

foi

Suvremene arhitekture IS-a i njihova primjenjivost u javnoj upravi

Pregled predavanja

foi

- Uvod u informacijske servise javne uprave
 - Osnovno o arhitekturama informacijskih sustava
 - Od ranih do suvremenih arhitektura IS-a
 - Moderne arhitekture – SOA, Cloud, SaaS, IaaS...
 - Implementacija i primjena ICT servisa javne uprave
- Primjenjivost suvremenih arhitektura u javnoj upravi
 - Općenito o razini primjenjivosti
 - Modeli zrelosti
 - Metodologija
- Zaključak

foi

Suvremene arhitekture IS-a ...

Zašto informacijsko-komunikacijska tehnologija i javna uprava?

foi

gradani online
uštedite vaše vrijeme i povećajte naš online korisnik

Detaljno na naš portal elektroničkih usluga koje želimo pružiti: kvalitetno i brzo servisiramo građane u poslovnom subjektima. Uvid u podatke javnih registara putem interneta omogućuje Vam da vidite ažurne podatke koje gradsko uprava stvara u svom svakodnevnom radu. Taj uvid možete napraviti u bilo koje vrijeme, iz svake druge lokacije, bez fizičkog dolaska u ured gradske uprave.

Poštete svoje usluge koje smo omogućili su javne usluge građanima kao što su: uvid u podatke o radnom mjestu, pristupni pregled Demografskog urbariološkog plana te vlastite usluge kojima se radi o radnom mjestu i radu osoba s radnim mjestom na kojem gledate, u kojem mjestu imate svoj usmeni uvid u državne, matične isprave, vanjskih te uvid u druge računa komunalnih materijala (usluge) i tako je to početak...

Ovim projektom želimo promovirati prednosti elektroničkog poslovanja (poslovanje općenito javno) i graditi na korištenje interneta u svakodnevnom radu i životu. Prednost uvođenja novih informacijskih tehnologija su: olakšavanje i brzo na teret, transparentnost, efikasnost i ne manje važno, jačanje vjere u javne službe.

Dinamično razvoje i proširivanje usluga koje će se naći na ovom portalu u budućnosti, vrši se o prilagodbi programa koji reguliraju podatke i sposobnosti i o vrstama elektroničkog dokumenta i digitalnog potpisa u praksi.

Veći smo da naš informacija kao što značajni i pouzdanost usluga koje se nam otkrivaju suptilno i praviše za nove usluge koje imate predložene.

www.zagreb.hr

4

Zašto informacijsko-komunikacijska tehnologija i javna uprava?

foi

- **Gov.hr** - Središnji državni portal;
- **NIAS** - Nacionalni identifikacijski i autentifikacijski sustav;
- **OKP** - Osobni korisnički pretinac



<http://www.uprava.hr/default.aspx?id=13899>

5

Zašto informacijsko-komunikacijska tehnologija i javna uprava?

foi

- **Elektronička razmjena podataka** (Electronic Exchange of Social Security Information – EESSI)
- **Informacijska mreža javnog zdravstva Europske unije** (European Union Public Health Information Network – EUPHIN)
- **Europski napredni višezječni informacijski sustav** (European Advanced Multilingual Information System – EURAMIS)
- **Mreža za neispitane prehrambene proizvode i prehrambene sastojke** (Novel Foods and Food Ingredients Network – NF-NET)
- **Informacijsko-komunikacijski sustav za upravljanje uvoznim carinskim kvotama u poljoprivredi** (The Agricultural Market Information System Quota – AMSI/QUOTA)
- **Baza podataka o podrijetlu i registraciji proizvoda koja podupire poljoprivrednu politiku kvalitete proizvoda** (Database of Origin and Registration – DOOR)
- **Europska mreža za civilnu zaštitu i hitne slučajeve vezane uz okoliš** (Civil Protection and Environmental Emergencies European Network – PROCIENET)
- **Mreža financijskih obavještajnih jedinica** (Financial Intelligence Units Network – FIU.NET)
- **Baza podataka o prometnim nesrećama u zajednici** (Community Road Accident Database – CARE)
- **Europska baza podataka otisaka prstiju tražitelja azila** (European Dactiloscopy – EURODAC)
- **Informacijski sustav protiv prijevара** (Anti Fraud Information System – AFIS)
- **Telematička mreža za razmjenu informacija o izdavanju tahograf kartica** (Telematics Network for the Exchange of Information Concerning the Issuing of Tachograph Cards – TACHONET)
- **Informacijski sustav za kontrolu eksplozivnih sredstava te borbu protiv terorizma i njegovu prevenciju** (Pan-European System of Explosives Control to Prevent and Fight against Terrorism – SCEPT)
- **Europski informacijski sustav prometnih i vozačkih dozvola** (European CAR and driving licence Information System – EUCARIS)
- **Sustav suradnje za zaštitu potrošača** (Consumer Protection Cooperation System – CPCS)
- **Integrirani sustav za upravljanje dozvolama za uvoz i izvoz** (Integrated System for Managing Exports and Import Licences – SIGL)
- **Zajednički sustav za veze i informiranje** (Common Emergency Communication and Information System – CECS)
- **Mreža za sigurno more** (Safe Sea Net – SSN)

6

Informacijski sustav...

foi

- ...je sustav koji prikuplja, pohranjuje, čuva, obrađuje, i isporučuje potrebne podatke na način da su dostupni svim članovima neke organizacije koji se njima žele koristiti te imaju odgovarajuću autorizaciju.
- ...je dio poslovnog sustava koji daje podatkovnu sliku procesa iz realnog sustava. To vrši modelom podataka, modelom procesa i modelom izvršitelja.
- Elementi IS-a:
 - HW
 - SW
 - LW
 - DW
 - OW
 - NW...

7

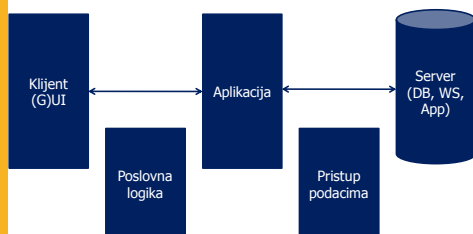
Osnovno o arhitekturama

foi

- Arhitektura informacijskog sustava...
 - ...predstavlja formalnu definiciju poslovnih procesa i pravila, strukture poslovnog sustava, tehničkog okvira i tehnologije poslovnog informacijskog sustava.
 - ...je opis dizajna, elemenata i njihovih međusobnih odnosa informacijskog sustava.
 - ...obuhvaća kombinaciju hardverskih i softverskih elemenata pomoću kojih se krajnjem korisniku pruža određena usluga ili proizvod.
- Arhitektura informacijskog sustava uobičajeno obuhvaća 4 sloja:
 - Arhitekturu poslovnih procesa
 - Sistemsku arhitekturu,
 - Tehničku arhitekturu, i
 - Arhitekturu proizvodnog/ uslužnog sustava.

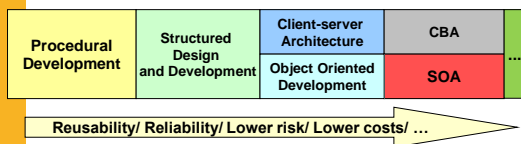
Osnovni elementi arhitekture IS-a

foi



Pristupi razvoju IS-a (1)

foi



Pristupi razvoju IS-a (2)

foi

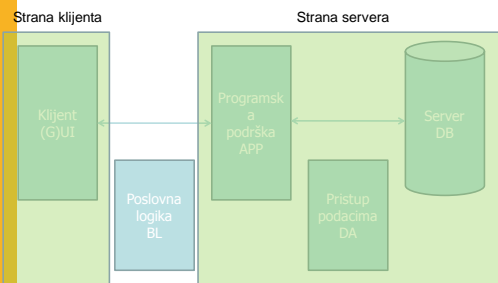
- Proceduralni razvoj
 - Velik broj potencijalnih ovisnosti između programskih algoritama, složeno održavanje i modifikiranje programskih rješenja
- Strukturni dizajn
 - Mali programski moduli koriste se kako bi se smanjila složenost programskog rješenja (divide-et-impera), te poboljšala razina njihove ponovne iskoristivosti.

Pristupi razvoju IS-a (3)

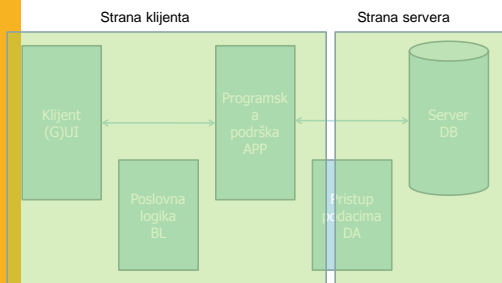
foi

- Razdvajanje programskih komponenti
 - Pojednostavljuju se, i uklanjaju snažne veze između programskih komponenti.
- Klijent-server arhitektura
 - Razdvajanje glavnih arhitekturnih slojeva
- Objektno orijentirani razvoj
 - Enkapsulacija podataka i ponašanja kroz apstraktne tipove podataka – klase

Arhitektura IS-a
 "tanki" klijent – "debeli" server



Arhitektura IS-a
 "debeli" klijent – "tanki" server

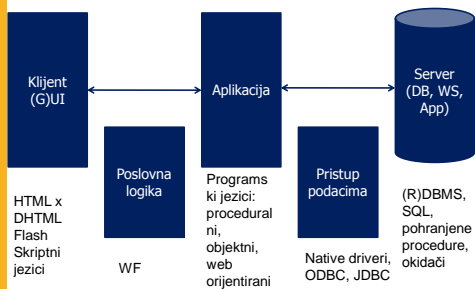


Arhitektura IS-a
 client side – server side (CSSS)



Osnovni elementi arhitekture IS-a

foi



Pristupi razvoju IS-a (4)

foi

- Moderne arhitekture IS-a
- Arhitektura temeljena na komponentama (CBA)
 - Programski moduli složeniji od objekata, sa definiranim programskim sučeljima i kontekstualnim odnosima.
 - Komponente su nezavisne.
 - Skup međusobno povezanih komponenti čini funkcionalnost aplikacije.
- Distribuirano računarstvo
 - Različiti dijelovi programa paralelno se izvode na dva ili više računala koja su međusobno povezana putem računalne mreže.
- SOA, Cloud, SaaS...

foi

SOA

Service Oriented Architecture

foi

Servisno orijentirane arhitekture VS Uslugama orijentirane arhitekture

19

SOA

foi

- Arhitektura temeljena na servisima
 - Princip dizajniranja softvera
- Oslanjanje na servise koji su dostupni na računalnim mrežama (intranet, internet)
- Servisi i korisnici koji ih koriste su međusobno neovisni (bez povezanosti karakteristične za objekte)
- Implementacija SOA-e podrazumijeva:
 - Razvoj aplikacija koje koriste servise, i/ili
 - Razvoj aplikacija koje se koriste kao servisi

20

Zašto SOA?

foi

- **PROBLEM:** pisanje programa postaje kompleksnije
 - nedovoljno iskorištavanje postojećeg koda
- **RJEŠENJE I:** modularno programiranje
 - Copy/Cut/Paste modula u različite aplikacije
 - Problemi sa održavanjem
- **RJEŠENJE II:** objektno programiranje
 - Rast kompleksnosti i opsega aplikacija
 - Razvoj i održavanje softvera je kompleksno
 - Bolje iskorištavanje funkcionalnosti
- **RJEŠENJE III:** Arhitektura temeljena na komponentama (CBA)
 - poboljšava iskoristivost
 - lakše i jednostavnije održavanje softvera
 - funkcionalnost cjeline podijeljena na manje funkcije
 - funkcije su enkapsulirane u komponente

21

Zašto SOA?

foi

- IS prelazi granice poslovnog sustava
- *Enterprise Application Integration*
- PROBLEMI integracije aplikacija uz:
 - Različitos platformi
 - Različitos protokola
 - Različitos uređaja
 - Distribuirani softver

22

SOA – definicije (1)

foi

- SOA je informacijska i komunikacijska (ICT) arhitektura koja pruža fleksibilnost potrebnu za implementiranje elemenata poslovnog procesa i postavljanje potrebne ICT infrastrukture u obliku sigurnih, standardiziranih komponenti (servisa) koje se mogu višestruko koristiti i međusobno kombinirati kako bi zadovoljile različite poslovne potrebe.

23

SOA – definicije (2)

foi

- SOA je ICT arhitektura koja se temelji na labavoj povezanosti (*loose coupling*), višestrukoj iskoristivosti (*reuse*) i interoperabilnosti između različitih programskih i organizacijskih sustava.

24

SOA – definicije (3)

foi

- SOA je aplikacijska arhitektura u kojoj su sve funkcije i usluge (servisi) definirani pomoću **standardiziranog jezika** (*description language*) te posjeduju **pristupna sučelja** pomoću kojih se pozivaju s ciljem potpore određenim segmentima poslovnih procesa. Svaki pristup i interakcija između servisa je neovisna od ostalih interakcija i komunikacijskih protokola.

25

SOA – definicije (4)

foi

- SOA je arhitektura IS-a u kojoj su aplikacijske funkcije realizirane kao komponente (servisi) koji su labavo povezani (*loosely coupled*) i precizno definirani kako bi omogućili interoperabilnost, fleksibilnost i ponovnu iskoristivost.

26

SOA – definicije

foi

- Servisno orijentirane arhitekture predstavljaju **okvir za integraciju poslovnih procesa i odgovarajuće ICT infrastrukture u komponente** (servise, usluge) koje se mogu **višestruko koristiti** i međusobno kombinirati kako bi odgovorili zahtjevima poslovanja u dinamičnom okruženju.

27

Karakteristike SOA

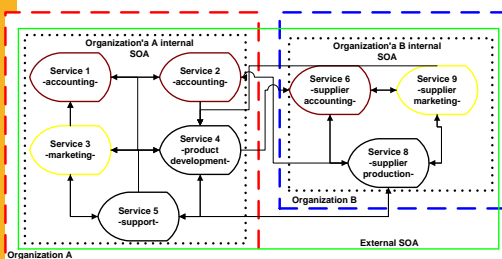
foi

- Interoperabilnost
- Raspoloživost i fleksibilnost
- Heterogenost
- Sigurno upravljanje komponentama

28

SOA – primjer

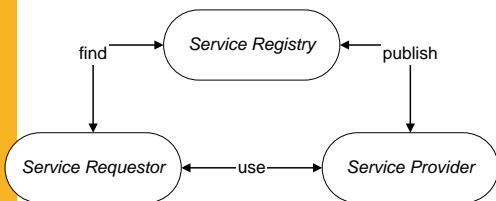
foi



29

SOA – osnovni model

foi



30

SOA – tehnički model

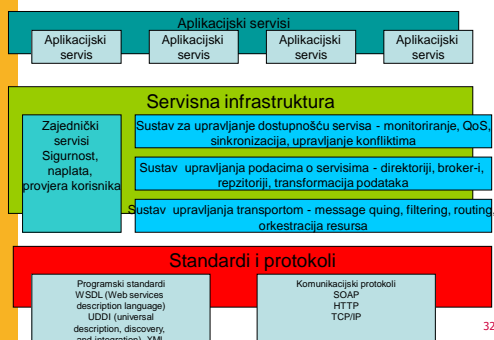
foi

5. SLOJ PRISTUPA/ PREZENTACIJE	6. INTEGRACIJA	7. OSIGURANJE KVALITETE USLUGA, SIGURNOST, UPRAVLJANJE NADZOR
4. SLOJ KOREOGRAFIJE		
3. SLOJ SERVISA		
2. SLOJ KOMPONENTI		
1. SLOJ OPERACIJSKOG SUSTAVA		

31

SOA – Arhitektura web servisa

foi



32

SOA - Pojmovi

foi

- **Servis:**
 - Poslovna funkcija koju obavlja davatelj usluga (*service provider*) za potencijalnog korisnika (*service requestor*)
 - Primjer: obrada narudžbe, konvertiranje valuta, obrada plaća...
- Servis vs servis?

33

SOA – Pojmovi

foi

- Servisi – svojstva:
 - **Sučelje** (WSDL) preko kojeg se servis koristi (poziva) je **neovisno o platformi**
 - Opis servisa sadrži:
 - parametre, ograničenja, svrha, način korištenja, sigurnosni protokoli koji se moraju koristiti
 - Servis se može dinamički locirati i pozvati
 - Servis je samostalan i neovisan o stanjima drugih servisa, procesa i funkcija (*stateless*)
 - *Loose coupling*

34

SOA - Pojmovi

foi

- Otkrivanje servisa (*Service Discovery*):
 - SOA se oslanja na mogućnost identificiranja (otkrivanja) servisa i njihovih mogućnosti
 - postojanje direktorija koji opisuje servise dostupne u svojoj domeni
- *Directory Service* (*Service Registry, UDDI*)
 - Posrednik između korisnika i providera
 - Organizacija i kategorizacija servisa na temelju nekih kriterija

35

SOA – komponente

foi

- Oglašavanje servisa:
 - Opis servisa koji je dostupan potencijalnim korisnicima
 - Metode oglašavanja:
 - *Pull* – potencijalni korisnik šalje zahtjev za opisom servisa (*kupujem!*)
 - *Push* – provider šalje opis usluge potencijalnim korisnicima (*prodajem!*)

36

SOA – komponente

foi

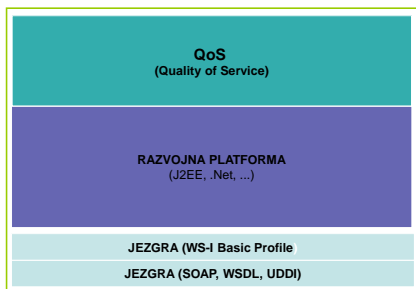
- **Povezivanje (Binding):**
 - Odnos između providera i korisnika je dinamički
 - Odnos se uspostavlja pomoću mehanizma za povezivanje na servis u toku interakcije
- **Poruke – temelj komunikacije**
 - Pružatelji usluga i korisnici komuniciraju izmjenom poruka
 - Tehnologija korištena za definiciju poruka mora biti neovisna o platformi (npr. XML)

40

SOA – IT infrastruktura

foi

Tipičan primjer SOA IT infrastrukture:



41

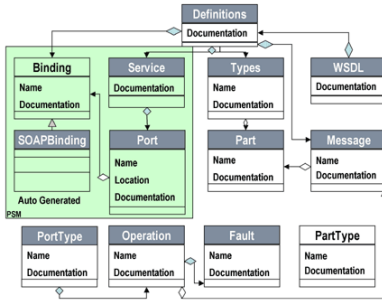
SOAP, WSDL, UDDI

foi

- **WSDL (Web Services Description Language)**
 - Jezik za opis servisa (XML dokument)
 - Specificira što servis radi, potrebne formate podataka i protokole, lokaciju servisa
- **UDDI (Universal Description, Discovery and Integration)**
 - Registracija i pretraživanje servisa
 - Omogućuje oglašavanje i pretraživanje servisa
 - Pristup WSDL dokumentima
 - Neovisan o platformi
- **SOAP (Simple Object Access Protocol)**
 - Protokol temeljen na XML-u koji omogućuje slanje poruka između providera i korisnika
 - Koristi npr. HTTP, SMTP, MIME
 - Neovisan o platformi

42

Figure 2 Stereotypes in the WSDL Profile



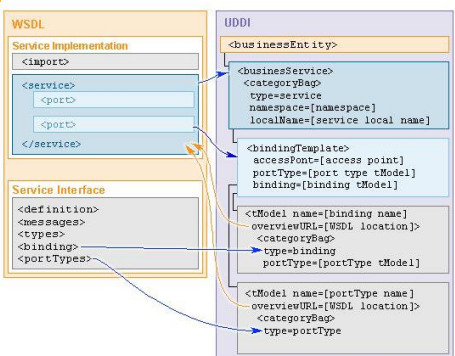
43

```

<?xml-stylesheet type="text/css" href="http://schemas.xmlsoap.org/wsdl/2003/XMLSchema.xsd" />
<!--
WSDL created by Apache Axis version: 1.3.1
Built on Jun 14, 2008 (09:18:07 EDT)
-->
<definitions targetNamespace="http://ib.e.douglas.net/2003/axis/DataService.wsdl"
  xmlns:tns="http://ib.e.douglas.net/2003/axis/DataService.wsdl"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <message name="getDataSource" type="xsd:string"/>
  <message name="getDataSourceResponse" type="xsd:string"/>
  <portType name="DataService"
    <operation name="getDataSource"
      <input message="http://ib.e.douglas.net/2003/axis/DataService.wsdl:getDataSource"/>
      <output message="http://ib.e.douglas.net/2003/axis/DataService.wsdl:getDataSourceResponse"/>
    />
  <binding name="DataServiceSoapBinding" type="soap:binding"
    <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"
    />
    <operation name="getDataSource"
      <input message="http://ib.e.douglas.net/2003/axis/DataService.wsdl:getDataSource"
      <output message="http://ib.e.douglas.net/2003/axis/DataService.wsdl:getDataSourceResponse"
      />
    />
  <port name="getDataSource"
    <binding href="#DataServiceSoapBinding"
    />
  />
  <service name="DataService"
    <port binding="http://schemas.xmlsoap.org/soap/envelope/#DataServiceSoapBinding"
      name="DataService"
      location="http://ib.e.douglas.net/2003/axis/DataService.wsdl"
    />
  />
</definitions>

```

44



45

QoS - *Quality of Service*

foi

- Postojanje naprednijih zahtjeva za kritične sustave:
 - Sigurnost: sigurnost poruke (integritet, autentičnost, povjerljivost)
 - Pouzdanost: osigurati isporuku poruke, izvješće o isporuci
 - Zahtjevi vezani uz transakcije
 - Politika komunikacije: provideri ponekad zahtijevaju određenu politiku ponašanja (npr. Kerberos)
 - Management: infrastruktura za administratore

46

QoS - *Quality of Service*

foi

- Specifikacije vezane uz QoS izdane su od strane navedenih tijela za standarde:
 - W3C (World Wide Web Consortium)
 - OASIS (Organization for the Advancement of Structured Information Standards)

47

SOA – integracija komponenti

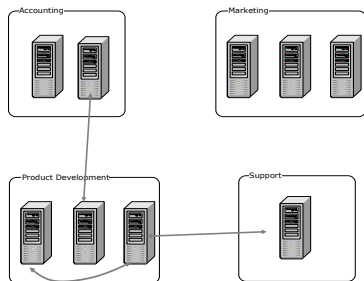
foi

- Enterprise Application Integration
 - Princip objedinjavanja SW, HW i ostalih komponenti IS-a
- CILJ: povećati učinkovitost poslovanja kroz povezivanje i optimizaciju IS-a
- EAI treba ukloniti uska grla u razmjeni informacija (podataka) između različitih podsustava IS-a

48

Otoci automatizacije (islands of automation)

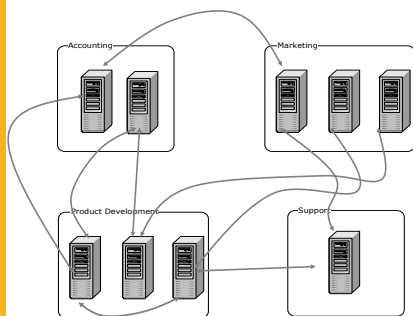
foi



49

Point to point povezivanje

foi



50

EAI razine

foi

- Integracija podataka
 - Pristup podacima, obrada podataka, prosjeđivanje podataka
- Integracija na razini aplikacija
 - Višestruka iskoristivost aplikacija, API
- Integracija na razini objekata
 - CORBA, DCOM
- Integracija korisničkih sučelja
 - Dojam cjelovitog i integriranog sustava
- EAI implementacija
 - Queues

51

EAI razine

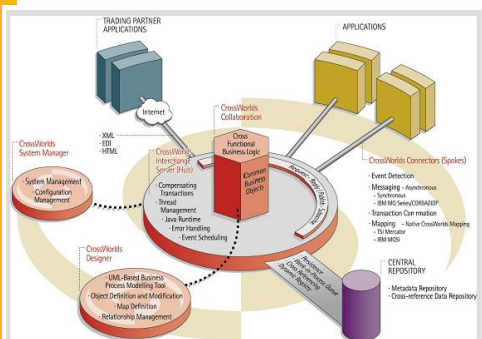
foi

- *EII – Enterprise Information Integration*
 - Abstrakti podataka
 - Heterogenost i kontekstualizacija podataka
- Integracija poslovnih procesa
 - Povezivanje poslovnih procesa kroz aplikacije
- Integracija temeljena na poslovnim pravilima
 - Implementacija poslovnih pravila uklanja ovisnost o platformi i aplikaciji

52

Hub and spoke

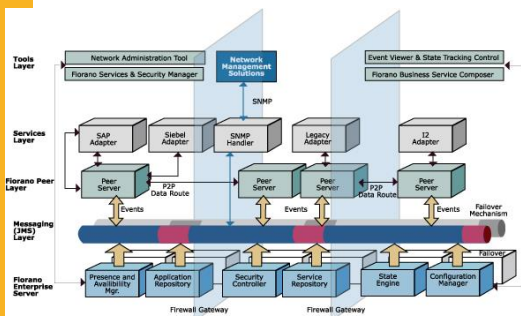
foi



53

Enterprise Service Bus

foi



ESB

foi

- Middleware
- Povezivanje podsustava i aplikacija:
 - Objedinjavanje različitih protokola i sustava
 - Povezivanje sučeljima
- Posredovanje:
 - Routing, modeli interakcije, upravljanje verzijama
 - Transformacije i mapiranje, upravljanje razmjenom poruka
 - orkestracija
- Upravljanje i nadzor:
 - Upravljanje porukama
 - Upravljanje transakcijama
 - Sigurnost, QoS
 - Upravljanje i nadzor servisa

55

ESB - integracija

foi

- Skup *service containera* koji objedinjavaju različite vrste IT resursa
- Kontejneri su međusobno povezani sustavom razmjene poruka (*secure message queing bus*)
- Kontejneri prilagođavaju IT resurse prema standardiziranom servisnom modelu (komunikacija XML porukama – *XML message exchange patterns*)

56

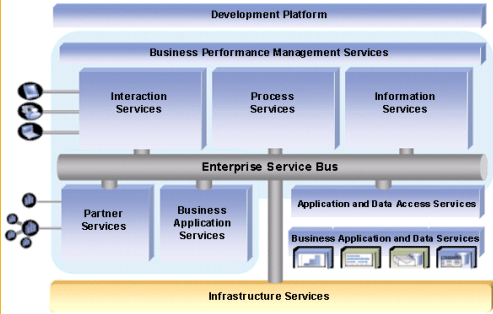
ESB - integracija

foi

- Središnja administracija i nadzor distribuiranog sustava upravljanjem i razmjenom poruka (*message transformation services, message routing services*)
- Omogućuje *ad-hoc* promjene u sustavu, bez potrebe za narušavanjem funkcionalnosti sustava

57

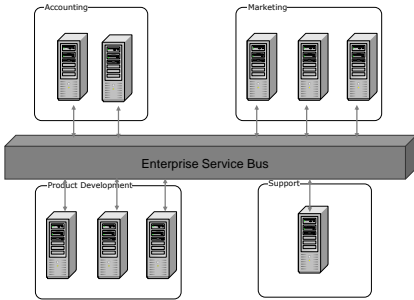
foi



58

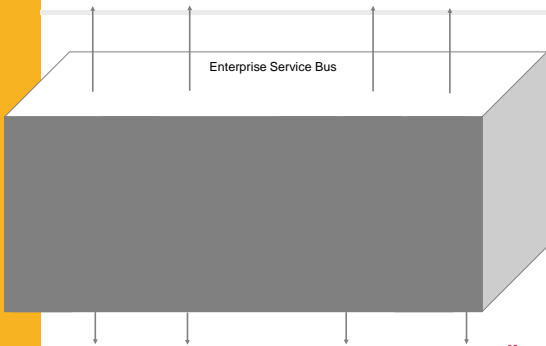
Integracija ESB-om

foi



59

foi



60

ESB prednosti

foi

- Jedinствeni okvir za integracijska rješenja
- Jednostavno povezivanje podsustava IS-a
- Sigurnost, pouzdanost

61

ESB nedostaci

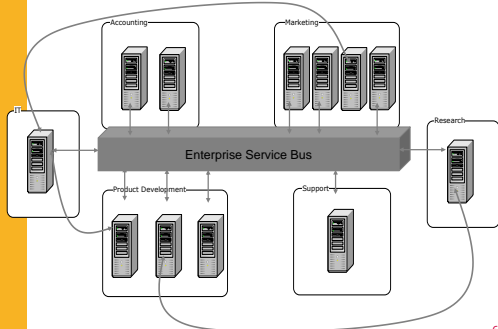
foi

- Nered je spremljen ispod tepiha
- Nedosljednost u primjeni ESB-a pogoršava stanje u odnosu na situaciju bez ESB-a
- Ovisnost o ESB-u
- Nemogućnost jednostavne zamjene ESB platforme
- Ovisnost o ESB provideru

62

Razvoj sa ESB-om

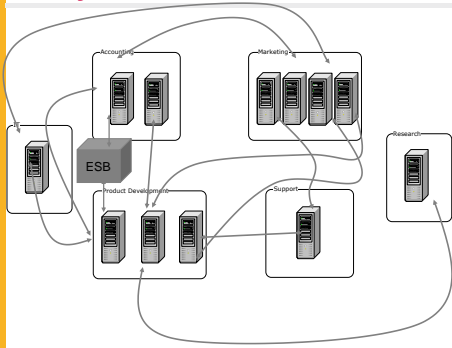
foi



63

Razvoj sa ESB-om

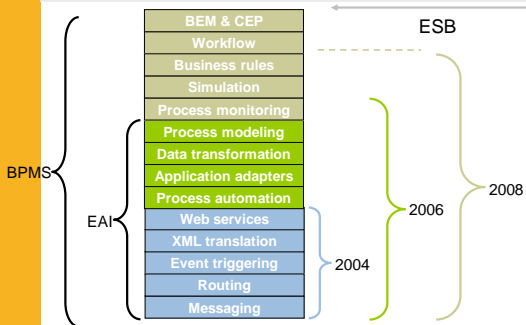
foi



64

ESB i EAI

foi



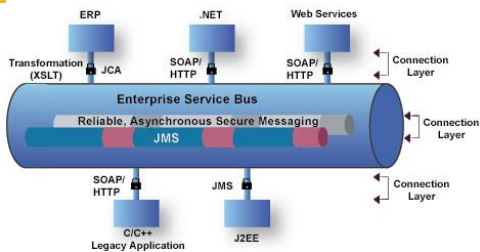
65

Business Domain Specific extensions	Various	Business Domain
Distributed Management	WSDM, WS-Management	Management
Provisioning	WS-Provisioning	
Security	WS-Security	Security
Security Policy	WS-SecurityPolicy	
Secure Conversation	WS-SecureConversation	
Trusted Message	WS-Trust	
Federated Identity	WS-Federation	
Portal and Presentation	WSRP	Portal and Presentation Transactions and Business Process
Asynchronous Services	ASAP	
Transaction	WS-Transactions, WS-Coordination, WS-CAF	
Orchestration	BPEL4WS, WS-CDL	Messaging
Events and Notification	WS-Eventing, WS-Notification	
Multiple message Sessions	WS-Enumeration, WS-Transfer	
Routing/Addressing	WS-Addressing, WS-MessageDelivery	
Reliable Messaging	WS-ReliableMessaging, WS-Reliability	
Message Packaging	SOAP, MTOM	Metadata
Publication and Discovery	UDDI, WSIL	
Policy	WS-Policy, WS-PolicyAssertions	
Base Service and Message Description	WSDL	
Metadata Retrieval	WS-MetadataExchange	

66

ESB - tehnologija

foi



67

Microsoftova rješenja ESB

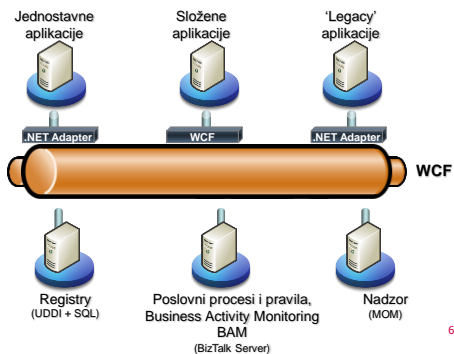
foi

- Windows Communication Foundation
 - WCF, *Indigo*
 - Komunikacijski podsustav
 - Veza između .Net baziranih aplikacija
- BizTalk server
- SQL server
- Microsoft Operations Manager
 - Upravljanje događajima i performansama sustava
 - Nadzor integrirane okoline
- UDDI

68

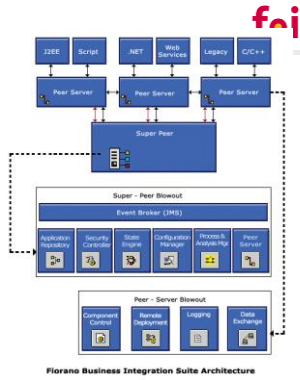
Microsoftova rješenja ESB

foi

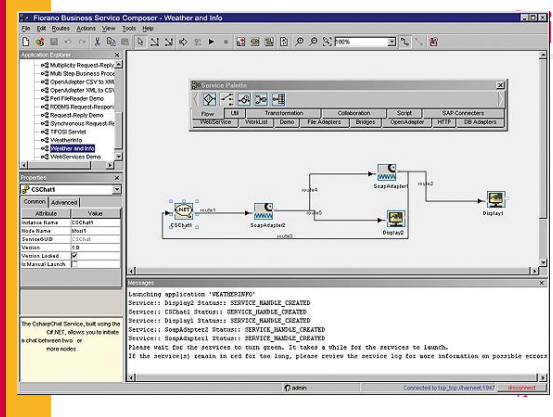


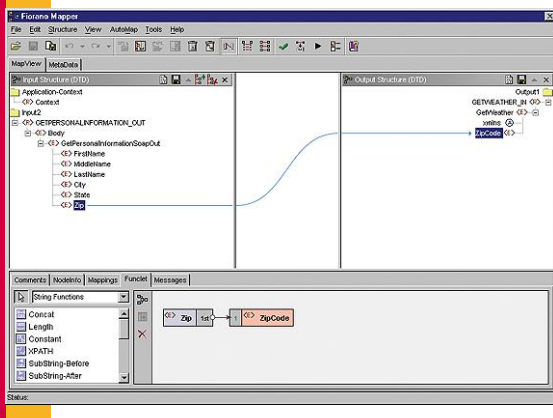
69

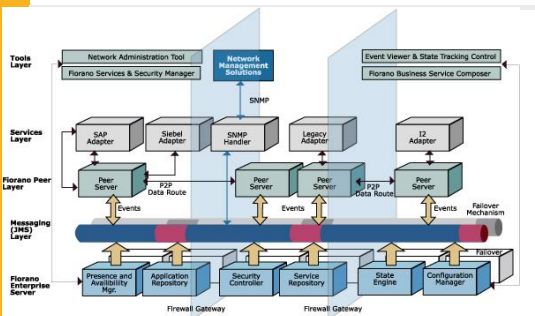
orano ESB



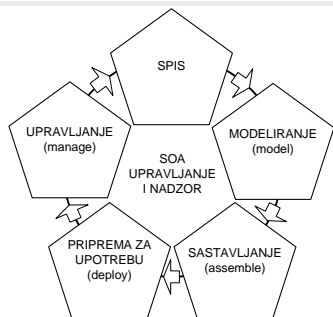
Fiorano Business Integration Suite Architecture







Razvoj SOA



SOA i ostale tehnologije

- SOA vs. web servisi
 - SOA definira "strategiju" kako se određeni proces treba provesti, a web servis definiraju kako će se to u stvarnosti i realizirati.
 - web servis je jedna od mogućih tehnika kojima se SOA može implementirati
- SOA vs. CORBA
 - Aplikacije i komponente temeljene na CORBA arhitekturi imaju niz karakteristika koje ih čine pogodnima i omogućuju njihovo povezivanje i interakciju sa SOA komponentama.
 - WSDL ~ IDL (*Interface Description Language*)

SOA i ostale tehnologije

foi

- SOA vs. MDA (*Model Driven Architecture*)
 - MDA: za svaki programski proces može se definirati model, za koji je moguće realizirati odgovarajuću programsku komponentu u aplikaciji.
 - MDA za poslovne procese izrađuje modele, koji se "pretvaraju" u programske komponente, a čijim prevođenjem i izvođenjem se dobiva programski proizvod koji se može koristiti na različitim platformama
 - MDA ~ BPEL
- SOA vs. GRID
 - *grid* predstavlja okvir u kojem su implementirani svi, ili samo neki, dijelovi SOA sustava – SOA je korisnik *grida*
 - *grid* može biti izgrađen na temelju SOA - svaka komponenta (servis) predstavlja element koji je sastavni dio *grida*

79

Prednosti SOA

foi

- mogućnost brzog prilagođavanja promjenama
- horizontale promjene – prilagođavanje WSDL sučelja je dovoljno za njihovu međusobnu povezanost i interakciju
- vertikalne promjene – izrada IS-a po mjeri
- iskorištavanje *legacy* aplikacija
- smanjen rizik uvođenja novih grešaka u procesu poboljšanja ili izrade novih poslovnih usluga
- bolje korištenje postojeće infrastrukture
- smanjuje se krivulja učenja za razvojni tim

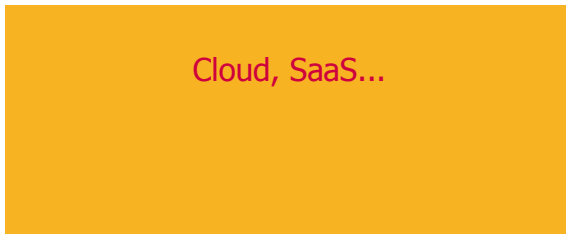
80

SOA danas...

foi

- Simple Object Access Protocol (**SOAP**), Web Services Description Language (**WSDL**), Universal Discovery Description and Integration (**UDDI**), XML Process Definition Language (**XPDL**), Web Services Flow Language (**WSFL**), Business Process Modelling Language (**BPML**), Business Process Execution Language for Web Service (**BPEL4WS**), Web Services Choreography Description Language (**WS-CDL**), Secure Sockets Layer (**SSL**), Security Assertion Markup Language (**SAML**), Electronic Business Service Oriented Architecture (**ebSOA**), **ebXML**, Framework for Web Services Implementation (**FWSI TC**), open Building Information Exchange (**oBIX TC**), Universal Business Language (**UBL**), **WS-Addressing**, **WS-AtomicTransaction**, WS Composite Application Framework (**WS-CAF**), WS Coordination Framework (**WS-CF**), **WS-Choreography**, Web Services Choreography Description Language (**WS-CDL**), WS Context (**WS-CTX**), WS Distributed Management (**WSDM**), WS Inspection Language - WS-Inspection (**WSIL**), WS-Resource Framework (**WS-RF**), Web Services Security (**WSS**), eXtensible Access Control Markup Language (**XACML**), XML signature, XML encryption...

81



Cloud, SaaS...

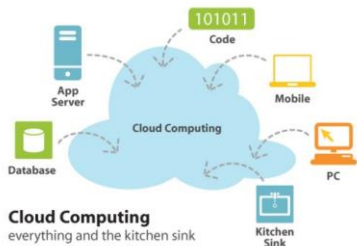
S. Gerić, Supporting eBusiness/eGov using modern IS Architectures - 2011

Cloud Computing (1)

- Sljedeći evolucijski korak u razvoju ICT usluga na zahtjev (*on-demand*) temeljenih na virtualnim ICT resursima.
- Listopad 2007, „Blue Cloud“ IBM&Google
- Pretpostavlja SOA, smanjeni ICT overhead za krajnje korisnike, veliku fleksibilnost, smanjenje troškova vlasništva (TCO), usluge na zahtjev...
- Integracija aktivnosti temeljnih na servisima se ostvaruje kroz niz "hodograma" (workflow) – definira niz strukturiranih aktivnosti koje povezane čine funkcionalnost informacijskog sustava.

S. Gerić, Supporting eBusiness/eGov using modern IS Architectures - 2011

Cloud Computing (2)



S. Gerić, Supporting eBusiness/eGov using modern IS Architectures - 2011

Cloud Computing (3)

foi

- In the context of "cloud computing," the key questions is whether the underlying infrastructure is supportive of the workflow-oriented view of the world.
- This includes on-demand and advance-reservation based access to individual and aggregated computational and other resources, autonomies, ability to group resources from potentially different "clouds" to deliver workflow results, appropriate level of security and privacy...

S. Gerić, Supporting eBusiness/eGov using modern IS Architectures - 2011

Cloud Computing (4)

foi

- **Virtualization** is another key element of cloud.
- It allows abstraction and isolation of lower-level functionalities and underlying hardware, what enables portability of higher-level functions and sharing and/or aggregation of the physical resources.
- It has been applied to all aspects of computing – memory, storage, processors, software, networks, as well as services that IT offers.

S. Gerić, Supporting eBusiness/eGov using modern IS Architectures - 2011

Cloud Computing (5)

foi

- **Users** the key element of cloud.
- A broad set of user categories:
 - Cyberinfrastructure (CI) developers,
 - Developers (authors) of different component services and underlying applications,
 - technology and domain personnel that integrates basic services into composite services and their orchestrations (workflows) and delivers those to end-users, and
 - finally users of simple and composite services.

S. Gerić, Supporting eBusiness/eGov using modern IS Architectures - 2011

SaaS (1)

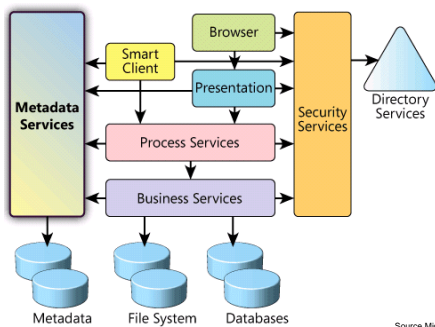
foi

- Software as a Service AKA Software on demand
- Further SOA/ Cloud evolution
- software that is deployed over the internet and/or is deployed to run behind a firewall on a local area network or personal compute
- SaaS Adoption
 - Level 1 – Ad-hoc/custom: Each customer has a customized version of the hosted application that runs as its own instance on the host's servers. Migrating a traditional non-networked or client-server application to this level of SaaS typically requires the least development effort, and reduces operating costs by consolidating server hardware and administration.
 - Level 2 – Configurable: This adds greater program flexibility through configurable metadata, so many customers use separate instances of the same application code. This lets the vendor meet different customer needs through detailed configuration options, while simplifying common code base maintenance and updating.
 - Level 3 – Configurable, multi-tenant-efficient: This adds multi-tenancy to the second level, so a single program instance serves all customers. This enables more efficient server resource use without apparent difference to the end user, but ultimately faces scalability limits.
 - Level 4 – Scalable, configurable, multi-tenant-efficient: The fourth and final SaaS maturity level adds scalability through a multitier architecture that supports a load-balanced farm of identical application instances that run on a variable number of servers. The provider can adjust system capacity to match demand by adding or removing servers without further altering the software architecture.

S. Gerić, Supporting eBusiness/eGov using modern IS Architectures - 2011

SaaS (2)

foi



Source Microsoft

S. Gerić, Supporting eBusiness/eGov using modern IS Architectures - 2011
