

# The Spring Framework

Hands-On Workshop



IndicThreads.com Conference On Java Technology  
26th & 27 Oct. 2007 Pune, India

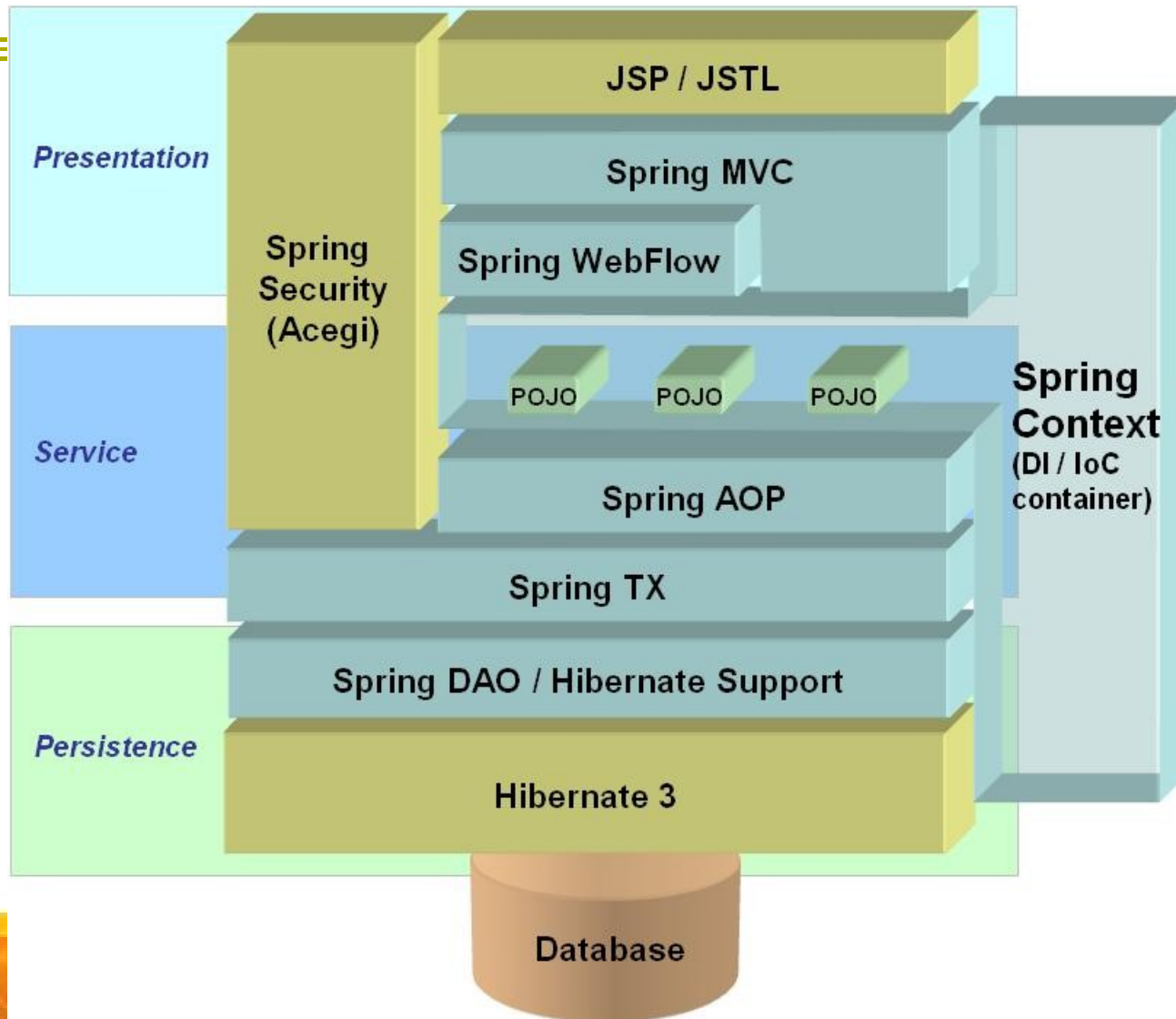


Peter Thomas

<http://ptrthomas.wordpress.com>

GMail: ptrthomas

# Sample Architecture ( <http://jtrac.info> )



# Dependency Injection

```
<bean id="jtrac" class="info.jtrac.JtracImpl">  
  <property name="dao" ref="dao" />  
</bean>
```

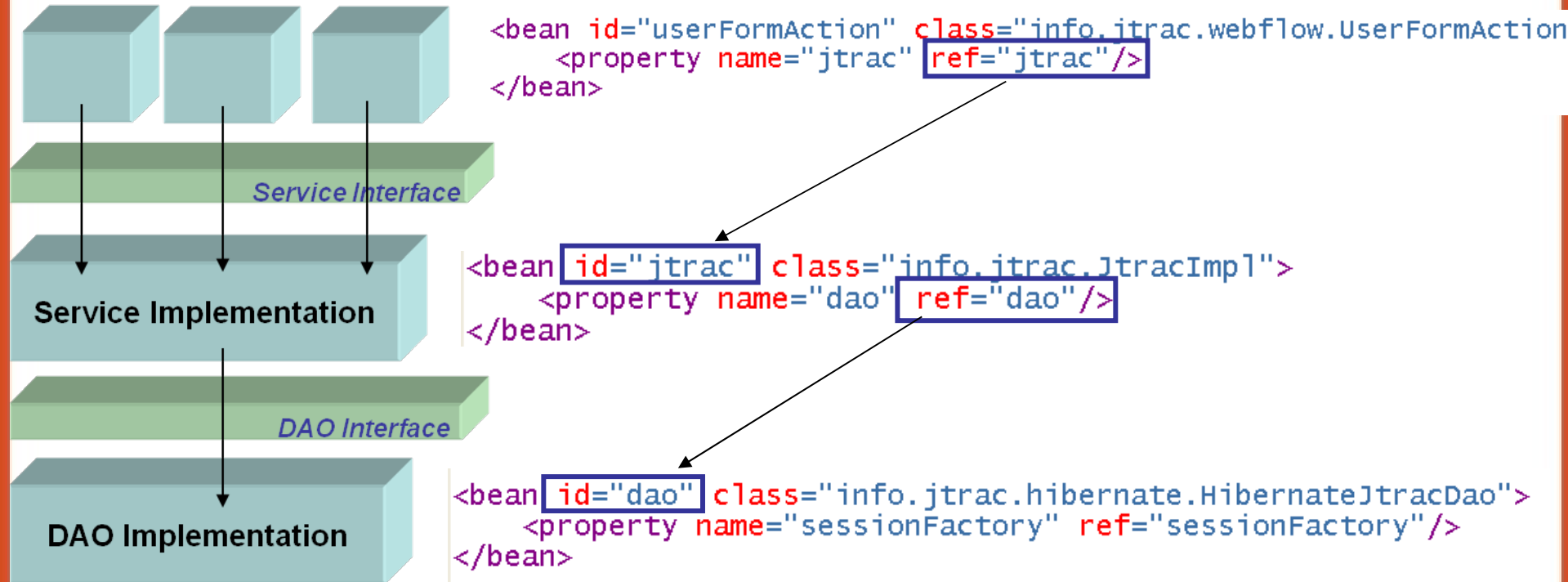
JtracImpl.java

```
private JtracDao dao;  
  
public void setDao(JtracDao dao) {  
    this.dao = dao;  
}  
  
public void storeUser(User user) {  
    dao.storeUser(user);  
}
```

```
<bean id="dao" class="info.jtrac.hibernate.HibernateJtracDao">  
  <property name="sessionFactory" ref="sessionFactory" />  
</bean>
```

# Dependency Injection (2)

## Controllers



# Hibernate with Spring

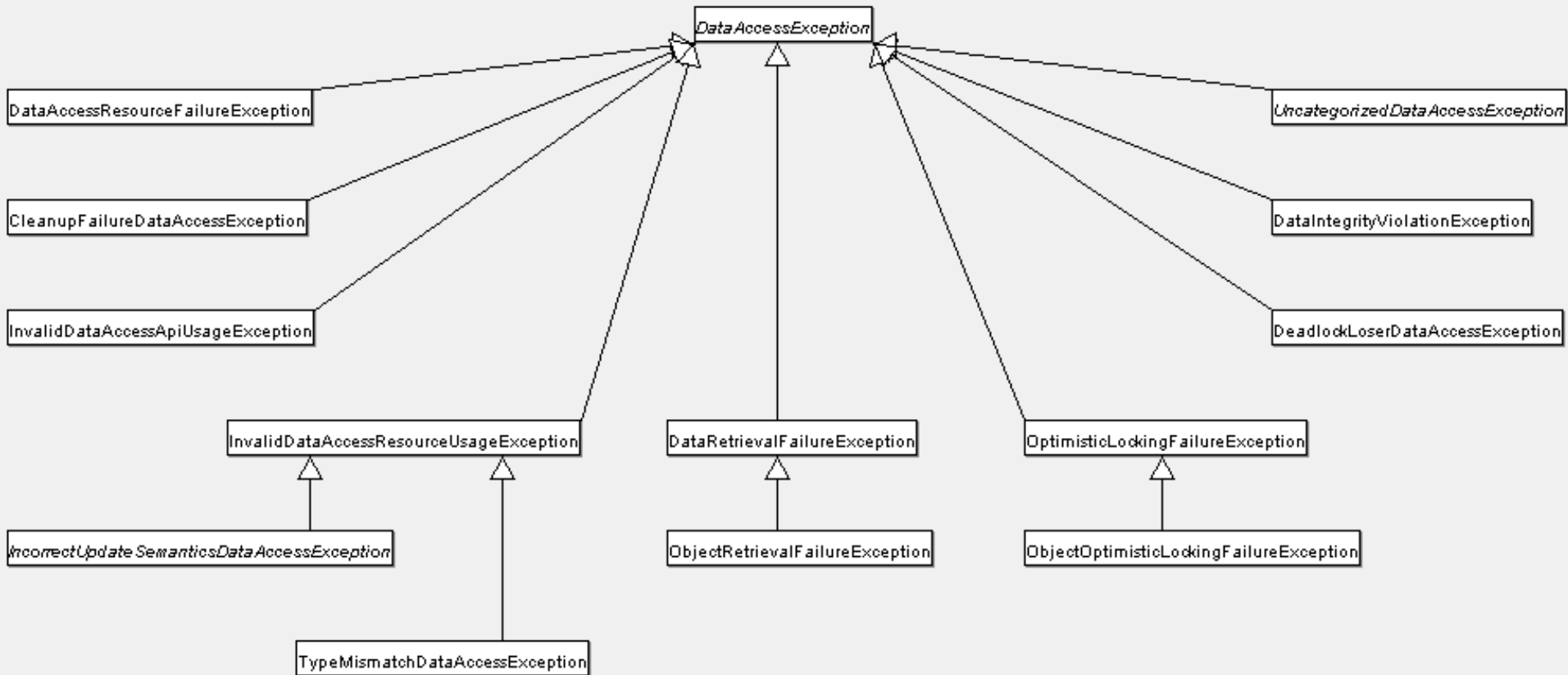
- One liners
- Template pattern
- No need to deal with TX, Session etc.

```
public void storeUser(User user) {
    getHibernateTemplate().merge(user);
}

public User loadUser(long id) {
    return (User) getHibernateTemplate().get(User.class, id);
}

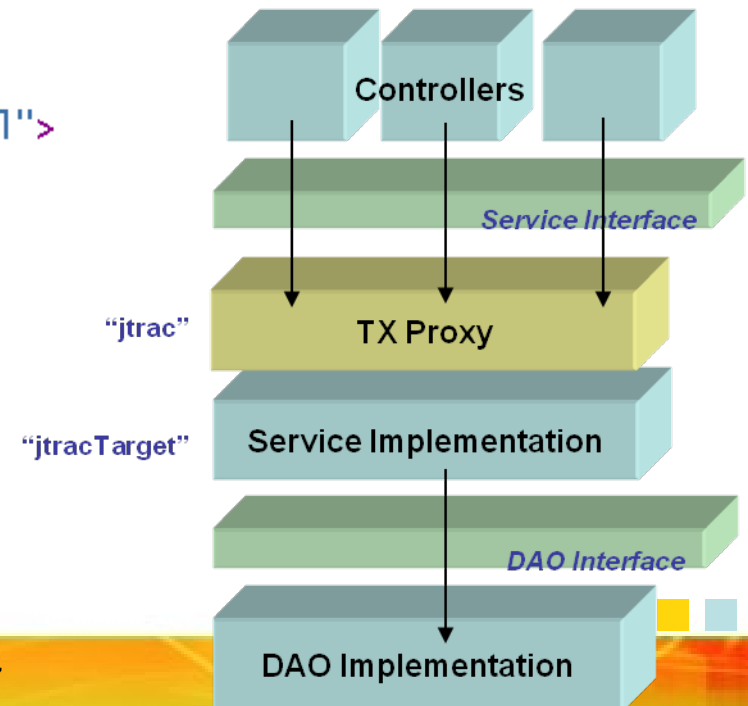
public List<User> findAllUsers() {
    return getHibernateTemplate().find("from User user order by user.name");
}
```

# Spring DAO – Exception Hierarchy



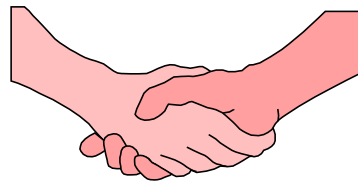
# Spring AOP / Declarative TX Mgmt.

```
<bean id="jtrac" class="org.[sf].transaction.interceptor.TransactionProxyFactoryBean">  
  <property name="transactionManager" ref="transactionManager"/>  
  <property name="target" ref="jtracTarget"/>  
  <property name="transactionAttributes">  
    <props>  
      <prop key="store*">PROPAGATION_REQUIRED</prop>  
      <prop key="remove*">PROPAGATION_REQUIRED</prop>  
      <prop key="*">PROPAGATION_SUPPORTS,readOnly</prop>  
    </props>  
  </property>  
</bean>  
  
<bean id="jtracTarget" class="info.jtrac.JtracImpl">  
  <property name="dao" ref="dao"/>  
</bean>
```



# JTrac: Spring implications

- Application deployed as a WAR file
- Portable: Tomcat, Jetty, JBoss or any Java EE app server
- Easily unit-testable
- Faster build-deploy-test cycles
- Clean OO design, designed to interfaces
- No EJBs
- No Singletons
- No Service Locator / JNDI lookup
- No custom “Factory Pattern” implementation
- No DTOs / VOs
- No Annotations (optional)
- Standard JSTL / JSP



Thank You



IndicThreads.com Conference On Java Technology  
26th & 27 Oct. 2007 Pune, India

