

Tvorba (velkých) webových aplikací

Technologie Java Enterprise Edition

Přemek Brada, KIV ZČU

8.6.2011

Přehled tématu

- Motivace a úvod
- Infrastruktura pro velké Java aplikace
- (Java – základní přehled)

- Části třívrstvé struktury – servlety, JSP stránky
- Jednoduchá a složitější aplikace

- Frameworky pro webové Java EE aplikace
- Další enterprise Java technologie

Co jsou velké webové aplikace?

- Často zvané „enterprise“
- Velké množství klientů (>1000)
- Složitá struktura, velký rozsah (subsites, tisíce stránek)
- Integrují více systémů dohromady
- Vytvářené postupně mnoha týmy



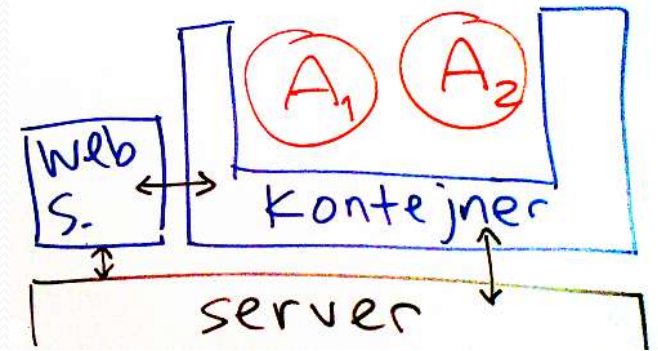
Pozn: jsou weby ještě větší...



... které nepoužívají Java technologie

Jak vypadá aplikace „zevnitř“

- Více aplikací v jednom serveru
- Vzájemně oddělené
 - administrace, aktualizace
 - bezpečnost, výkon, ...



- Složená z komponent, podle 3-vrstvé architektury
 - **výkonné** – tzv. „servlety“
 - **zobrazovací** – generují HTML stránky
 - **persistenční** – ukládání dat
 - **pomocné** – např. filtry

Zastávka: přehled jazyka Java



image by Mokund <http://www.flickr.com/photos/trevi55/296946221/>

Java – co je to zač?

- Jeden z nejdůležitějších jazyků pro sw průmysl
 - zhruba 20% veškerého software [data z tiobe.com]
 - spolu s C, C++, Microsoft .NET jazyky (C#), PHP, Python
- Většina syntaxe původem z C
 - spolu s C++, C#, PHP
- Objektový jazyk
- Velká síla knihoven
- Překládána do bytecode, prováděna JVM
 - platformová nezávislost

Hello.java

```
package cz.zcu.kiv.hello;

public class Hello {
    public static void main() {
        System.out.println("Hello, " + kohoPozdravit());
    }

    private static String kohoPozdravit() {
        return "world";
    }
}
```



```
import java.util.ArrayList;
import java.util.List;
```

```
public class Hello {
```

```
    public static void main(String[] args) {
```

```
        int pocetParam = args.length;
```

```
        List<String> parametry = new ArrayList<String>();
```

```
        for (int i = 0; i < pocetParam; ++i) {
```

```
            String p = args[i].trim();
```

```
            if ("".equals(p))
```

```
                continue;
```

```
            parametry.add(p);
```

```
        }
```

```
        // zpracovani seznamu
```

```
    }
```

```
}
```

Hello2.java

Java EE – součástí jsou nástroje pro web

Java: API

The image shows a collage of screenshots from the Java EE API Specification website. The main screenshot displays the 'Overview' page for the 'Java™ Platform, Enterprise Edition 6 API Specification'. It lists various packages such as `javax.activation`, `javax.annotation`, `javax.annotation.security`, `javax.decorator`, `javax.ejb`, `javax.ejb.embeddable`, `javax.ejb.spi`, `javax.enterprise.context`, `javax.enterprise.context.spi`, and `javax.enterprise.deploy`. A yellow speech bubble highlights the text 'Java EE – součástí jsou nástroje pro web'. Other screenshots show the 'Class StackOverflowError' page, the 'Interface HandlerContext', and the 'Method Summary' for `ServletContext`.

Výhybka: umíte HTML
formuláře?

HTML formuláře

- Příklad
- form - action - method
- input type, name, value

- Validace hodnot, předvyplnění
- Dvojí odeslání

Základní programování webů v Javě

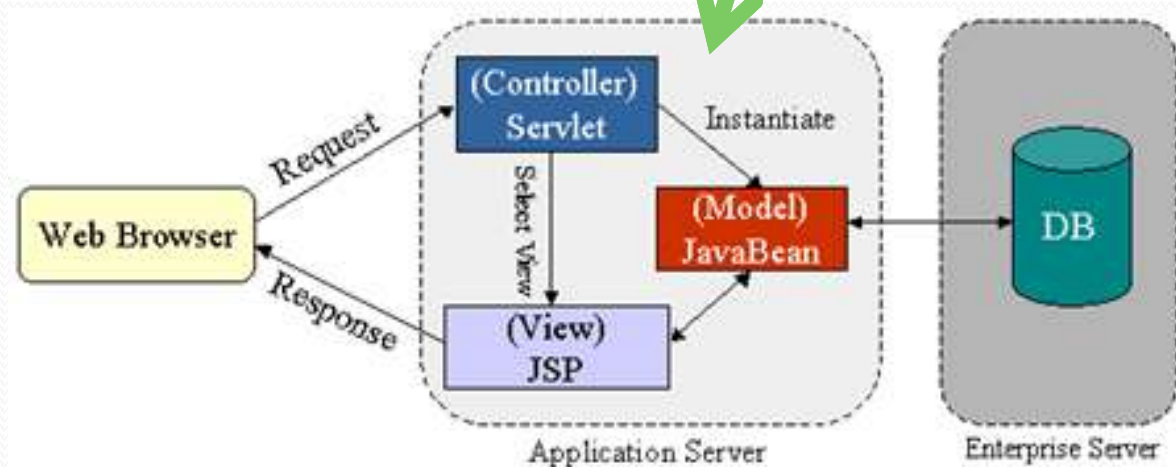
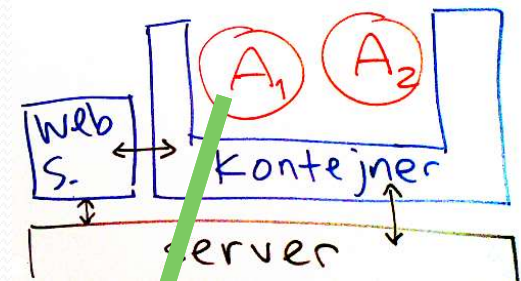
Části třívrstvé struktury v Java EE

Java EE

- **servlety** – výkonné komponenty
- **JSP stránky** – zobrazovací vrstva
 - generují HTML stránky
- filtry

Standardní Java

- persistenční třídy – ukládání dat



Jednoduchá aplikace

- Servlet
 - obslužné metody – zpracování GET, POST
 - získávání dat požadavku = hodnot z formuláře
 - předávání hodnot do dalších kroků zpracování
 - nasměrování na zobrazovací JSP
- JSP
 - vkládání Java kódu do HTML
 - získání hodnot nastavených servlety
 - zobrazení hodnot a dalších výsledků

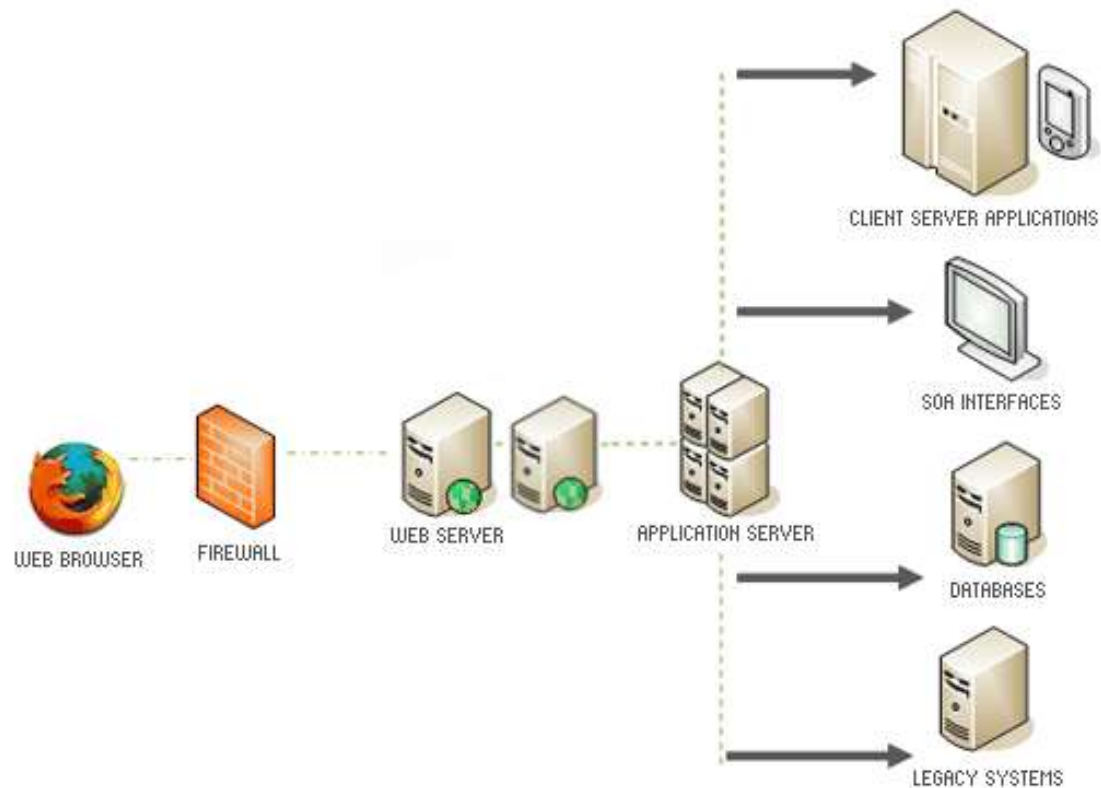
Vybrané oblasti

- Aplikační logika
 - ověřování a přihlašování uživatelů
 - udržování hodnot během session – nákupní košík
 - validace vstupních hodnot
- Zobrazení
 - společné části stránek
 - JSTL v JSP stránkách – procházení kolekcí
 - lokalizace a výpis zpráv
- Vázané na HTTP a vlastnosti webu
 - chybové kódy
 - nastavování vlastností odpovědi

Provoz webové aplikace

Co je potřeba pro provoz takových „enterprise“ aplikací

- Komplexní infrastruktura
- web server + proxy
- aplikační server(y)
- databázový server
- adresářové služby
- integrační služby
- clustery, firewall



Co je potřeba pro aplikaci webovou

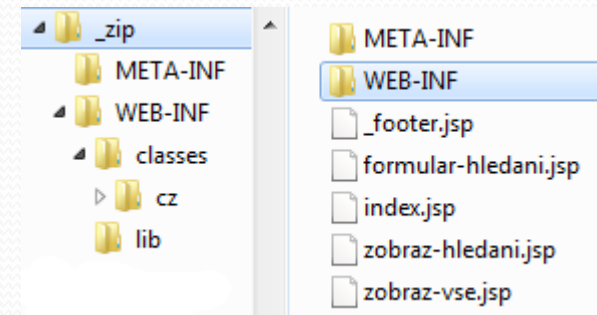
- Podmnožina předchozího schematu
- server (hardware)
- webový kontejner
- (databáze)
- někdy předřazen „holý“ web server
- Kontejner = aplikace pro provoz aplikací



Struktura a nasazení aplikace

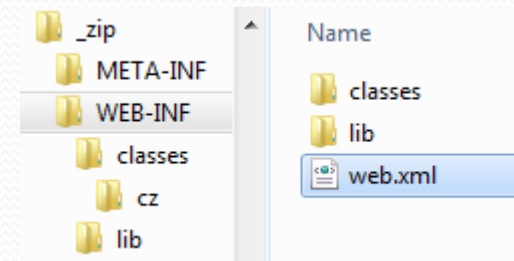
- Implementační struktura

- standardizovaný archivní formát (.war)
- adresářová struktura – kontejner načítá
- popisný soubor aplikace



- Proces nasazení

- vytvoření archivu
- nasazení do kontejneru – hot deploy / management app
- (restart kontejneru)

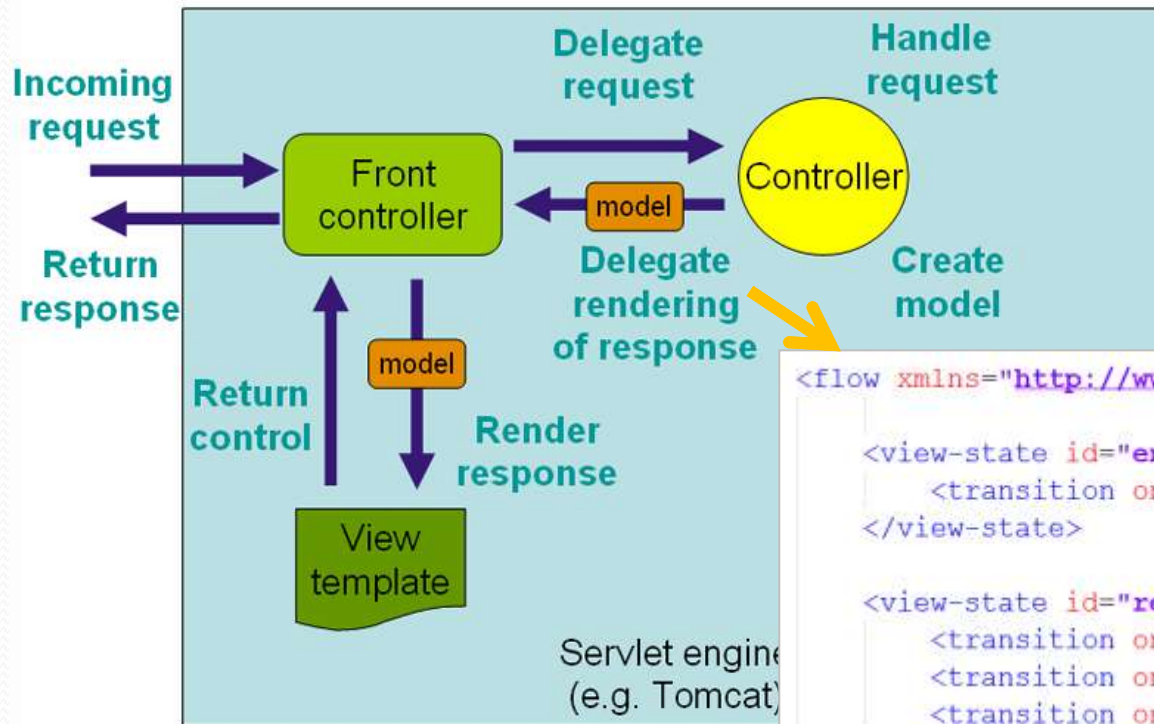


Jak se velké aplikace vytváří?

- Technologická stránka věci
- Omezení rutinních činností a částí programového kódu
 - práce s daty v databázi
 - ověřování uživatelů
 - zobrazování bloků stránky
 - předzpracování dat z formulářů
- Usnadnění vývoje a aktualizace
 - rozdělení do modulů, které jsou samostatně nasaditelné
 - snadné logování – sledování provozu
 - generování vzhledu pro různá koncová zařízení

frameworky

Frameworky pro webové Java EE aplikace – Spring MVC, ...



```
<flow xmlns="http://www.springframework.org/schema/webflow">
  <view-state id="enterBookingDetails">
    <transition on="submit" to="reviewBooking" />
  </view-state>

  <view-state id="reviewBooking">
    <transition on="confirm" to="bookingConfirmed" />
    <transition on="revise" to="enterBookingDetails" />
    <transition on="cancel" to="bookingCancelled" />
  </view-state>

  <end-state id="bookingConfirmed" />

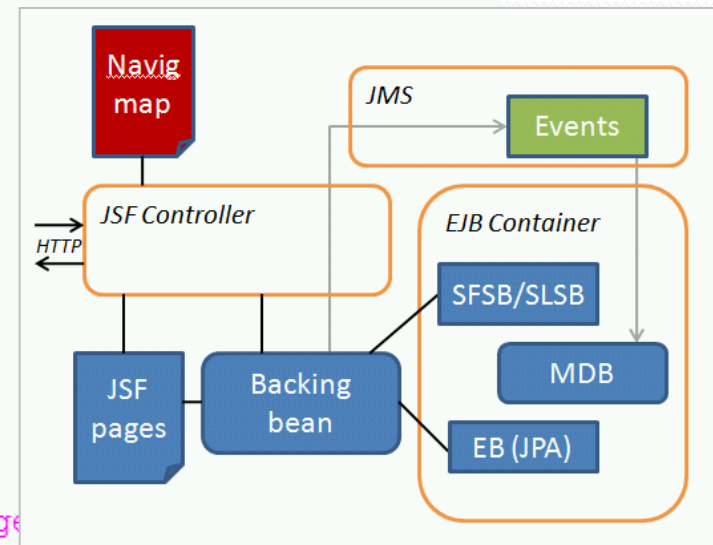
  <end-state id="bookingCancelled" />
</flow>
```

obrázek z <http://static.springsource.org/spring/docs/2.5.x/ref>

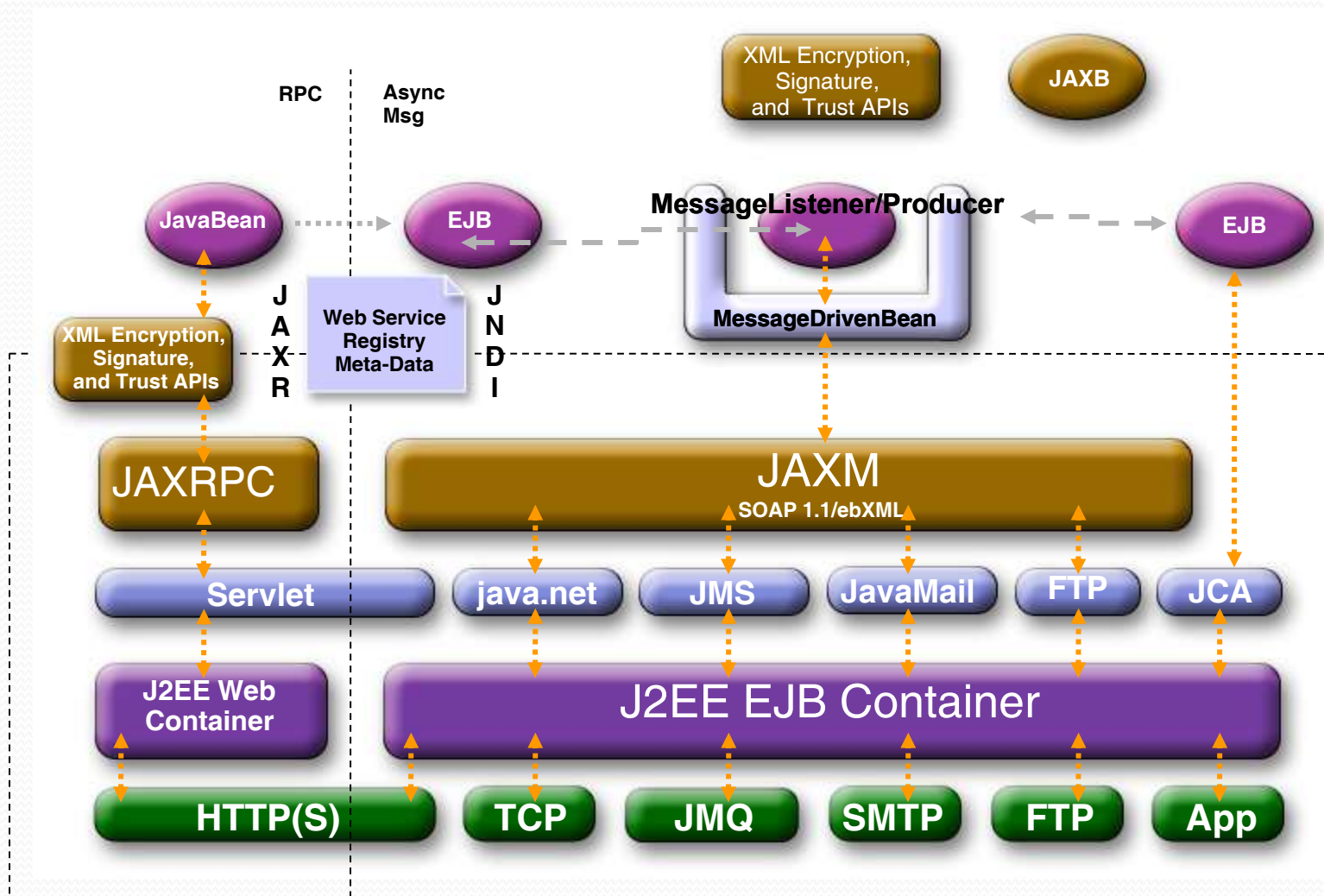
Frameworky pro webové Java EE aplikace – Java ServerFaces

- Podobný přístup jako desktopové aplikace – zpracování událostí na GUI

```
<h:panelGrid>
<h:dataTable value="#{inventoryManagerBean.inventoryForCategory}" var="items"
border="1" frame="none" style="text-align:center;">
<h:column>
<h:inputText value="#{items.item.name}"/>
<f:facet name="header">
<h:outputText value="Name"/>
</f:facet>
</h:column>
...
<h:column>
<f:facet name="header">
<h:outputText value="Select"/>
</f:facet>
<h:commandButton value="orders"
type="submit"
actionListener="#{inventoryManage
action="orderItem">
<f:facet name="extraParameter">
```



Další enterprise Java technologie



Výhody?

- Je s tím poměrně dost práce (v porovnání s PHP)...
- Robustnost
 - možnost otestovat, zabezpečit
 - ověřené technologie s garancí spolehlivosti => enterprise
- Škálování
 - rozložení vrstev mezi fyzické stroje
 - clustery, cache – součást možností kontejnerů
- Udržovatelnost
 - dobré (a vynucené) oddělení vrstev a modulů aplikace

Zdroje informací

Knihy. Web

- Computer Press, Grada – ?
- Amazon.com
- <http://www.oracle.com/technetwork/java/javasee/> > The Java EE 5 Tutorial
- javaworld.com > „servlet tutorial“, „jsp tutorial“
- stackoverflow.com

