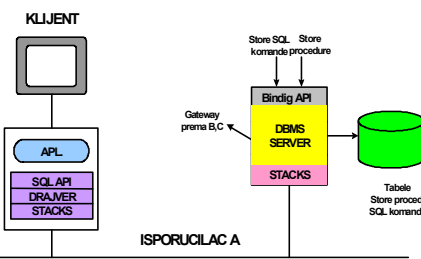
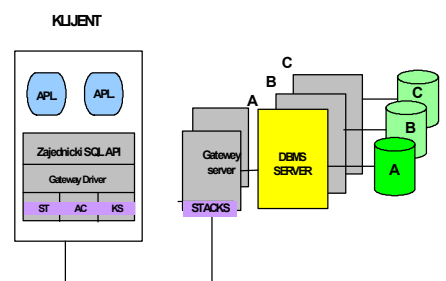


<p>114. Primeri otvorenih IDE-ova Besplatni otvoreni IDE-ovi(free/open source)-Besplatni više-platfomski IDE- Besplatni Windows IDE-Besplatni Linux/Unix IDE-Besplatni Java-bazirani IDE- Besplatni Macintosh IDEViše-platfomski IDE-:GNAT, Programming Studio, za Linux, MS Win, Solaris i jezike: Ada,C,C++,Python,-NAVJ, vizualni sw razvojno okruženje, alternativa za Delphi i VB-:Emaas i XEmacs, jedan od najpopularnijih IDE-:Lazarus, alternativa za Borland Delphi, baziran na Free Pascal-u - Windows IDE-:Dev-C++-SharpDevelop, otvoreni Net IDE-Magnum Open Studio - Linux/Unix IDE -KDevelop, na bazi GNU razvojnih alata-Anjuta, za GNOME developere-Gambas, dizajner na VB pod Linux-om-Open!Dev, grafički front-end za Linux razvojne alate-Motor, text bazirani IDE za C/C++ programiranje na Linux-u-Java-bazirani IDE -Eclipse, dizajner za Javu, ali proširiv-JGrasp, za: Java, Ada,VHDL, C, C++-Oracle iDeveloper, za razvoj Java, xml, web servisa, PL/SQL, PHP, -juni 2005-NetBeans, modularan, napisan 100% u Javi-TrueStudio Foundation, projekat, alati za skiping jezike, napravljen povrh Eclipse platforme-Macintosh IDE, sa pojavom Mac OS X, svi UNIX/Linux alati mogu raditi na Mac-u, prepostavka da je dignut X.11 server</p> <p>115. MS.NET okvir (Framework) -Obezbudeju širok skup pre-kodiranih rešenja za tipične programske zahteve i upravlja izvršavanjem programa napisanih specijalno za ovaj okvir. -Net Framework je komponenta MS Windows OS-Pre-kodirana rešenja formiraju biblioteku klasa okvira obuhvatajući-korisnički-servisni-pristup podacima-kriptografija-numerički algoritmi-mrežne komunikacije-Programi koriste funkcije iz biblioteka klasa, kombinujući ih u aplikacije.-Programi pisani za .Net okvir se izvršavaju u sw okruženju koje upravlja run-time zahtevima programa, tzv. CLR (Common Language Run-time)CLR obezbudeju izgled aplikacione VM, tako da se ne mora voditi računa o specifičnostima CPU na kome se program izvršava.- Biblioteka klasa i CLR zajedno čine. Net okvir.-Prvi put se pojavio 2002, m ože od Win. 98, Verzija 2.0 puštena Nov. 2005 je i danas aktuelna.-Net okvir je nezavisan od platforme, korišćenjem među jezika, slično kao i Java VM.-U osnovi raspoloživ samo na Windows OS-Izvorni kod</p>	<p>117. Celovite arhitekture srednjee sloja, sličnosti i razlike Za izgradnju c/s aplikacija u heterogenim distribuiranim okruženjima:-OSF DCE (Open Software Foundation Distributed Computing Environment)-OMG CORBA (Object Management Group Comon Object Request Broker Architecture).- Osnovna sličnost: MW za C/S-Osnovna razlika: -DCE je bio dizajniran za podršku proceduralnom programiranju, dok je-CORBA bila dizajnirana za podršku objektno-orijentisanom programiranju118. OSF DCE -U osnovi sadrži NOS funkcionalnost, tj. NOS na nivou celog preduzeća.-DCE je razvijen od strane OSF (danas Open Group), kreira otvoreno Enterprise NOS okruženje koje obuhvata višestruke arhitekture, protokole i OS.-Predstavlja korak na putu razvoja od Department NOS (1990), preko Enterprise NOS (1996) do Internet NOS (1998).-Poziva udaljene procedure (RPC)-Distribuirane servise imena (Naming services)-Servise vremenske sinhronizacije-Distribuiranog fajl sistema-Servise sigurnosti mreže (security)-Paket sa nitima (threads)</p> <p>120. Distribuirani Naming servisi-OSF je usvojio ove servise od DEC i Siemens produkata.-DNS omogućavaju resursima da budu identifikovani sa korisnički orijentisanim imenima u namenskoj distribuiranoj BP koja opisuje objekte od interesa.-Imena objekata su nezavisna od njihove lokacije u mreži.-DCE servisi direktorijuma se sastoje od dva elementa-celjskog servisa direktorijuma (CDS)-globalnog servisa direktorijuma (GDS)-Obezbudeju lokalnu naming autonomiju i glob. interoperabilnost.-Globalni pristup korišćenjem X.500 naming sistema ili TCP/IP Internet DNS-a (Domain Name System).</p> <p>121. Tipovi middleware-a-Komunikacioni : omogućavaju aplik. da komuniciraju, kako međusobno, tako i sa serverima.-Informacioni : omogućavaju aplikacijama pristup i zajedničko korišćenje informacija u mreži.-Upravljački : omogućavaju pozivanje udaljenih aplik., koordinaciju izvršavanja između višestrukih apl. i upravljanje višestrukim izvršavanjem u okviru jedne apl.</p>	<p>122. Srednji sloj RPC tipa -Najstariji tip MW-RPC MW proširuje interfejs poziva lok. procedure apstrakcijom da je sposoban da pozove proceduru čiji se glavni deo nalazi negde u mreži.-Inicijalno razvijen od strane firme Sun Microsystems kao deo njihove Open Network Computing (ONC) platforme, kao i Apollo Network Computing System (NCS), koji je postao deo OSF DCE.-Serverske komponente koje izvršavaju RPC-ove se zovu RPC programi-RPC program ima definiciju interfejsa koja specificira procedure koje mogu biti pozvane iz daljine.ID definiše i tipove podataka argumentata koji se prosledjuju udaljenoj proceduri.-Omogućava da se funkcija pozove iz jednog programa, a da se izvršava unutar drugog, na drugom računaru.-Klijent i server rade kao dva odvojena procesa. Ovi procesi komuniciraju putem stub-ova. -Stub je s/w koji obavlja konverziju lokalnih proceduralnih poziva u seriju mrežnih RPC funkcijskih poziva.-Stub određuje adresu serverskog procesa, usmerava i konvertuje pozivne parametre procedure iz interne predstave u standardnu (ASN.1)-Podaci se predaju RPC transportnom protokolu nižeg nivoa, koji šaje poruke (send/receive)-Stub je komunikacioni interfejs koji implementira RPC protokol i specificira kako se poruke kreiraju i razmenjuju.-IDL kompajler je ključni deo koji se koristi za generisanje stub-ova, koji se potom linkuju sa programima klijenta-Postoji veći broj komercijalno raspoloživih protokol kompajlera:-HP: Appolo Network Computing System (NCS)-Sun: Transport Independent RPC-Netwise: RPC Tool-Većina UNIX sistema isporučuje RPC razdvojne biblioteko i alate kao deo baznog OS.-RPC bio osnova OSF DCE, koristi se i od strane MS COM-a, i drugih.-OO MW vodi poreklo od RPC-al-Danas IDL kompajler: u sastavu OS-Zbog ograničenja po performansama ne preporučuje se korišćenje RPC-a po sporijim mrežama kakva je Internet.</p>
<p>123. Srednji sloj MOM tipa-Kreiran sa ciljem prevazilaženja nedostataka RPC-a, posredstvom korišćenja poruka (message).-MOM obezbudeju apstrakciju que-a poruka kome se može pristupiti preko mreže. Predstavlja generalizaciju poznatog OS konstrukta, mailboxa, tj. poštanskog sandučeta.-Veoma je fleksibilan.- Tradicionalni MOM je tipično quing s/w koji koristi poruke kao mehanizam za prebacivanje informacije od tačke do tačke.-Koristi se asinhroni način komunikacije koji ne blokira apl. Arhitektura ↔-MOM produkti imaju que-ove sa persistencijom, replikacijom, real-time performansama125. Transakciono orijentisani MW (aplikacioni servisi)-AS nisu nešto novo.-AS obezbudeju savremeni pristup i deljenje metoda kao i mehanizam za integraciju. -Većina AS se danas koristi kao Web-enabled MW, koji obrađuje transakcije od web-omogućenih aplikacija. -Koriste oo jezike (npr. Java) umesto, proceduralnih (npr. C, COBOL), koji se često koriste kod tradicionalnih TPM-a.-AS su servisi koji omogućuju: ne samo zajedničko korišćenje i obradu apl logike, nego i konekciju sa back-end resursima (npr. BP, ERP apl. čak i nasledene MF apl.)-Npr. sa AS koji zahteva samo oko 50 konekcija, više od 1.000 klijenata može pristupiti DB serveru.-Obezbudeju mehanizme za razvoj UI.-Obezbudeju i mehanizme za puštanje aplikacije na platformu za Web-Primeri (Web) AS:-Sun/AOL/ Netscape/Kiva: Application Server (NAS)-Inprise/Visigenic: Application Server -IBM: WebSphere-BEA/WebLogic: Tenga-Oracle Application Server 4.0-Microsoft: IIS-MSW TPM i AS tipa ima značajan ulogu u EAI126. MW baziran na distribuiranim objektimaPostoje dve osnovne arhitekture, kao i na njima bazirane tehnologije, alati:-OMG CORBA-MS COM/DCOM (Component Object Model)DO-MW obezbudeju apstrakciju udaljenog objekta ali čiji metodi mogu biti pozvani kao da su u adresnom prostoru onoga koji poziva.-Distribuirani objekti su u osnovi mali aplikacioni programi koji koriste standardne interfejs i protokole za međusobnu komunikaciju.-Smatraju se i MW pošto omogućavaju komunikaciju između aplikacija, ali i tehnologiju koja podržava zajedničko korišćenje metoda na nivou preduzeća.</p>	<p>129. Povezivanje/integracija aplikacija u preduzeću (EAI) Motivacija: prelazak sa monolitnih specijalno pravljenih apl. na komercijalno raspoložive aplikacije na heterogenim platformama. Veći broj aplikacija, svaka sa svojom BP kreira tzv. "ostrva podataka" što stvara brojne probleme-svako ostrvo ima svoje značenje ent. objekata-svako ostrvo ima podatke koji se preklapaju -ni jedno ostrvo ne sadrži svu potrebnu info o enterprise objektu.Dopunski zahtevi-web-izacija preduzeća / aplikacija-B2B integracijasto zahteva prethodnu internu integraciju (EAI), za koju je potrebna odgovarajuća arhitektura koja definiše strategiju integracije.</p> <p>130. EAI na nivou procesa -Bavi se izgradnjom opšte kompanjskih poslovnih procesa i uključivanjem postojećih aplikacija u te procese. EAI na ovom nivou predstavlja proširenje EAI na nivou poruka.-Stvarna razmena podataka se i dalje vrši putem messaging-a, dok se EAI MW ponaša kao workflow mašina.</p> <p>131. Brokerski poruka -Brokerski poruka (Message Brokers) su centralni deo integracione arhitekture i implementacije na nivou procesa. Osnovne funkcije-Rutiranje poruka-Transformacija podataka-Implementacija poslovnih pravila-Podrška poslovnim procesima-za veći deo navedenih Fja MB imaju odgovarajuća GUI-bazirana sredstva.-MB se grade "iznad" postojeće MW tehnologije, žargon: midlver midlvera.-Omogućavaju prebacivanje informacija između dva ili više resursa (izvorne ili ciljne apl.) i mogu uvažiti razlike u aplikacionoj semantici, kao i u platformama.-MB mogu transformisati šemu (promeniti strukturu poruke) i sadržaj informacije.</p>	 <p>SQL FAP</p> <p>138. Povezivanje sa ODBC, JDBC-MS Open Data Base Connectivity (ODBC) Windows stnd za SQL, proširena verzija SAG CLI-a.-MS ODBC 1.0 SDK: 1992. (DBA za Win), spor, bagovit.-MS ODBC 2.0 SDK: 1994. (za 32 bitne)-Većina DB server vendora danas podržava ODBC API, osim nativnih139. Povezivanje sa OLE-DB i ADO-Konkretna Microsoft arhitektura-Interfejs podataka nižeg nivoa OLE DB obezbudeju mehanizam za pristup proizvoljnom broju data resursa, uključujući BP, kao standardnim COM objektima.-Interfejs podataka višeg nivoa ADO, ActivX Data Objects (DAO ranije) obezbudeju pristup podacima za jezike i alate bez pointera (VB, VBscript, Java script)</p>
<p>142. Povezivanje putem zajedničkog gateway-a</p>  <p>Gateway FAP</p> <p>144. Otvoreni sistemi i IT standardi, standardizacione organizacije -U realizaciji savremenih IS treba se opredeliti za koncepciju otvorenih sistema, koja omogućava korisniku da elemente bira na bazi standarda.-Sistem koji implementira dovoljno otvorenih (standardizovanih) specifikacija za interfejs, servise i formate podrške (IEEE).Osobine otvorenih sistema.-Portabilnost: - Skalabilnost-Interoperabilnost-Interkonektivnost, Stabilnost</p>	<ol style="list-style-type: none"> Osnovna terminologija (IT, IS, DBMS, BP), Oblasti primene IT, Kategorije problema podesne za rešavanje uz pomoć IT, Oblici primene IT Evolucija odnosa aplikacije, OS i H/W Generacije programskih jezika Generacije H/W Etape u razvoju sistema za upravljanje podacima Obrada zasnovana na fajlovima Obrada zasnovana na korišćenju BP ANSI-SPARC arhitektura Kategorije savremenih DBMS Relacione BP Multimedijalne BP Sistemi za upravljanje bazama podataka (DBMS) DBMS-osnovne funkcije DBMS-osnovne komponente DDL funkcije DDL kompajler Funkcije SQL DB servera Arhitektura SQL DB servera Izbor arhitekture SQL DB servera Arhitektura RDBMS Oracle10g Detaljna arhitektura Oracle10g servera Oracle10g procesi Oracle instance Logička struktura Oracle BP Fizička struktura Oracle BP Objektno-relacioni DBMS Objektno-orijentisani DBMS 	<ol style="list-style-type: none"> Kanalisanje i TPM Standardi za TP monitore Komparacija TPM i memorijskih procedura Načini distribucije procesa Udaljeni zahtev Udaljena transakcija Distribuirani zahtev Distribuirana jedinica posla Protokol dvofaznog izvršavanja Osnovna svojstva distribuiranog DBMS Multidatabase sistemi Federalizovani MDBMS sistemi Arhitektura federalizovanih MDBMS sistema Vlasništvo nad podacima Modeli obrade podataka Arhitekture informacionih sistema Centralizovana višekorisnička arhitektura Distribuirana jedнокorisnička arhitektura Distribuirana višekorisnička arhitektura Prednosti i nedostaci distribuiranih IS Arhitekture distribuiranih IS (DIS) Master-slave model Klijent-server model Model od-sloja-do-sloja (P-to-P) Grupni model Model distribuiranih objekata Model multimedijalnog toka Klasifikacija klijent-server sistema prema hijerarhiji, lokaciji Klasifikacija klijent-server sistema prema nameni servera
<ol style="list-style-type: none"> Srednji sloj (middleware) C/S sistema, namena, položaj, servisi Celovite arhitekture srednjeg sloja, sličnosti i razlike OSF DCE OSF DCE arhitektura Distribuirani Naming servisi Tipovi middleware-a Srednji sloj RPC tipa Srednji sloj MOM tipa Transakciono orijentisan MW (transakcioni monitori) Transakciono orijentisan MW (aplikacioni servisi) MW baziran na distribuiranim objektima Middleware tipa OMG CORBA Centralni deo CORBA arhitekture Povezivanje/integracija aplikacija u preduzeću (EAI) EAI na nivou procesa Brokerski poruka Srednji sloj za povezivanje sa BP (namena, položaj i funkcije) Načini povezivanja klijenta i servera Native/proizvodjački DB MW Arhitektura c/s sistema sa DB MW u homogenom okruženju Arhitektura c/s sistema sa DB MW u heterogenom okruženju Implementacije tipa zajedničkog interfejsa Povezivanje sa ODBC, JDBC Povezivanje sa OLE-DB i ADO MS ADO NET Arhitektura ADO NET Povezivanje putem zajedničkog gateway-a Povezivanje putem zajedničkog protokola Otvoreni sistemi i IT standardi, standardizacione organizacije 		