

Wprowadzenie do Spring Framework

Paweł Kłeczek

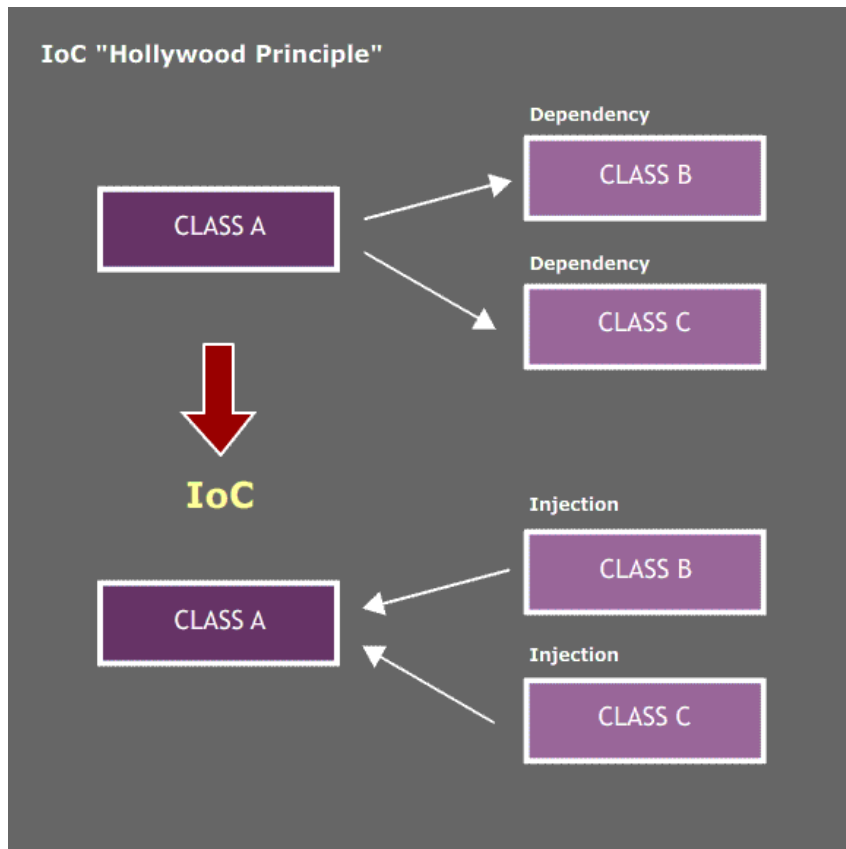
pkleczek@pgs-soft.com

<https://github.com/pkleczek/spring-politechnika>

Plan

- Odrobina teorii
- Od „złej” aplikacji do Springa
- Spring Boot
- Live coding

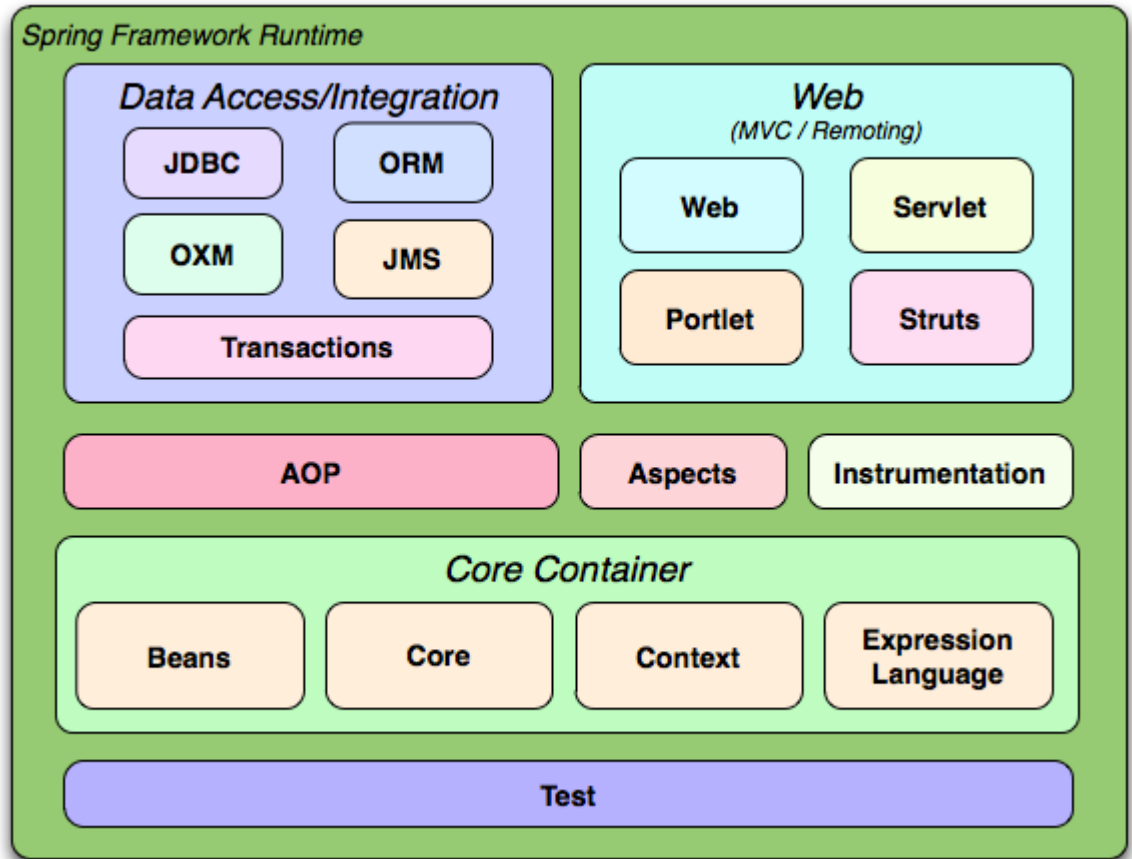
Inversion of control



- Dependency injection – jedna z realizacji
- Demo
 - no-dependency-injection
 - simple-dependency-injection

Spring Framework

- Rozwijany od 2003 roku
- Bodziec dla rozwoju Java EE
- Fundament ekosystemu *Spring Project*
 - Spring Data
 - Spring Security
 - Spring Social
 - Grails
 - Groovy...
- Spring to nie tylko IoC



```

<?xml version="1.0" encoding="UTF-8"?>
<!--
    Repository and Service layers
-->
<beans xmlns="http://www.springframework.org/schema/beans" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:p="http://www.springframework.org/schema/p" xmlns:context="http://www.springframework.org/schema/context"
    xmlns:tx="http://www.springframework.org/schema/tx" xmlns:jpa="http://www.springframework.org/schema/data/jpa"
    xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd
        http://www.springframework.org/schema/data/jpa http://www.springframework.org/schema/data/jpa/spring-jpa.xsd
        http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx.xsd
        http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context.xsd">

    <beans profile="jdbc">
        <!-- Transaction manager for a single JDBC DataSource (alternative to JTA) -->
        <bean id="transactionManager" class="org.springframework.jdbc.datasource.DataSourceTransactionManager"
            p:dataSource-ref="dataSource"/>

        <bean id="jdbcTemplate" class="org.springframework.jdbc.core.JdbcTemplate">
            <constructor-arg ref="dataSource"/>
        </bean>

        <bean id="namedParameterJdbcTemplate"
            class="org.springframework.jdbc.core.namedparam.NamedParameterJdbcTemplate">
            <constructor-arg ref="dataSource"/>
        </bean>

        <context:component-scan base-package="org.springframework.samples.petclinic.repository.jdbc"/>
    </beans>

    <beans profile="jpa">

```

Co trudnego jest w Springu? 😊

Trudne na początku

- Zależności Maven/Gradle
- Konfiguracja serwera w projekcie
- Konfiguracja frameworka
- Integracja z JPA, bazą danych
- „Jak nazywała się ta klasa?”
- Demo

org.springframework.web.servlet.support
Class AbstractAnnotationConfigDispatcherServletInitializer
java.lang.Object
org.springframework.web.servlet.support.AbstractContextLoaderInitializer
org.springframework.web.servlet.support.AbstractDispatcherServletInitializer
org.springframework.web.servlet.support.AbstractAnnotationConfigDispatcherServletInitializer

All Implemented Interfaces:
WebApplicationInitializer

public abstract class **AbstractAnnotationConfigDispatcherServletInitializer**
extends **AbstractDispatcherServletInitializer**
Base class for WebApplicationInitializer implementations that registers a **DispatcherServlet** configured with annotated classes, e.g. Spring's **@Configuration** classes.
Concrete implementations are required to implement **getRootConfigClasses()**, **getServletConfigClasses()**, as well as **getServletMappings()**. Further template and customization methods are provided by **AbstractDispatcherServletInitializer**.

Spring Boot

- Prekonfigurowane pakiety
- Samorozszerzająca się konfiguracja
- JavaConfig
- Przejrzysty pom.xml, predefiniowane wersje wielu zależności
- <http://start.spring.io> – generator załączka aplikacji
- [Startery dla wielu technologii](#)

Live coding

- @Autowiring i @ComponentScan
- Bean scopes (singleton, prototype, request, session)
- Komponenty @Service, @Repository, @Controller
- Encje i mechanizm @Transactional
- Spring Data JPA

Dziękuję





PGS Software S.A.

Tel.: +48 71 79 82 692

Fax: +48 71 79 82 690

E-mail: pgs-soft@pgs-soft.com

www.pgs-soft.com

