

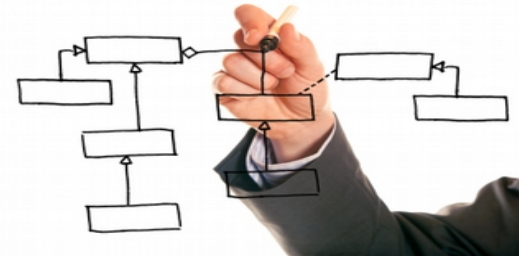
Spring Ekosisteminde Kurumsal Yazılım Geliřtirme

Kenan Sevindik
Harezmi Biliřim özmleri A.ř.



Hakkımızda

- Harezmi Bilişim Çözümleri 2011 yılında kuruldu
- Uzmanlık alanlarımız
 - Kurumsal uygulama geliştirme
 - Java teknolojileri
- Hizmetlerimiz
 - Kurumsal uygulama geliştirme faaliyetleri yürütüyoruz
 - Kurumsal Java eğitimleri düzenliyoruz
 - Danışmanlık ve koçluk hizmetleri veriyoruz



Çözümlerimiz

- Speedy Framework
 - Kurumsal Java web uygulamaları için altyapısal servisler sunar
- Harezmi E-İmza
 - Kamu SM ile uyumlu elektronik imza servisleri sunar

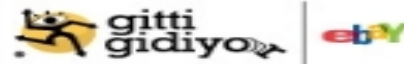


Eğitimlerimiz

- Spring Application Framework Eğitimi
- Spring Security Framework Eğitimi
- Spring Web Servis Framework Eğitimi
- Hibernate Persistence Framework Eğitimi
- Vaadin UI Framework Eğitimi
- Java Programlama Dili Eğitimi
- Nesne Yönelimli Tasarım Prensipleri ve Tasarım Örüntüleri Eğitimi
- AspectJ ve Spring AOP ile Aspect Oriented Programlama Eğitimi



Referanslarımız



Spring Nedir?

- Kurumsal Java uygulamalarını
 - kolay,
 - hızlı ,
 - test edilebilir

biçimde geliştirmek ve **monolitik uygulama sunucularının** dışında ortamlarda da çalıştırabilmek için ortaya çıkmış bir “**framework**”tür

- Özellikle kurumsal uygulamaların **orta katman (middleware)** ihtiyaçlarına yönelik çözümler sunar



Spring'in Gelişim Süreci

- Rod Johnson **2002** sonlarında **Expert One on One J2EE Design and Development** kitabını yazdı
- **Şubat 2003**'de Spring SourceForge'da açık kaynaklı bir proje oldu
- İlk sürüm **Mart 2004**'de çıktı
- 2004 yılında Jürgen Höller ile birlikte birde "**without EJB**" kitabını çıkardılar
- 2004'de **Interface21** kuruldu, 2007'de **SpringSource**'a dönüştü
- 2009'da SpringSource'da **VMVare**'e dahil oldu (420 M\$'a satıldı)
- Şu anda **Pivotal Software** bünyesinde **açık kaynak kodlu** olarak devam ediyor



Kurumsal Uygulamaların Özellikleri

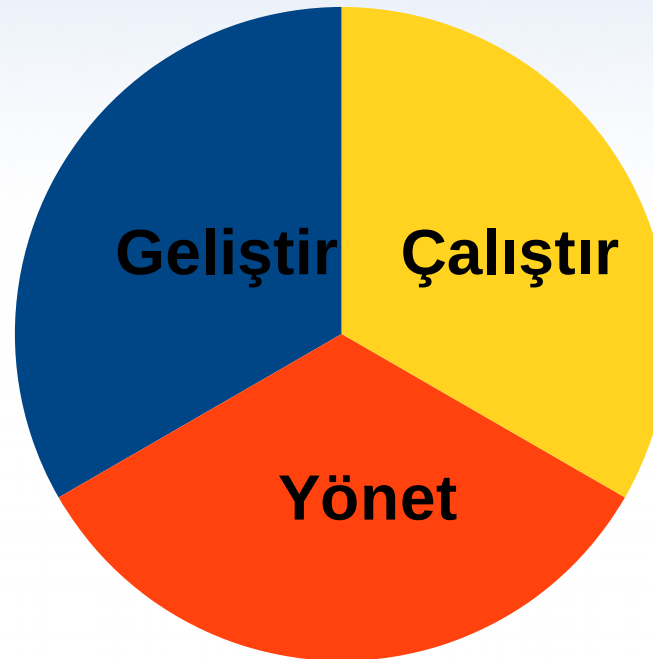
- Klasik istemci – sunucu mimarisinden farklı olarak **çok katmanlı**dırlar
- Günümüzde artık neredeyse tamamı **web tabanlı** olarak geliştirilmektedir
- **Veri yoğun** uygulamalardır, veri merkezlidirler
- Dolayısı ile **transaction** ve **güvenlik ihtiyaçları** mevcuttur
- Farklı sistemlerle **entegrasyon ihtiyaçları** da söz konusudur

Orta Katman (Middleware) İhtiyaçlar

- Veri Erişimi
- Transaction
- Güvenlik
- Audit
- Validasyon
- Cache
- Mesajlaşma
- RMI
- Nesne Havuzu Yönetimi (Pooling)
- Nesne Yaşam Döngüsü Yönetimi (Object Lifecycle)
- Loglama
- Sistem yönetimi
- ...

Spring Ekosistemi

- Kod geliştirme için Eclipse tabanlı STS IDE kullanılması
- Spring Framework ürün ailesini kullanarak uygulamaların geliştirilmesi



- Tomcat tabanlı tc Server, dm Server ortamlarında uygulamanın çalıştırılması

- vRealize Hyperic Server ile OS, middleware ve uygulamaların monitor edilmesi

Spring Programlama Modeli

Inversion of Control
Prensibi

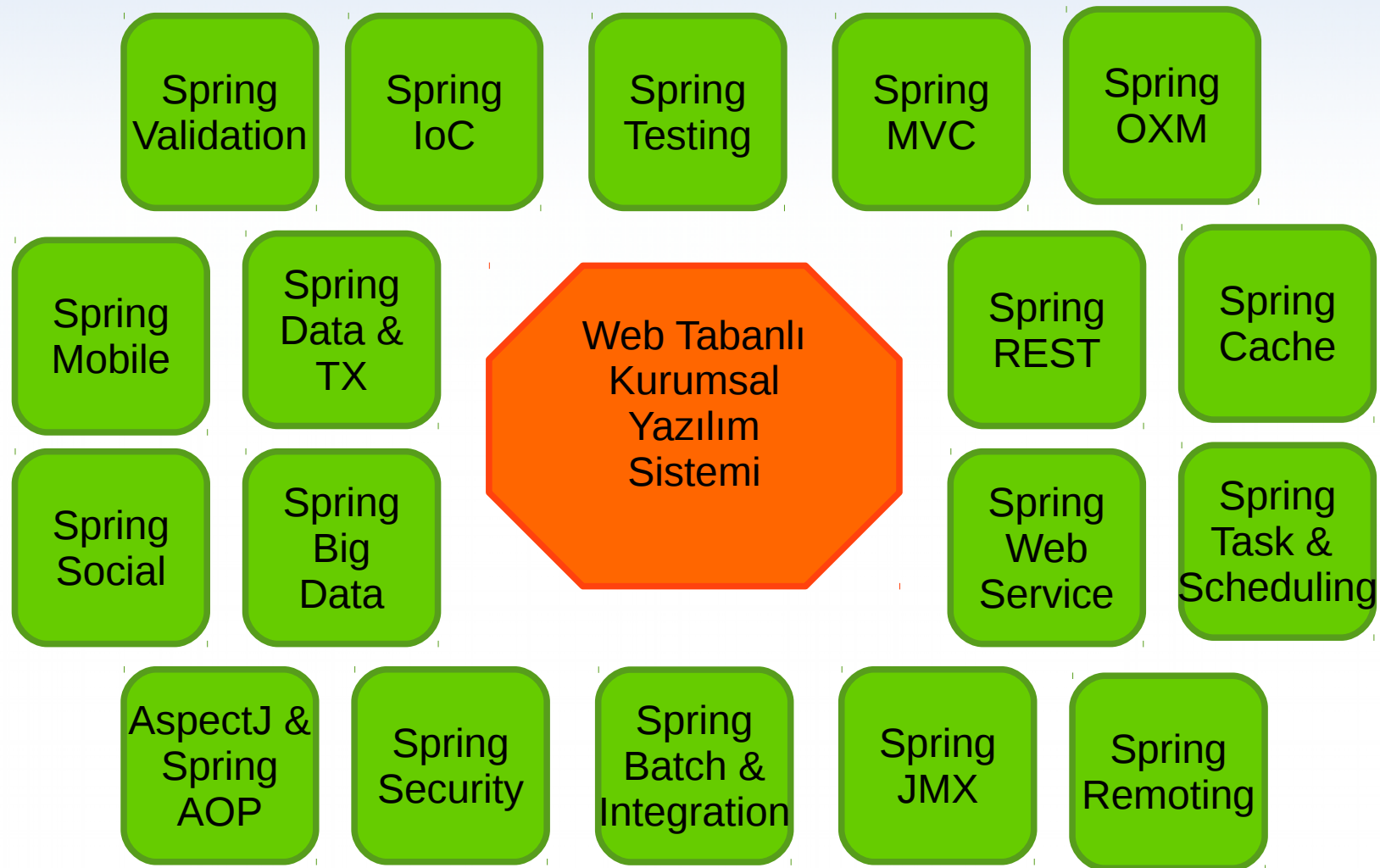
Aspect Oriented
Programlama



POJO

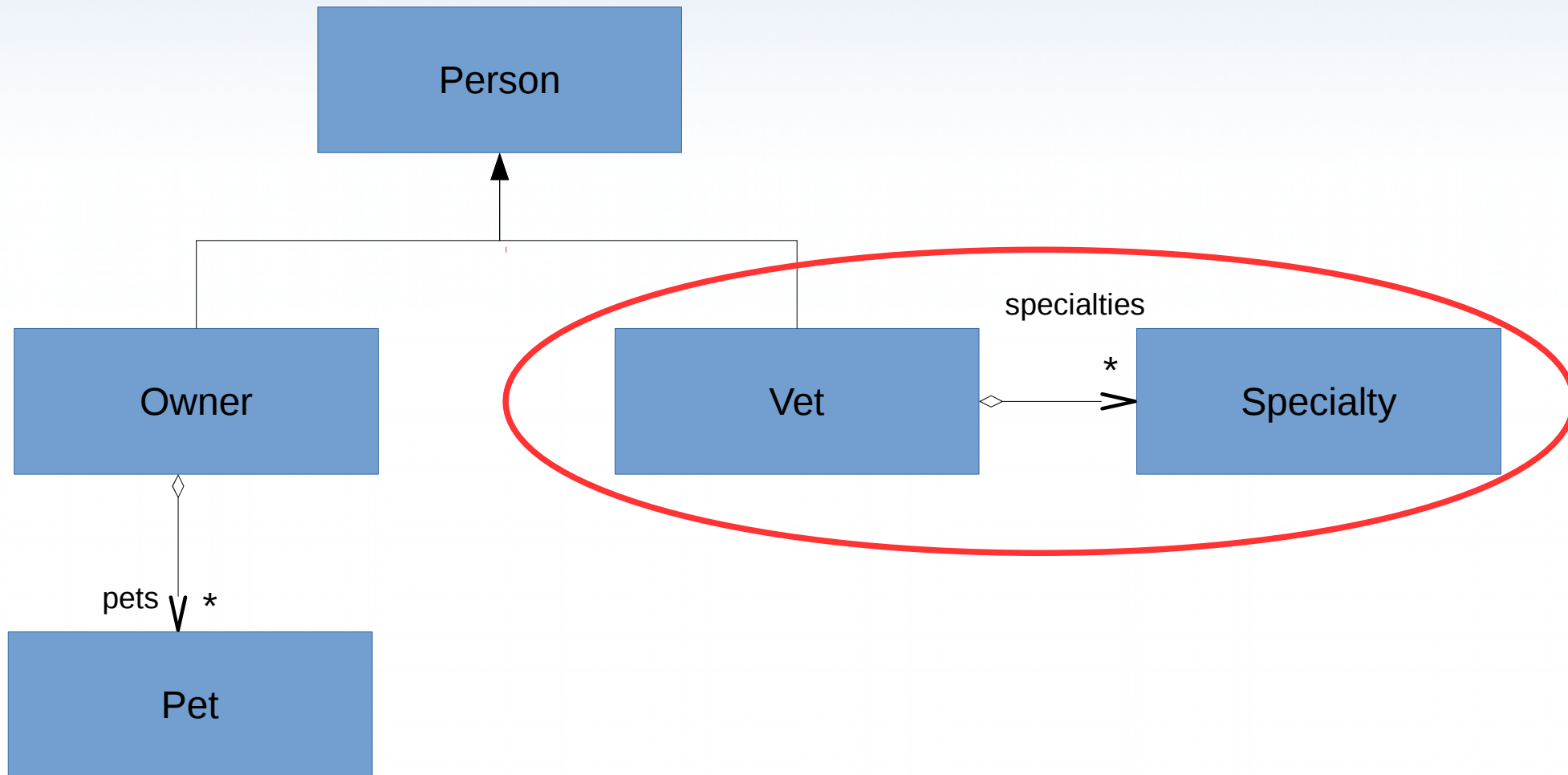
Kurumsal Altyapı
Servisleri

Spring Ekosisteminin Bileşenleri

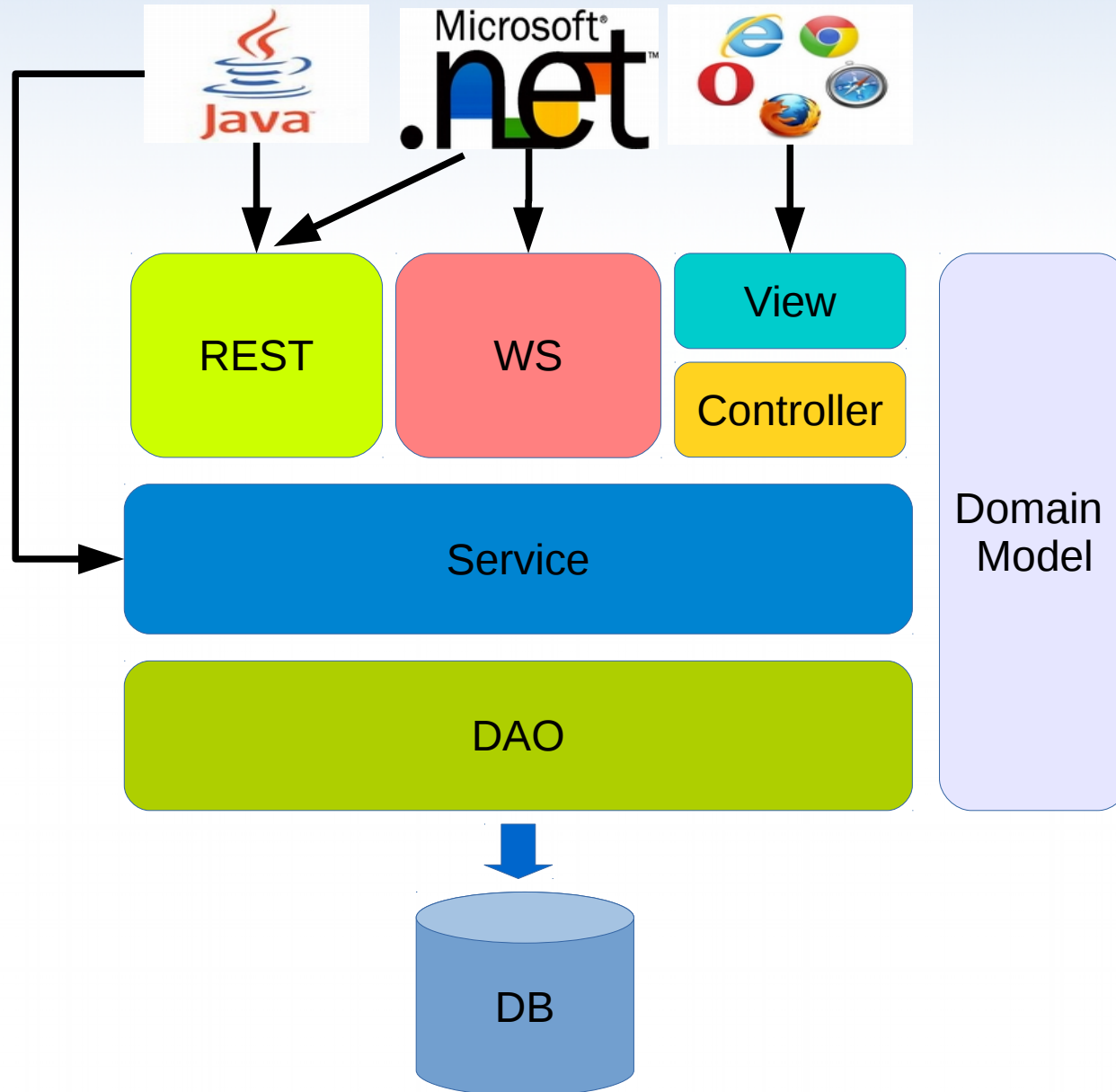


PetClinic Demo

PetClinic Domain Model



PetClinic Katmanlı Mimarisi



Demo Adımları

- Standalone Ortamda Çalışma
- Gömülü Veritabanı Kullanımı, JDBC ile DAO Gerçekleştirimi (findVets) ve Profil Kabiliyeti
- Spring ile Entegrasyon Testlerinin Yazılması
- JDBC ile DAO Gerçekleştirimi (createVet) ve Transaction Yönetimi
- Servis Düzeyinde Cache Kabiliyeti
- Servis Düzeyinde Validasyon Kabiliyeti
- Servis ve DAO Metotları için LoggingAspect Gerçekleştirimi
- Spring Konfigürasyonun Web Ortamına Aktarılması
- Spring MVC ile Veterinerlerin Görüntülenmesi (vets.jsp)
- REST Kabiliyeti
- Spring Security Konfigürasyonu ile Form Tabanlı ve HTTP Basic Kimliklendirme
- Spring Remoting ile Servislerin Java İstemciler Tarafından Erişilebilmesi
- Spring WS ile WSDL Tabanlı Web Servisleri Geliştirme

Soru & Cevap

İletişim

- Harezmi Bilişim Çözümleri A.Ş.
- <http://www.harezmi.com.tr>
- info@harezmi.com.tr