

J2EE architektúra v praxi

Skúsenosti s J2EE implementáciami

Peter Obert

04.11.2006

SOFTEC

Frameworkový prístup

- Z pohľadu architektúry je framework množina spolupracujúcich objektov, ktoré umožňujú znovupoužitie návrhu pre špecifickú oblasť.
- Framework možno považovať za jednu konkrétnu realizáciu vzoru alebo skupiny vzorov.

2

SOFTEC

Prečo FW?

- Nemerateľné prínosy
 - **Standardizácia**
 - Životného cyklu projektu
 - Životného cyklu aplikácie
 - Architektúry, podpory a údržby
 - **bezpečnosti, škálovateľnosti, dostupnosti a transakčnosti**
 - Kontrola kvality
 - Personálna nezávislosť
 - Údržba jednotného frameworku
 - Znížené riziko pri vývoji aplikácií
 - Znížené náklady vývoja, prevádzkovania, údržby
 - Presnejšie plánovanie



3

SOFTEC

Prečo FW?

- Merateľné prínosy
 - Jednotná architektúra a metodika
 - Demo
 - Znovu použitie komponentov
 - Znovupoužitelnosť aplikačnej logiky
 - Znížené náklady na dokumentáciu
- „Stupeň frameworkizácie“
- Prínosy závisia od životnosti jednotlivých súčastí FW prístupu

4

SOFTEC

Jednoduchý Web Framework

- Front controller stratégia
 - Service servlet
 - Custom JSP tag library
 - Stateless session EJB
 - Dataaccess komponent
 - Aplikačná logika v databáze
 - Externé riadenie XML konfigurácie
 - Aplikačné xml
 - View xml
 - FormValidator xml
 - Dataaccess xml

5

SOFTEC

MDA prístup v J2EE

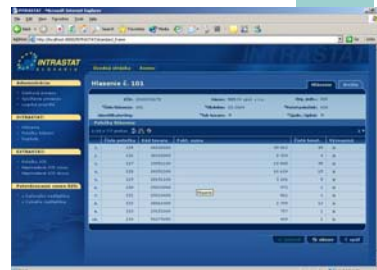
- MDA - **Model-driven architecture** je metodológia [softvérového návrhu](#), vydaná a podporovaná konzorciom [Object Management Group](#).
- Modelové riešenie aplikácie sa transformuje postupne do programového riešenia
 - Platform independent model
 - Platform specific model
- Zahŕňa viacero štandardov
 - [Unified Modeling Language](#) (UML)
 - [Meta-Object Facility](#) (MOF)
 - [XML Metadata interchange](#) (XMI)
 - [Common Warehouse Metamodel](#) (CWM)

6

SOFTEC

JavaTEC – určenie

- agendové aplikácie
 - relačná databáza
 - aj rozsiahle aplikácie so zložitou aplikačnou logikou
 - typické zobrazenia
 - zoznam objektov (browse)
 - detail objektu (detail form)
 - kombinácia oboch
- riešenie v sieti WAN a LAN
(centralizované alebo distribuované spracovanie)
- webové používateľské rozhranie

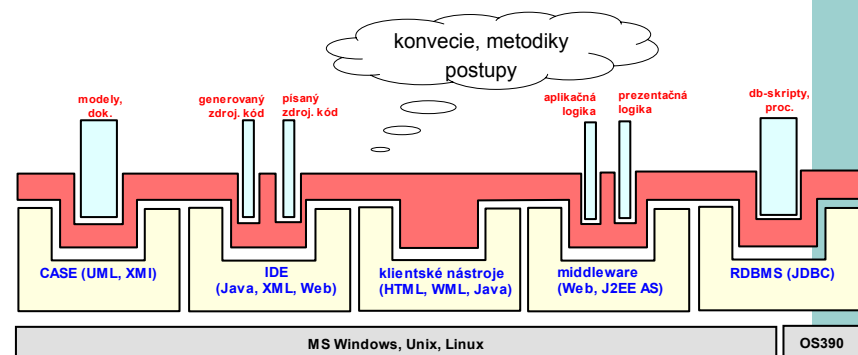


7

SOFTEC

Framework (JavaTEC)

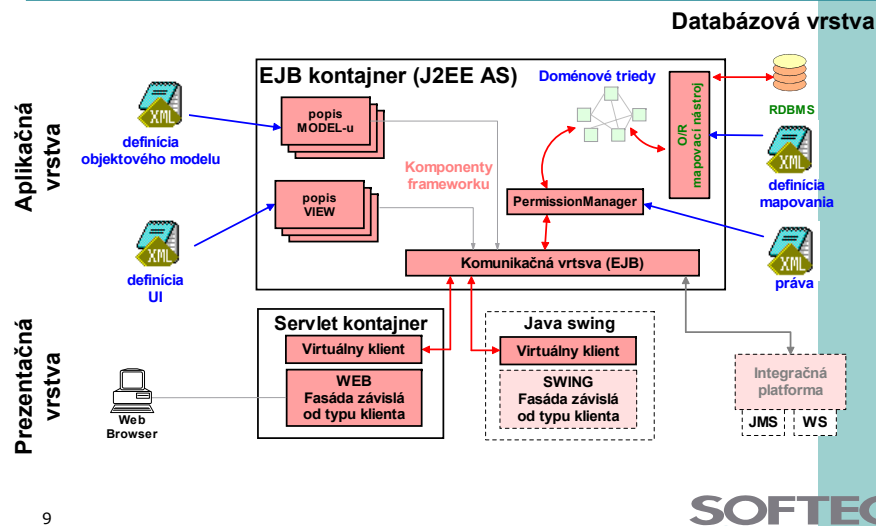
- vzťah
 - technológia – nástroj – framework – aplikácia
 - "umiestnenie" framework-u



8

SOFTEC

Architektúra – JavaTEC



9

SOFTEC

SOA Prístup v J2EE

- A service-oriented architecture je množina služieb, ktorá komunikuje medzi sebou navzájom. Služby sú self-contained a nezávisia od kontextu alebo stavu inej služby. Služby dokážu pracovať v distribuuovanej systémovej architektúre.

10

SOFTEC

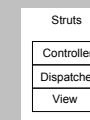
Zjednotenie obsluhy klienta

- Definícia architektury
 - Integrácia systémov
 - Transakčné systémy
 - Portálové systémy
 - Intranet a Internet
 - Front-endy legacy systémov
- Zrýchlenie vývoja aplikácií (nástroje, postupy, komponenty,...).
- Zjednodušenie podpory aplikácií (štandardizácia, dokumentácie).
- Zjednodušenie prevádzky aplikácií (jednotná platforma).
- Zníženie nákladov
 - Vývoj (sdieľanie spoločných komponent)
 - SW a HW (sdílená prostredí)

11

SOFTEC

Výsledná architektúra



12

SOFTEC

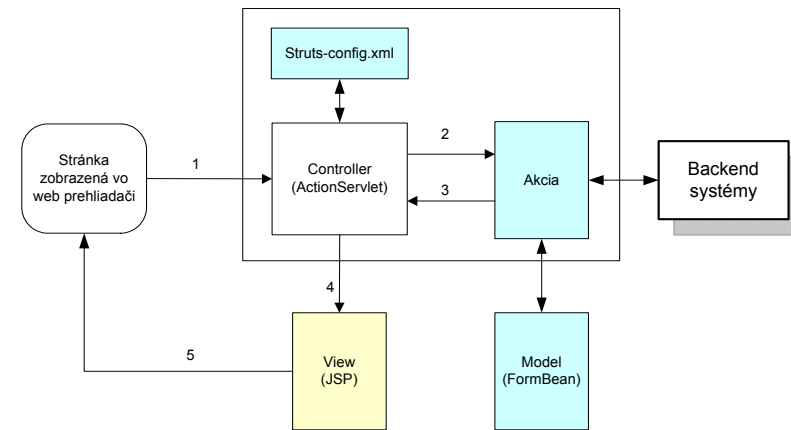
Struts - základ prezentačnej vrstvy

- Struts
 - Open source projekt zo združenia APACHE
 - Implementácia tenkého webového klienta na základe MVC patternu
- MVC pattern
 - Model - kontajner pre dáta (formbeany)
 - View – prezenčná vrstva (JSP)
 - Controller – aplikačná logika a smerovanie toku aplikácie (servlet a akcie)

13

SOFTEC

Tok požiadavky

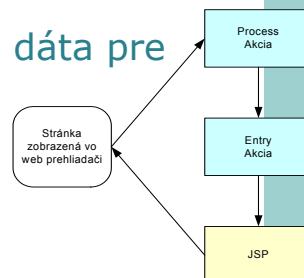


14

SOFTEC

Scenár spracovania akcie

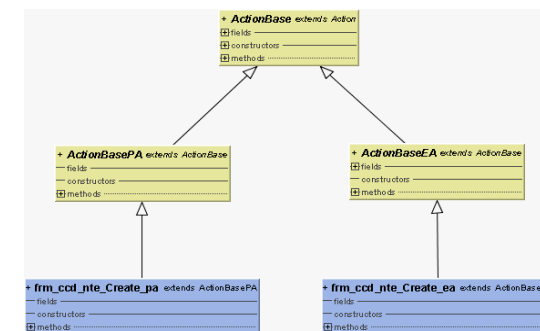
- Akcie pokrývajú funkčnosť aplikácie, sú volané na základe konfigurácie controller servletom
- Základné rozdelenie akcií:
 - Process akcie – spracúvajú request od klienta
 - Entry akcie – pripravujú dáta pre nasledujúce JSP
- Funkčnosť striktne rozdelená
 - process a entry akcie
- Predpísaná postupnosť akcií



15

SOFTEC

Objektový model akcie



16

SOFTEC

Nadstavba nad Struts

- Potreba zaviesť udalosti v akciovo orientovanom frameworku
- Objekt LocalContext – kontext akcie:
 - akcii sprístupňuje runtime informácie
 - prostredníctvom udalostí určuje „mód“ akcie
 - akcia prostredníctvom LocalContext-u určuje nasledujúci tok aplikácie

17

SOFTEC

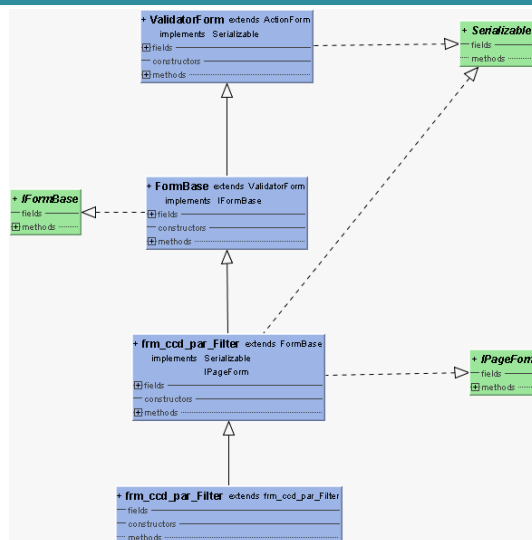
Formbeany

- Kontajner dát držaných na FE
- Všetky dáta vo formbeanch držané ako stringy
- Reprezentácia dát HTML formulára na strane servera
- Rozsah platnosti musí byť session
- Okrem samotných dát si formbeana drží dotatočné technologické informácie (ComponentState)
- Možnosť rozšírenia definície formbeany o podporu stránkovania a triedenia (cez interface)
- Základná frameworková implementácia formbeany poskytuje funkčnosť, ktorá je požadovaná akciami (implementuje IFormBase)

18

SOFTEC

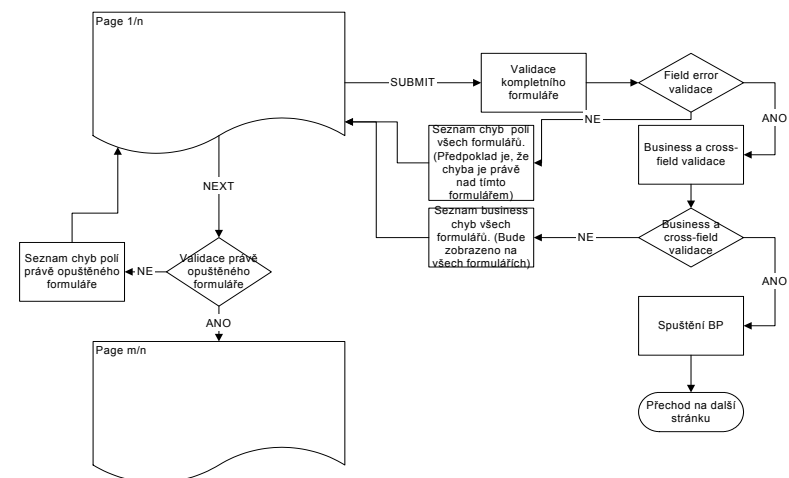
Formbeany



19

SOFTEC

Multiformbeany

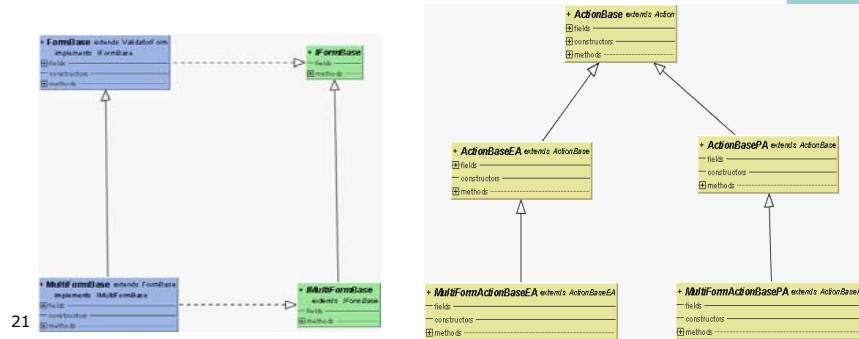


20

SOFTEC

Multiformbeany

- Vyžadujú použitie samotných predkov pre:
 - multiformbeanu, ktorá sústreďuje členské formbeany
 - entry akcie (multiformbeany a aj členských formbeanov)
 - process akcie (multiformbeany a aj členských formbeanov)



Dáta v HTTP session

- Aplikácia by nemala ukladať dáta do HTTP session nekontrolovateľne
- Odporúčané miesta na ukladanie dát do session
 - `formbeany`
 - `trieda implementujúca IFeUserData`
- Základná trieda implementujúca `IFeUserData` je `FeUserData`
 - drží dáta o aktuálne prihlásenom používateľovi
 - poskytuje funkčnosť požadovanú frameworkom
 - možnosť jej dedenia – držanie kontextu aplikácie

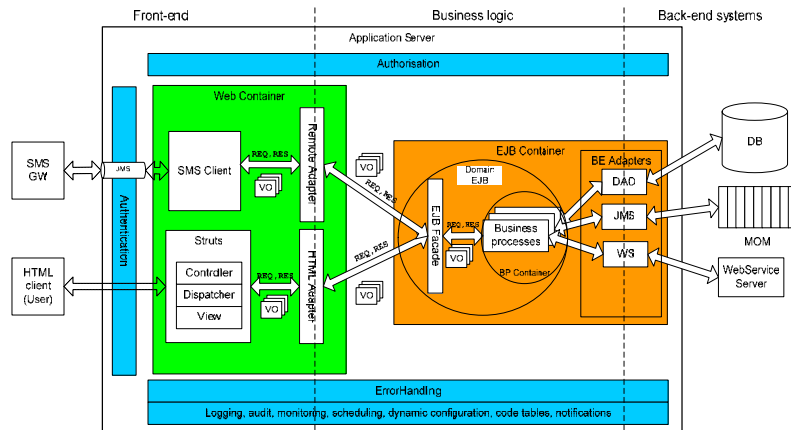
Ďalšie rozšírenia pôvodného Struts

- Napojenie na používateľské práva
 - definícia práv sa preberá pri prihlásení
 - používateľa z dynamického správanie sa ovládacích prvkov aplikácie na základe práv
- Dynamické správanie ovládacích prvkov na základe aplikačnej funkčnosti
- Formátovanie číselných výstupných hodnôt podľa masky
- Držanie dát v kontexte aplikácie
- Využitie internacionalizácie

Metamodel prezentačnej vrstvy

- Metamodel, ktorý drží dáta o rôznych oblastiach aplikácie
- Dáta potrebné pre prezentačnú vrstvu FE:
 - Definície obrazoviek
 - Definície textov
 - Napojenie obrazoviek na help
 - Definície chýb
 - Napojenie obrazoviek na BL message
 - Práva priradené ovládacím prvkom
 - Zloženie menu
 - Definícia javascriptu špecifického pre stránky
 - Technická dokumentácia
 - Definícia toku aplikácie

Aplikačná vrstva



25

SOFTEC

Základné komponenty architektúry

- Rozhranie reprezentované VO
- Business Process
- Fasáda na prístup k BP
- FE adaptéry
- BE adaptéry
- Základné služby
 - „bean utils“,
 - spracovanie a logovanie chýb,
 - monitorovanie aplikácie,
 - bezpečnosť,
 - parametrizácia a konfigurácia,
 - plánovač úloh,
 - číselníky,
 - konverzie dát,
 - spracovanie dávok(batch), servisná aplikácia,...

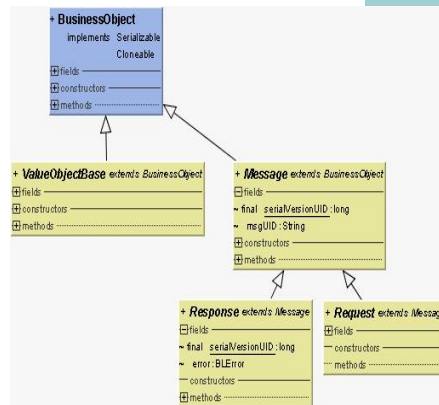
26

SOFTEC

Rozhranie aplikačnej Logiky

- Value objekty(VO)
 - prenos zložitých dátových štruktúr medzi vrstvami
 - základné všeobecné operácie nad dátami (clone, toMap, fromMap, same, ...),
 - delenie VO a ich hierarchia:
 - messages (request, responsy)
 - objekty doménovej dátovej štruktúry
 - názvová konvencia a štandardizácia názvov
 - prepojenie requestov na doménovú logiku – message routing

27



SOFTEC

Business process

BP základná jednotka logiky z pohľadu

- použitia (najjemnejšia granularita poskytovaných služieb),
- transakčného – jedna atomická transakcia,
- procesu vývoja – jednotka technického designu, implementácie a unit testu,
- monitoringu, auditu, bezpečnosti

BP spracováva požiadavku na jednotlivé fázy inicializácia

- validácia
- preprocess
- process
- zalogovanie akcií

28

SOFTEC

Prostredie pre aplikačnú logiku

Business Process Container

- riadenie spracovania (volanie jednotlivých krokov, prípadné riadenie transakcie)
- infraštruktúra pre jednotlivé BP (logovanie, autorizácia, spracovanie chýb, riadenie transakcie ...)
- pre-procesing a post-procesing dotazov
- business process kontajnery pre
 - Externé volanie
 - Interné volanie (volanie BP z BP)
 - Batchové spracovanie
 - Možnosť vytvorenia vlastného kontajneru

29

SOFTEC

Stav a riadenie aplikačnej logiky

Context BP – dáta prístupné BP

- stav autentifikácie
- stavové dáta aplikácie
- údaje o používateľovi
- údaje o použítom kanále, IP adrese používateľa, session id ...

Konfigurovanie základných vlastností BP

- logovanie
- práva
- povolené kanály
- validácie vstupných dát

30

SOFTEC

Prístup k aplikačnej logike

• Fasáda na prístup k BP

- riadi vytvorenie BP a business process kontajneru, umiestňuje BP do BPC
- logovanie chýb na výstupe z BL
- rôzne typy fasády
 - Stand-alone (pre tomcat a samostatne spúšťané BP)
 - EJB (remote, local, batch facade)
- pre cieľové prostredie sa v čase deploymentu EJB určuje transakčné správanie, priradenie do thread poolu (riadienie výkonu), bezpečnosť na technickej úrovni
- fyzicky je fasáda reprezentovaná pomocou EJB

31

SOFTEC

Organizácia aplikačnej logiky

- Organizovanie doménovej logiky do BP
 - BP základná jednotka organizácie BL
 - SUBDOMÉNA – organizácia BP do logických skupín. Sprehľadňuje členenie aplikácie. Členenie do subdomén je možné vykonať viacerými spôsobmi:
 - na základe use case
 - na základe príslušnosti BP k jednotlivým entitám
 - DOMÉNA – doména je reprezentovaná jedným EJB. Vytvorenie jednotlivých domén súvisí:
 - so škálovateľnosťou
 - s prioritami
 - s vytváraním znovupoužiteľných komponentov
 - APLIKÁCIA – základná jednotka z pohľadu:
 - vývojového cyklu tvorby softvéru
 - deploymentu a konfigurácie

32

SOFTEC

Prístup z klienta

• Poskytovanie služieb aplikačnej

- HTML Klient
 - HTMLAdaptér
- Iní klienti
 - BL Adaptér

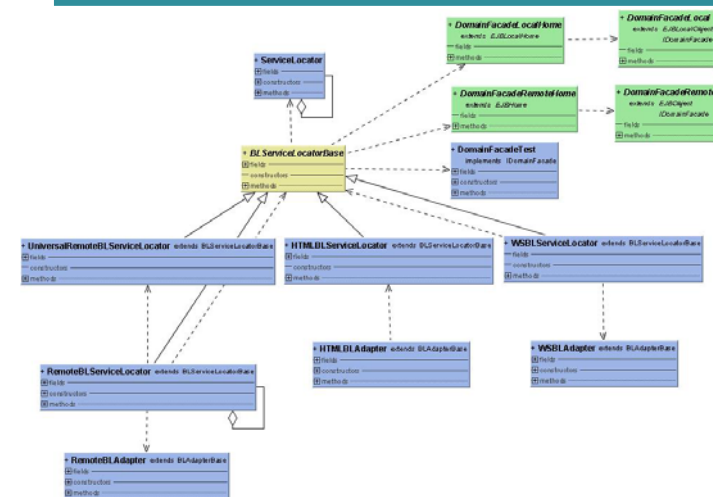
• Použitie Service Locator patternu

```
IBLAdapter blAdapter =
    HTMLBLServiceLocator.getBLAdapter("TS1", null);
REQ_BL_TS1_ROL_Create rolex = new REQ_BL_TS1_ROL_Create();
rolex.setId(rolePol);
RES_BL_TS1_POL_Base resRolex = (RES_BL_TS1_POL_Base)
    blAdapter.execute(rolex);
```

33

SOFTEC

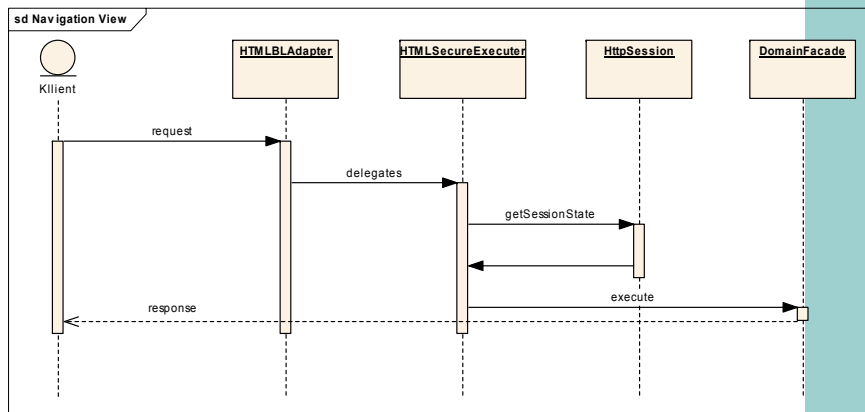
Prístup k aplikačnej logike



34

SOFTEC

Prístup k aplikačnej logike



35

SOFTEC

BE adaptéry

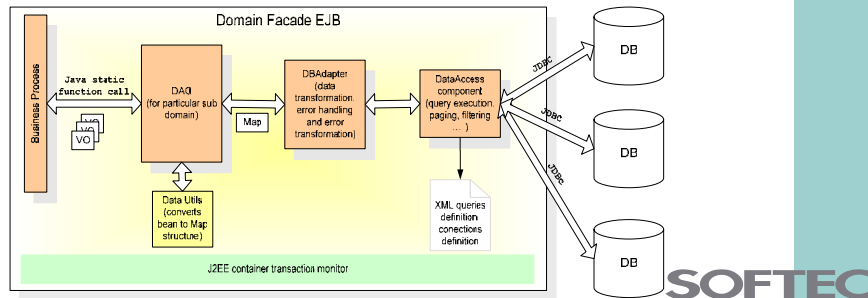
- BE Adaptéry zodpovedajú za:
 - transformáciu vstupných dát do formy akceptovanej BE,
 - transformáciu odpovede z BE do formy akceptovanej BL,
 - mapovanie BE chýb na chyby BL v zhode s chybovým frameworkom
- Implementácie adaptérov:
 - Prístup k databáze (DAO)
 - WS adaptéry:
 - Autorizačné služby
 - Notifikačné služby (Email, SMS),
 - Tlačové služby (CTS - centrální tiskové služby).

36

SOFTEC

DAO (prístup k DB)

- Data Access Object:
 - návrhový vzor - BP nekomunikuje priamo s DB – používa DAO objekty
 - DAO objekty používajú adaptér na DB komponent „Data access“
 - podpora na prácu s value objektami



Frameworkový prístup

- aplikačná architektúra na mieru
 - Končí snahou o univerzalizáciu
- univerzálna aplikačná architektúra
 - Postupne nabaľuje špecifické časti
- recept na dlhú životnosť riešenia
 - Použitie paternov pokiaľ je to možné
 - Používanie najsilnejšej stránky Javy – inerfejs
 - Umožňuje zamieňať implementácie
 - Budovanie otvorenej architektúry

38

SOFTEC

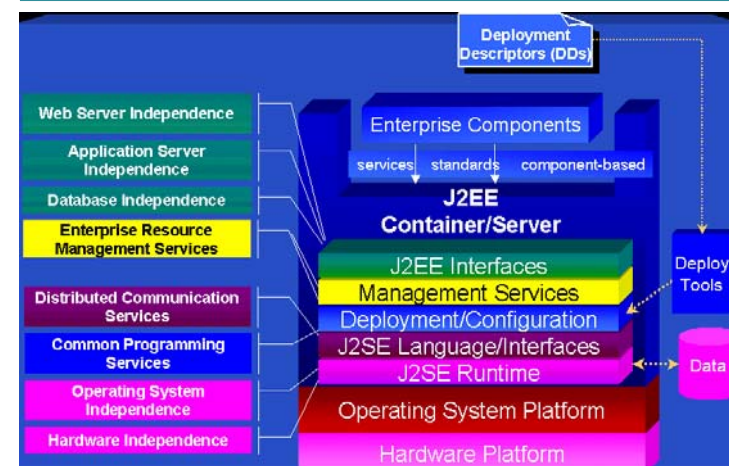
Frameworkový prístup

- Použitie Existujúcich Open Source frameworkov
 - Možnosť ich redukcie/rozšírenia vzhľadom na požadované typy riešení
 - Kedy uvažovať „customizácii“? a kedy o vlastnom riešení
- Výstavba aplikačnej logiky v praxi
 - charakteru aplikácie
 - akolie aplikácie
 - atomizácia aplikačnej logiky
 - stupeň previazanosti s analytickým riešením a jeho transformácie do aplikačnej infraštruktúry
- Perzistencia a komunikácia s backend systémy
 - DAO
 - Entity EJB
 - Objektovo relačné mapovadlá
 - Backend adaptéry

39

SOFTEC

J2EE zhrnutie III.



40

SOFTEC

J2EE a .NET

- .NET a vplyv na J2EE
 - Ako začať v J2EE a ako v .NET
 - Prístup k aplikačnému vývoju
 - Tvorba jednoduchých web aplikácií
 - Viacvrstvové architektúry
 - Zložité - na mieru šité architektúry
 - SOA
 - Možnosti prispôsobovania aplikačnej architektúry
 - Podpora a otvorenosť
 - Množstvo a dôveryhodnosť zdrojov
 - JEE 5.0 v vs. ".NET 3.0"

41

SOFTEC

J2EE a .NET

- Možnosti a obmedzenia .NET a J2EE
 - Architektonické/výkonnostné limity vs. spôsob, miesto, spoľahlivosť, odolnosť a robustnosť použitia v jednotlivých platformách
 - J2EE - vytvorenie aplikačnej architektúry (založenej na štandardoch) na mieru
 - Zložitosť
Návrh/Vývoj/Prevádzkovanie/Údržba/Rozširovanie riešení
 - Vplyv operačných systémov, HW....a jeho možnej zmeny na aplikáciu

42

SOFTEC

Užitočné zdroje

- J2EE špecifikácie
 - <http://java.sun.com/reference/docs/index.html>
 - <http://java.sun.com/reference/api/index.html>
 - <http://java.sun.com/javase/5/docs/tutorial/doc/>
- J2EE design prehľad:
 - http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2e/DEA2eTOC.html
 - http://java.sun.com/blueprints/guidelines/designing_enterprise_applications/apmTOC.html
 - <http://java.sun.com/reference/blueprints>
 - <http://martinfowler.com/>
- Ako budovať Web MVC aplikačný framework:
 - http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2e/web-tier/web-tier5.html#1078527
 - <https://bpcatalog.dev.java.net/nonav/webtier/index.html>
- SOA – prístupy k SOA architektúre
 - <http://java.sun.com/reference/soa/webservices/index.html>
 - <https://blueprints.dev.java.net/servlets/ProjectDocumentList?folderID=3401&expandFolder=3401&folderID=4743>
 - <http://java.sun.com/developer/technicalArticles/WebServices/index.html>
- Aplikačné serveri – nie len J2EE
 - <http://www.devx.com/enterprise/html/16268>
- Základné použitia J2EE technológií:
 - <http://interval.cz/vyvoj-aplikaci/java/>
 - http://my.execpc.com/~gopalan/java/java_tutorial.html
- Iné <http://www.w3.org/TR/>

43

SOFTEC

• **Ďakujem
Vám za
pozornosť.**

• **Otázky ?**

• **Kontakt:**
peter.obert@softec.sk



44

SOFTEC