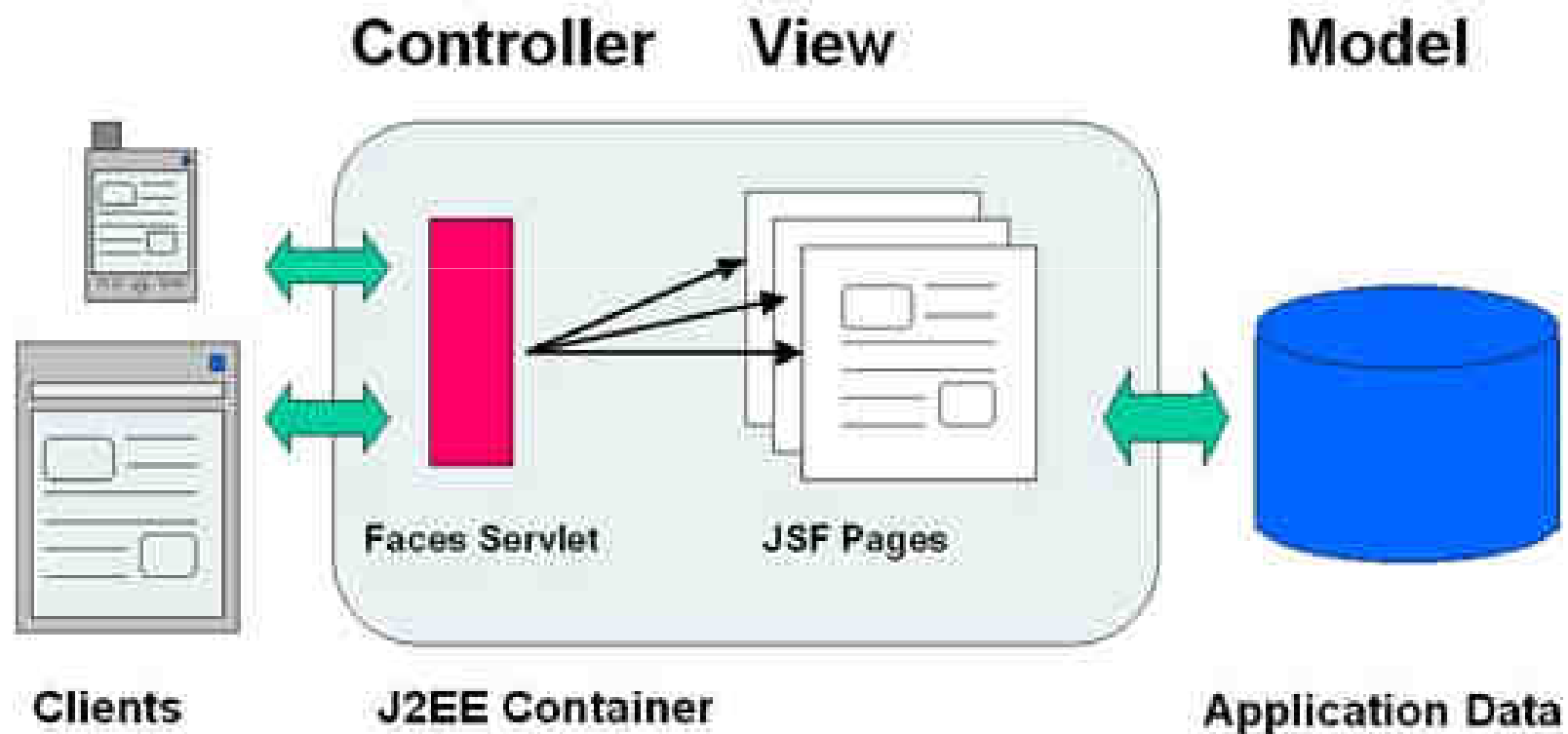


# MVC

## Model View Controller



# JSF-MVC





# JSF Ortamı

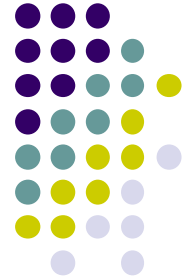
- web.xml konfigurasyonu
- faces-config.xml
- Core JSF kütüphaneleri
- 3. party JSF kütüphaneleri



# web.xml

- Java' da web ortamı ile ilgili tanımların bulunduğu xml dosyasıdır.
- JSF projelerinde JSF tanımları yapılmalıdır.
- WEB-INF altında bulunur.

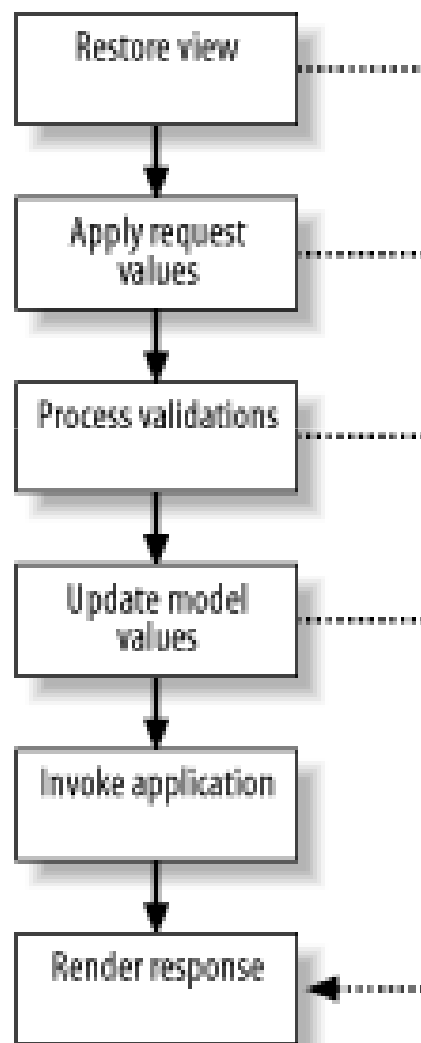
# faces-config.xml



- JSF ile ilgili tanımların bulunduğu dosyadır.
- WEB-INF altında bulunur.



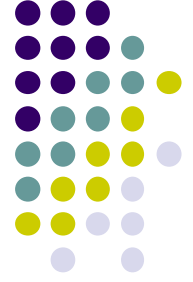
# JSF Yaşam Döngüsü





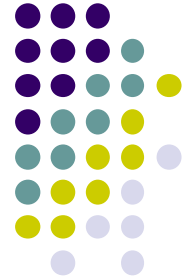
# JSF Arayüz Bileşenleri

- core
  - Validation, conversion, event handling ...
  - 18 standart core tag
- html
  - presentation (sunum)
  - 25 standart html tag



# JSF HTML Bileşenleri

- Çıktı veren bileşenlerdir.
- İki JSF standart bileşen kütüphanesinden biridir.
- Standart olarak h: ile başlarlar
- Sayfa başında taglib tanımı olmalıdır.



# HTML bileşenleri

- `h:inputText`
- `h:outputText`
- `h:commandButton`
- `h:dataTable`
- `h:column`
- ...



# Core Bileşenler

- Core bileşenlerdir.
- İki JSF standart bileşen kütüphanesinden biridir.
- Standart olarak f: ile başlarlar
- Sayfa başında taglib tanımı olmalıdır.

# Core Bileşenler

- f:view
- f:convertDateTime
- f:selectItem
- f:validateLength
- ...



# Bir JSF Sayfası



```
<%@ taglib uri="http://java.sun.com/jsf/html" prefix="h" %>
<%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %>
<html>
  <head>
    <title></title>
  </head>
  <body>
    <f:view>
      <h:outputText value="Merhaba Dünya" />
    </f:view>
  </body>
</html>
```

# Geliştirme Ortamı Kurulumu



- Java 1.6
- Eclipse
- JBoss Tools
- JBoss AS
- MySql
- MySql JDBC Driver
- MySql workbench

# HTML Çıktısı



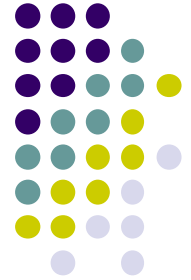
```
<html>
  <head>
    <title></title>
  </head>
  <body>
    Merhaba Dünya
  </body>
</html>
```

# f:view

- Root bileşendir.
- JSF bileşen ağacı bu bileşen ile başlar.



# EL (Expression Language)

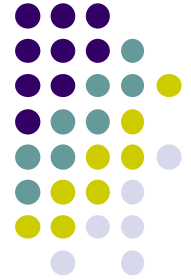


- Özel bir tanım dilidir.
- JSTL ile benzerlik gösterir.
- `{managedBeanAdi.attribute}` şeklinde yazılır.
- Genişletilmiş (Extended) EL seçenekleri bulunabilir.



# ManagedBean

- Basit Java sınıflarıdır.
- Herhangi bir sınıfı miras almasına gerek yoktur.
- JSF tarafında otomatik oluşturulurlar.
- JSF context' i içerisinde erişilebilirler
- faces-config.xml içerisinde tanımlanmalıdırlar



# ManagedBean

- JSF ManagedBean Scope
  - Session
  - Application
  - Request
- EL (Expression Language) ile erişilebilirler.

# Örnek Managed Bean



```
public class OrnekManagedBean {  
    private String ad;  
  
    public String getAd() {  
        return ad;  
    }  
  
    public void setAd(String ad) {  
        this.ad = ad;  
    }  
}
```

# faces-config.xml tanımı



```
<managed-bean>  
  <managed-bean-name>ornek</managed-bean-name>  
  <managed-bean-class>  
    com.msakarya.OrnekManagedBean  
  </managed-bean-class>  
  <managed-bean-scope>session</managed-bean-scope>  
</managed-bean>
```



# Facelet

- JSF için JSP ye alternatif bir yapıdır.
- Tamamen xml tabanlıdır ve xhtml olarak yazılır.
- Xml standartlarına uyulmak zorundadır.
- Gelişmiş facelet tagleri bulunur.
- f:view a ihtiyaç yoktur.
- .xhtml dosyaları olarak yazılır.

# xhtml



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
  Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
  transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  xmlns:h="http://java.sun.com/jsf/html"
  xmlns:f="http://java.sun.com/jsf/core"
  xmlns:ui="http://java.sun.com/jsf/facelets">
  <head>
    <title>Sayfa Basligi</title>
  </head>
  <body>
    icerik
  </body>
</html>
```

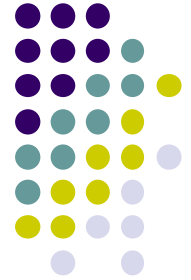
# Seam Framework



- JSF
- JPA
- EJB 3.0
- Ajax
- RichFaces
- A4J
- Drools
- iText PDF
- JFreeChart
- Facelet
- Security
- SSL
- Web Services
- Pojo
- IoC
- JBPM
- Hibernate
- Groovy
- Javascript
- Hibernate
- Localization
- Asynchronous
- Remoting
- Testing
- GWT
- Hibernate Search
- Mail
- Captcha
- JBoss Cache

# Seam Framework

- Red Hat
- Exadel
- JBoss
- Hibernate
- Gavin King (Proje lideri)





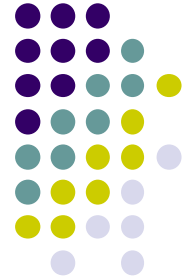
# Neden Seam

- Tümüleşik bir yapı
- Hızlı ve kolay geliştirme ortamı (JBoss Tools)
- Açık kaynak
- Ücretsiz
- Kolay entegrasyon (Spring, IceFaces ...)
- Ücretsiz geliştirme araçları (JBoss Tools)
- Kolay test edilebilirlik
- Farkli ortamlarda deployment desteği

# Geliştirme ortamı

- Eclipse
- JBoss Tools
- Java 1.5 ve üzeri
- J2EE 1.5 ve üzeri



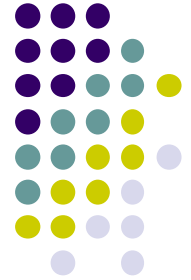


# Desteklediği Ortamlar

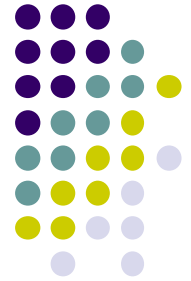
- JBoss AS
- Websphere AS
- OC4J
- Weblogic
- Tomcat
- ...

# Seam-gen

- Bizim için hazırlanan proje ortamı
- Code generation
- DB oriented code generation

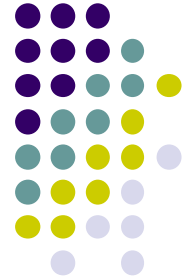


# Seam User Interface Seçenekleri



- Facelet
- JSF RI
- Richfaces
- Ice Faces
- My Faces
- ...

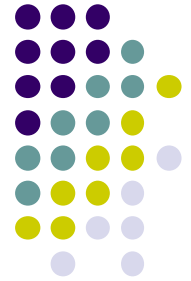
# JSF-Facelet ve Seam



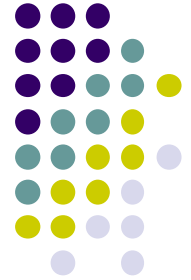
- Template desteđi
- Kolay ve yetenekli EL (Expression Language)
- RichFaces
- Theme desteđi

# Seam Component

@name("ornek")



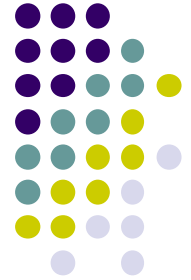
# Seam Context



- Event context
- Stateless context
- Session context
- Conversation context
- Application context
- Page context
- Business Process context

# EJB 3.0

- Stateful
- Stateless
- Message Driven Bean





# EJB tanımı

```
@Name("loginAction")
```

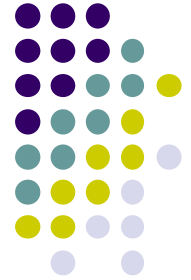
```
@Stateless
```

```
public class LoginAction implements Login {
```

```
    ...
```

```
}
```

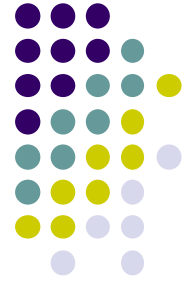
# JPA



- Annotation ile deklarasyon
  - @Entity
  - @Id
  - ...

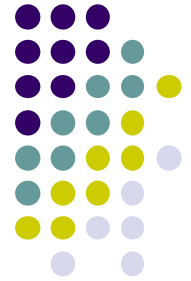
# CRUD Operasyonları

- Insert
- Update
- Delete
- Select



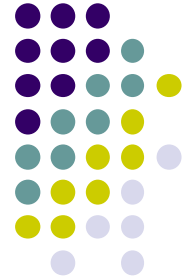
# IoC (Inversion of Control)

- Injection
- Outjection
- Bijection



# EJB 3.0

- Stateful
- Stateless
- Message Driven Bean



# Son



# Sorular ?????