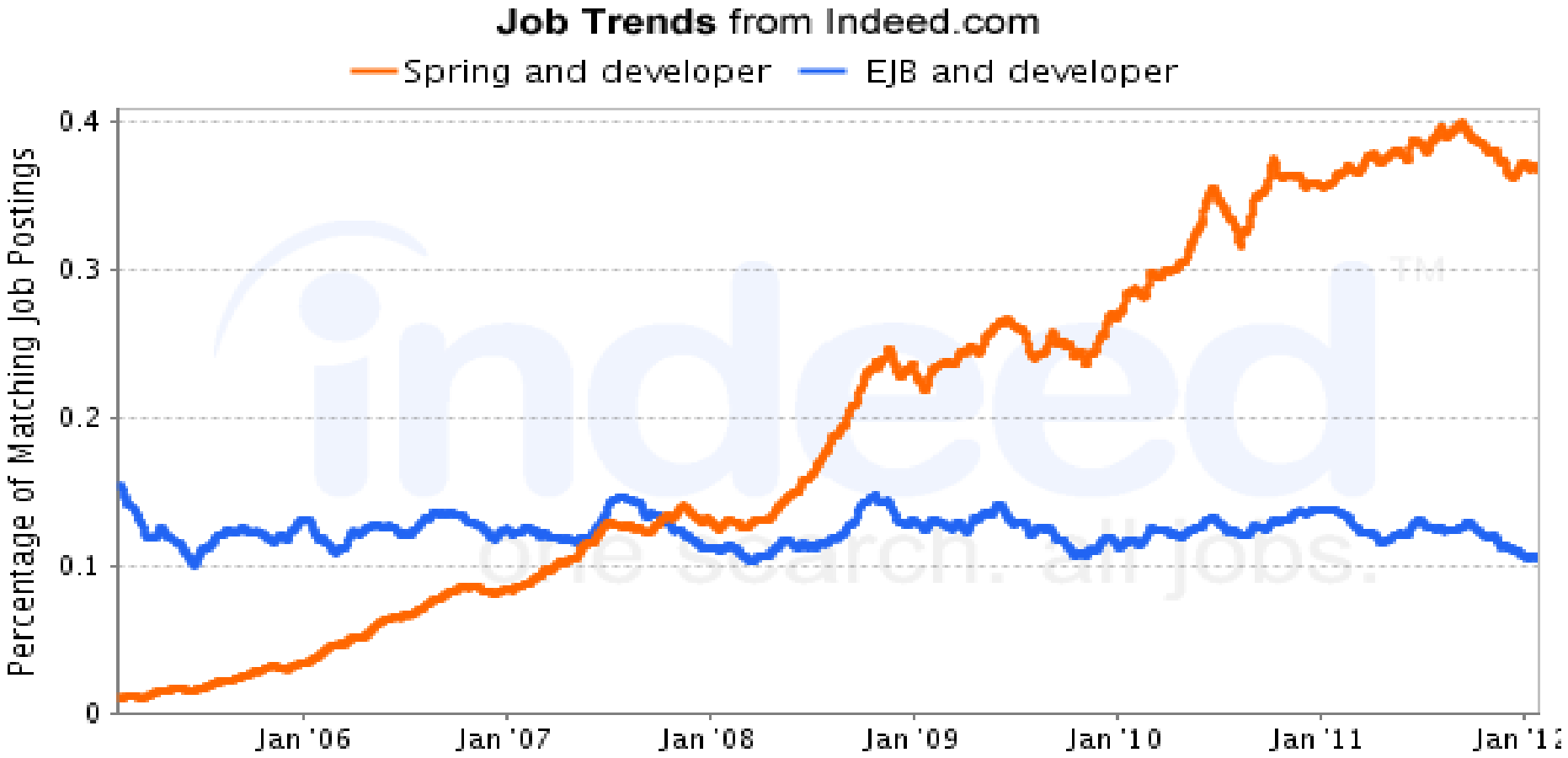
Pavel Mička

Spring Framework (A7B39WPA)

Co je to Spring Framework

- Inversion of Control/Dependency injection container
- Lze jej použít v jakékoliv Java aplikaci, nepotřebuje aplikační server
  - Široké využití na serverové straně, kde obvykle nahrazuje celý EJB container
- Snadná integrace s mnoha frameworky (JPA, JSF, Vaadin, Quartz...)
- POJO based
- Uvolněn pod Apache 2.0 licencí

Spring v praxi



Komponenty Springu

- Spring Core
  - Dependency injection container
- Spring MVC
  - MVC framework (page centric – Front controller + obvykle JSP stránka)
- Spring AOP
  - Podpora aspektově orientovaného programování
- Spring tx
  - Transaction management abstraction

Komponenty Springu 2

- Spring JDBC, Spring ORM
  - Přístup k databázi
- Spring OXM
  - Object-to-XML mapping
- Spring test
  - Balíček rpo testování spring aplikací
- A mnohé další
  - Spring Roo, Spring Security, Spring Social, Spring Mobile...

Dependency injection ve Springu

- XML based
  - Setter injection
  - Constructor injection
- Annotation based
  - Metadata injection
- Springem definované rozsahy platnosti bean
  - Request
  - Session (skutečně se váže na http session)
  - Prototype
  - Singleton (výchozí)
  - Custom (lze přidat vlastní, např. JSF2 flash)

Setter Injection

```
<bean id="myDataSource" class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close">
  <!-- results in a setDriverClassName(String) call -->
  <property name="driverClassName" value="com.mysql.jdbc.Driver"/>
  <property name="url" value="jdbc:mysql://localhost:3306/mydb"/>
  <property name="username" value="root"/>
  <property name="password" value="masterkaoli"/>
</bean>
```

- Beana dané třídy má pro dané vlastnosti setter
- Odkazovat lze jak hodnoty, tak kolekce, tak jiné Spring beany
- Tímto způsobem lze od jedné třídy vytvořit až mnoho různě konfigurovaných instancí
  - Vhodné při tvorbě cache různých rozsahů
- Setter injection je používána zejména pro konfiguraci aplikace a integraci dalších technologií

Constructor injection

```
<bean id="exampleBean" class="examples.ExampleBean">
  <!-- constructor injection using the nested <ref/> element -->
  <constructor-arg>
    <ref bean="anotherExampleBean"/>
  </constructor-arg>
  <!-- constructor injection using the neater 'ref' attribute -->
  <constructor-arg ref="yetAnotherBean"/>
  <constructor-arg type="int" value="1"/>
</bean>
```

- Obdoba setter injection, která používá konstruktor pro nastavení vlastností
- Nejméně používaná

Metadata injection

```
@Component("MovieRecommender")
@Scope("request")
public class MovieRecommender {

    @Autowired
    private ApplicationContext context;

    public MovieRecommender() {
    }

    // ...
}
```

- V rámci aplikace nejpoužívanější
- @Component – jedná se o Spring beanu
- @Scope – rozsah platnosti beany
- @Autowired – injektuj daný field

Příklad použití (Spring + JPA + jUnit)