

Prof. dr Ljiljana Kaščelan

INFORMACIONE TEHNOLOGIJE
ZA PODRŠKU POSLOVNOM
ODLUČIVANJU

Podgorica, 2012.

PREDGOVOR

U nekoliko proteklih decenija, otkako su se pojavili sistemi za podršku odlučivanju, izgrađivan je pomalo kontroverzan stav o njima, počev od nerealno visokih očekivanja, mnogo spekulacija i u velikoj mjeri frustracija za menadžere. I pored toga što se kompjuterska tehnologija transformisala gotovo uporedo sa poslovnim transakcijama i podacima koje one obrađuju, menadžeri su često bili obeshrabreni u pokušaju primjene informacionih tehnologija u procesu poslovnog odlučivanja. Međutim u posljednje vrijeme, zahvaljujući visokim tehnološkim dostignućima, kao što su data warehouse, OLAP (*Online Analytical Processing*), data mining i vizuelno modeliranje, stav menadžmenta se u značajnoj mjeri mijenja.

Cilj ove knjige jeste da pomogne menadžerima i studentima Ekonomije i Menadžmenta da nauče da primjenjuju ova tehnološka dostignuća. Knjiga je nastala iz dugogodišnjeg predavačkog iskustva autora na predmetima Informatika, Baze podataka i Sistemi poslovne inteligencije na redovnim i postdiplomskim studijama Ekonomskog fakulteta, kao i na osnovu učešća u projektima razvoja i realizacije ovih sistema.

U **prvom poglavlju** definisane su osnovne informacione tehnologije, kao što su hardver, softver, mreže i baze podataka, kao i njihova primjena u poslovanju. Ovo poglavlje namijenjeno je studentima Ekonomskog fakulteta za predmet Informatika, kao i studentima Studija menadžmenta za predmet Poslovna informatika.

Podaci predstavljaju virtuelni ali vitalni resurs u poslovanju. Upravljanje podacima podrazumijeva poznavanje nekih logičkih koncepata koji omogućavaju logičko organizovanje i strukturiranje podataka u bazi. **Drugo poglavlje** knjige posvećeno je relacionom modelu podataka koji predstavlja strukturu najčešće korišćenu u poslovnim organizacijama, kao i upitnom jeziku SQL, koji predstavlja bazičnu informacionu tehnologiju za podršku odlučivanju u poslovanju. Ovo poglavlje namijenjeno je studentima Ekonomskog fakulteta za predmete Baze podataka i Poslovne baze podataka, kao i svim sofisticiranim menadžerima koji žele da svoje podatke ad hoc pretražuju i koriste.

Treće poglavlje definiše sisteme za podršku odlučivanju (Decision Support System-DSS) i njihovu ulogu u procesu poslovnog odlučivanja. U okviru njega definisane su i vrste DSS i njihove specifičnosti.

Posebnu kategoriju sistema za podršku odlučivanju čine sistemi poslovne inteligencije (Business Intelligence – BI), koji spadaju u DSS zasnovane na podacima, ali i u sugestivne DSS. Osnovne komponente BI čine data warehouse, OLAP (On Line Analytical Processing) i data mining. Modeliranje data warehouse podataka, OLAP i data mining tehnologije, kao i njihove primjene u poslovanju biće predmet razmatranja **četvrtog poglavlja**.

Treće i četvrto poglavlje namijenjeno je studentima osnovnih i magistarskih studija Ekonomskog fakulteta za predmet Sistemi poslovne inteligencije.

U **prilogu** je dat primjer realizacije jednog OLAP sistema pomoću *oracle* BI alata. Ovaj primjer može da posluži ne samo kao model za implementaciju ovih sistema nego i kao model za upotrebu i mogućnosti ovakvih sistema. Predmet ovog poglavlja jeste postupak razvoja BI sistema ali i efikasnost i prednosti velikog broja analiza koje nam jedan ovakav sistem omogućava. Prilog može poslužiti studentima osnovnih i magistarskih studija prilikom realizacije praktičnog projekta na Ekonomskom fakultetu za predmet Sistemi poslovne inteligencije.

Na kraju svakog poglavlja data su pitanja i zadaci za vježbu, kroz koje studenti mogu obnoviti i utvrditi znanja.

Recenzentima, prof. dr Vujici Lazoviću i prof. dr Dragani Bečejski-Vujaklija, dugujem zahvalnost na korisnim primjedbama i sugestijama.

Autor

Podgorica, april 2012.

SADRŽAJ

1. OSNOVNE INFORMACIONE TEHNOLOGIJE.....	12
1.1. UVOD.....	12
1.2. INFORMACIONI SISTEMI – OSNOVNI KONCEPTI.....	12
1.2.1. KOMPONENTE I AKTIVNOSTI INFORMACIONIH SISTEMA.....	14
1.2.2. VRSTE IS-.....	22
1.3. RAČUNARSKI HARDVER.....	24
1.3.1. MIKROKOMPJUTERSKI SISTEMI.....	30
1.3.2. MIDRANGE KOMPJUTERSKI SISTEMI.....	33
1.3.3. MAINFRAME KOMPJUTERSKI SISTEMI.....	33
1.3.4. KONCEPT RAČUNARSKOG SISTEMA.....	34
1.3.5. RAČUNARSKE PERIFERIJE: ULAZNE, IZLAZNE I MEMORIJSKE TEHNOLOGIJE.....	37
1.4. RAČUNARSKE MREŽE.....	54
1.4.1. TRENDOWI U TELEKOMUNIKACIJAMA I RAČUNARSKIM MREŽAMA.....	54
1.4.2. PREDNOSTI PRIMJENE RAČUNARSKIH MREŽA U POSLOVANJU.....	55
1.4.3. INTERNET.....	56
1.4.4. POJAM I ULOGA INTRANETA U POSLOVANJU.....	58
1.4.5. POJAM I ULOGA EKSTRANETA U POSLOVANJU.....	60
1.4.6. TIPOVI TELEKOMUNIKACIONIH MREŽA.....	60
1.4.7. MREŽNE TEHNOLOGIJE.....	65
1.4.8. MREŽNE TOPOLOGIJE.....	76
1.4.9. MREŽNI SERVISI.....	78
1.4.10. MREŽNE ALTERNATIVE.....	81
1.5 RAČUNARSKI SOFTVER.....	82
1.5.1. SOFTVERSKI TRENDOWI.....	83
1.5.2. APLIKATIVNI SOFTVER – SOFTVER ZA KRAJNJE KORISNIKE.....	84
1.5.3. SISTEMSKI SOFTVER.....	101
1.6. ORGANIZACIJA PODATAKA.....	115
1.6.1. OSNOVNI KONCEPTI PODATKA.....	116

1.6.2. SISTEMI ZA UPRAVLJANJE BAZAMA PODATAKA.....	117
1.6.3. TIPOVI BAZA PODATAKA.....	120
1.6.4. PROBLEMI I IZAZOVI UPRAVLJANJA PODACIMA.....	123
1.6.5. TEHNIČKE OSNOVE UPRAVLJANJA PODACIMA.....	124
1.6.5.1. STRUKTURE PODATAKA.....	125
1.6.5.2. METODE PRISTUPA PODACIMA.....	131
1.6.6. RAZVOJ BAZA PODATAKA.....	133
1.7. PITANJA I ZADACI ZA VJEŽBU.....	138
 <u>2. RELACIONE BAZE PODATAKA.....</u>	 147
2.1. POJAM BAZE PODATAKA.....	147
2.2. FIZIČKI, KONCEPTUALNI I KORISNIČKI NIVO APSTRAKCIJE PODATAKA.....	147
2.3. OSNOVNI POJMOVI O MODELU PODATAKA.....	148
2.4. MODEL ENTITETI – ODNOSI (E-R MODEL).....	149
2.4.1. ENTITETI I SKUPOVI ENTITETA.....	150
2.4.2. ATRIBUTI.....	150
2.4.3. ODNOSI I SKUPOVI ODNOSA.....	151
2.4.4. EGZISTENCIJALNA ZAVISNOST.....	152
2.4.5. PRIMARNI KLJUČEVI.....	152
2.4.6. INTEGRITETSKA OGRANIČENJA E-R MODELA.....	155
2.4.7. DIJAGRAMI E-R MODELA.....	155
2.4.8. INFORMACIONA STRUKTURA E-R MODELA.....	158
2.4.8.1. PREDSTAVLJANJE JAKIH SKUPOVA ENTITETA.....	158
2.4.8.2. PREDSTAVLJANJE SLABIH SKUPOVA ENTITETA.....	159
2.4.8.3. PREDSTAVLJANJE SKUPOVA ODNOSA.....	159
2.4.9. PREDNOSTI I NEDOSTACI E-R MODELA.....	160
2.4.10. PRIMJER: DIO BAZE PODATAKA BIBLIOTEKE.....	161
2.5. RELACIONI MODEL.....	163
2.5.1. RELACIJA – OSNOVNA INFORMACIONA STRUKTURA RELACIONOG MODELA.....	163
2.5.2. AŽURIRANJE RELACIJA.....	165
2.5.3. RELACIONA ALGEBRA.....	167
2.5.3.1. SKUPOVNE OPERACIJE.....	168

2.5.3.2. OPERATOR SELEKCIJE.....	168
2.5.3.3. OPERATOR PROJEKCIJE.....	170
2.5.3.4. OPERATOR SPAJANJA.....	170
2.5.3.5. PRIMJERI ZA RELACIONU ALGEBRU.....	171
2.5.4. UPITNI JEZIK SQL.....	173
2.5.4.1. PRETRAŽIVANJE PODATAKA.....	174
2.5.4.2. AŽURIRANJE PODATAKA I POGLEDI.....	178
2.5.4.3. PRIMJER: DIO BAZE PODATAKA STUDENTSKE SLUŽBE – SQL UPITI.....	181
2.5.4.4. PRIMJER: RELACIONA ALGEBRA – SQL UPITI.....	183
2.5.5. LOGIČKO PROJEKTOVANJE RELACIONIH BAZA.....	184
2.5.5.1. FUNKCIJSKE ZAVISNOSTI.....	185
2.5.5.1.1. OSOBINE FUNKCIJSKIH ZAVISNOSTI.....	187
2.5.5.1.2. LOGIČKE IMPLIKACIJE FUNKCIJSKIH ZAVISNOSTI.....	188
2.5.5.1.3. PRIMJERI.....	191
2.5.5.2. DEKOMPOZICIJA RELACIONIH ŠEMA.....	192
2.5.5.2.1. NORMALNE FORME.....	193
2.5.6. NEDOSTACI RELACIONOG MODELA.....	200
2.5.7. FUNKCIJE RELACIONOG DBMS-A.....	201
2.5.7.1. POJAM KONZISTENTNOSTI I TRANSAKCIJE.....	202
2.5.7.2. RESTAURACIJE KONZISTENTNOSTI BAZE PODATAKA.....	203
2.5.7.3. KONKURENTNI DOSTUP.....	204
2.6. PITANJA I ZADACI ZA VJEŽBU.....	206
<u>3. SISTEMI ZA PODRŠKU ODLUČIVANJU</u>	<u>211</u>
3.1. UVOD.....	211
3.2. UPRAVLJAČKI INFORMACIONI SISTEMI I SISTEMI ZA PODRŠKU ODLUČIVANJU.....	211
3.3. DSS – POJAM I VRSTE.....	213
3.4. EVOLUCIJA KONCEPATA DSS.....	219
3.5. DSS I TRANSAKCIONI SISTEMI – PARALELA.....	221
3.6. DSS U PROCESU POSLOVNOG ODLUČIVANJA – PREDNOST ILI NEDOSTATAK?.....	223

3.7. DSS ZASNOVANI NA PODACIMA.....	225
3.7.1. PRIRODA I IZVORI DSS PODATAKA.....	226
3.7.2. METODE ZA PRIKUPLJANJE DSS PODATAKA.....	227
3.7.3. KOMPARACIJA DSS I OPERATIVNIH PODTAKA.....	228
3.7.4. SPECIFIČNOSTI RAZVOJA DSS ZASNOVANIH NA PODACIMA.....	231
3.8. DSS ZASNOVANI NA MODELIMA.....	232
3.8.1. STANDARDNI KVANTITATIVNI MODELI.....	236
3.8.1.1. RAČUNOVODSTVENI I FINANSIJSKI MODELI.....	237
3.8.1.2. MODELI ZA ANALIZU ODLUČIVANJA.....	238
3.8.1.3. PREDIKTIVNI MODELI.....	242
3.8.1.4. OPTIMIZACIONI MODELI.....	244
3.8.1.5. SIMULACIONI MODELI.....	244
3.8.1.6. HEURISTIČKI I KVALITATIVNI MODELI.....	246
3.9. SUGESTIVNI DSS – EKSPERTNI SISTEMI.....	249
3.9.1. DATA MINING.....	253
3.9.2. SPECIFIČNOSTI RAZVOJA SUGESTIVNIH DSS.....	257
3.10. GRUPNI DSS (GDSS).....	258
3.10.1. KOMUNIKACIONE I KOLABORATIVNE TEHNIKE.....	258
3.10.2. KARAKTERISTIKE I METODI GRUPNOG ODLUČIVANJA.....	267
3.10.3. DEFINICIJA GDSS.....	269
3.10.3.1. CILJ I TEHNOLOGIJE GDSS.....	270
3.11. IZVRŠNI DSS (EIS, ESS).....	271
3.11.1. DEFINICIJA EIS I ESS I PROCES IZVRŠNOG ODLUČIVANJA.....	271
3.11.2. KARAKTERISTIKE EIS.....	274
3.11.3. KOMPARACIJA I INTEGRACIJA EIS, MIS I DSS.....	275
3.12. PITANJA I ZADACI ZA VJEŽBU.....	279
<u>4. POSLOVNA INTELIGENCIJA.....</u>	280
4.1. UVOD.....	280
4.2. DATA WAREHOUSE	280
4.2.1. MODELIRANJE PODATAKA U RELACIONOM DATA WAREHOUSE-U.....	283

4.3. OLAP – POJAM, VRSTE I ALATI.....	290
4.4. DATA MINING.....	295
4.4.1. DATA MINING ZASNOVAN NA INDUKTIVNOM MAŠINSKOM UČENJU.....	298
4.4.1.1. KLASE, OPISI KLASA I KLASIFIKACIONA PRAVILA.....	299
4.4.1.2. PROSTOR PRETRAŽIVANJA.....	301
4.4.1.3. STRATEGIJE PRETRAŽIVANJA.....	303
4.4.1.4. PROBLEMI KOJI SE JAVLJAJU ZBOG BAZE PODATAKA KAO <i>TRAINING SETA</i>	311
4.4.2. PRIMJERI PRIMJENE DATA MINING METODA....	313
4.4.3. POSLOVNE PRIMJENE DATA MINING-A.....	319
4.5. PITANJA I ZADACI ZA VJEŽBU.....	323
<u>PRILOG</u>.....	325
1. ORACLE WAREHOUSE BUILDER I FAZE RAZVOJA DATA WAREHOUSE-A.....	325
1.1. FAZA DEFINISANJA DW KONCEPTA.....	327
1.2. FAZE GENERISANJA, INICIJALNOG PUNJENJA I PERIODIČNOG OSVJEŽAVANJA DW-A.....	345
2. ORACLE DISCOVERER – REALIZACIJA I ANALIZA OLAP MODELA.....	351
2.1. REALIZACIJA MODELA.....	354
2.2. ANALIZA MODELA.....	366
2.3. REZIME.....	375
<u>LITERATURA</u>.....	377

1. OSNOVNE INFORMACIONE TEHNOLOGIJE

1.1. UVOD

Različiti profili menadžmenta zahtijevaju različite vrste informacija. Menadžment informacije mogu biti detaljne informacije o poslovnim transakcijama ili samo njihovi sumarijumi, grafikoni, tabele ili prosto brojevi. Poslovne informacije ili analize moraju posjedovati određene osobine. Najprije, one moraju biti vremenski zavisne i aktuelne. Ove osobine zapravo znače to da su tačne i kompletne informacije uvijek dostupne u onom trenutku kada ih menadžer zahtijeva. Informacije moraju biti prezentovane u odgovarajućem formatu koji će biti lako razumljiv za korisnika.

Uopšteno govoreći, informacioni sistem obezbjeđuje takve poslovne informacije i pomaže menadžmentu u razumijevanju poslovnih rezultata. Informacioni sistem se pri tome oslanja na osnovne informacione tehnologije hardver, softver, mreže i baze podataka.

1.2. INFORMACIONI SISTEMI – OSNOVNI KONCEPTI

Informacioni sistem (IS) jeste organizovana kombinacija ljudi, hardvera, softvera, komunikacionih mreža i resursa podataka, koji sakupljaju, transformišu i prenose informacije u okviru neke organizacije.

Ljudi se pomažu informacionim sistemom da bi komunicirali jedni sa drugima, pri čemu koriste mnoštvo tehničkih uređaja (hardver), uputstva i procedure za obradu informacija (softver), komunikacione kanale (mreže) i uskladištene podatke (resursi podataka). Neki od tih informacionih sistema koriste samo jednostavne ručne uređaje (papir i olovka), kao i govorne komunikacione kanale. Međutim, za nas će biti interesantni kompjuterski bazirani informacioni sistemi, koji koriste kompjuterski hardver i softver, internet i druge računarske mreže, kompjuterski bazirane tehnike upravljanja podacima i mnoge druge informacione tehnologije za

transformisanje resursa podataka u skup informacionih proizvoda za svoje korisnike.

Da bismo razumjeli informacioni sistem kao dio poslovnog sistema, moramo razumjeti koncept opšteg sistema.

Sistem se uopšteno može definisati kao skupina povezanih i interaktivnih elemenata objedinjenih u jednu cjelinu. Sisteme srećemo u različitim oblastima, ali sljedeća definicija sistema najviše odgovara oblasti informacionih sistema:

Sistem je grupa međusobno povezanih komponenti koje zajedno rade na zadatku prihvatanja ulaza (input) i generisanja odgovarajućih izlaza (output) organizovanim procesom transformisanja. Dakle, sistem ima tri osnovne komponente ili funkcije:

- **Input** omogućava prihvatanje i prilagođavanje ulaza u sistem za njihovu obradu.
- **Procesiranje** jeste proces transformisanja ulaza u odgovarajuće izlaze.
- **Output** podrazumijeva transfer izlaza, dobijenih kao rezultat procesiranja, na odgovarajućim destinacijama.

Na primjer, proizvodni sistem na ulazu prihvata materijale i proizvodi gotove proizvode kao izlaz. Informacioni sistem za ulaz ima podatke koje procesira u proizvode-informacije kao izlaz. Poslovna organizacija predstavlja sistem koji ekonomske resurse različitim poslovnim procesima transformiše u dobra ili usluge.

Obično sistemi uključuju dvije dodatne komponente feedback (povratne informacije) i kontrolu. Ovakvi sistemi nazivaju se *cybernetic* sistemi ili samoregulacioni sistemi.

- **Feedback** predstavlja povratne informacije o performansama sistema. Na primjer, podaci o performansama prodaje predstavljaju feedback za menadžera prodaje.
- **Kontrola** uključuje posmatranje i evaluaciju feedback-a kako bi se utvrdilo napredovanje sistema prema cilju. Kontrolna funkcija garantuje da će se od određenih ulaza dobiti odgovarajući, tj. zahtijevani izlazi. Na

primjer, menadžer prodaje kontroliše efekte pojedinih prodajnih mjesta ocjenjujući feedback o performansama prodaje.

Jednostavan primjer samoregulacionog sistema jeste termostatom kontrolisani sistem zagrijavanja prostorija, koji nadzire i automatski reguliše željenu temperaturu. Drugi primjer jeste ljudsko tijelo, koje se može posmatrati kao cybernetički sistem koji sam prati i reguliše mnoge funkcije kao što su temperatura, disanje i drugo.

Sistem ne egzistira u vakuumu, već funkcioniše u okruženju sa drugim sistemima. Ukoliko je sistem dio nekog većeg sistema, onda je on *podsystem*, a veći sistem predstavlja njegovo okruženje. Sistem koji komunicira sa drugim sistemima iz okruženja (putem input i output interfejsa) naziva se *otvoreni sistem*. Na kraju, sistem koji ima sposobnost sopstvenih izmjena ili promjene okruženja jeste *adaptivni sistem*.

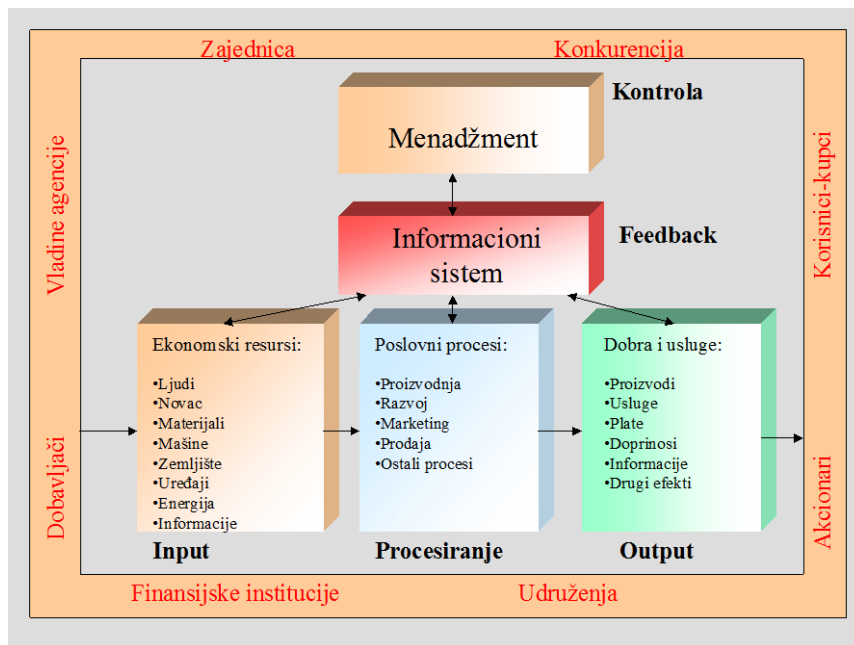
Poslovni sistem predstavlja primjer jednog otvorenog sistema koji funkcioniše u poslovnom okruženju (slika 1.1.). Informacioni sistem jeste podsystem poslovnog sistema, koji obezbjeđuje njegov feedback, tj. generiše performanse poslovanja, čiju kontrolu dalje preuzima menadžment. Informacioni sistem povezan je sa poslovnim sistemom svojim ulazima i izlazima, a poslovni – sa okruženjem, tako da oni predstavljaju otvorene sisteme. Informacioni sistem mora biti adaptivan, tj. ne smije u velikoj mjeri zavisiti od poslovnog sistema i okruženja.

Sada koncepte sistema možemo iskoristiti da bismo bolje shvatili informacioni sistem i njegovo funkcionisanje. Dakle, možemo reći da je IS sistem koji prihvata resurse podataka kao ulaze, a zatim ih procesira u informacije kao izlaz. Kako to IS radi, tj. Koje su komponente i aktivnosti uključene u taj proces?

1.2.1. KOMPONENTE I AKTIVNOSTI INFORMACIONIH SISTEMA

Informacioni sistem se, prilikom obavljanja input i output aktivnosti, procesiranja, skladištenja podataka i informacija i kontrole u cilju konvertovanja resursa podataka u informacione proizvode, oslanja na: ljudske resurse (krajnji korisnici i IS specijalisti), hardver (mašine i mediji), softver (programi i procedure), podatke (baze podataka i baze znanja) i mrežu (komunikacioni mediji i mrežni uređaji). Dakle, IS se sastoji od pet

glavnih komponenti: ljudi, hardver, softver, podaci i mreže. U daljem tekstu reći ćemo nešto više o svakoj od nabrojanih komponenti i njihovoj ulozi u funkcionisanju IS-a kao cjeline.



Slika 1.1. Poslovni sistem kao otvoreni sistem i IS kao njegov podsistem

Ljudski resursi. – Ljudi su neophodni za funkcionisanje kompletnog IS. Uspjeh IS zavisi najviše od ovog resursa. Ma koliko dobar hardver i softver imali, ukoliko ljudi nisu dobro obučeni da rade sa njima i ukoliko ne prihvataju IS, on ne može uspješno funkcionisati. Pod ljudskim resursima se podrazumijevaju krajnji korisnici i IS specijalisti.

- **Krajnji korisnici** (često se nazivaju samo korisnici ili klijenti) jesu ljudi koji koriste IS, tj. informacije koje on produkuje. Oni mogu biti računovođe, prodavci, inženjeri ili menadžeri. Tipični korisnici jesu oni koji većinu svog radnog vremena provode tako što sarađuju i komuniciraju u okviru radnih timova, tj. tako što kreiraju, koriste i distribuiraju informacije.
- **IS specijalisti** jesu ljudi koji razvijaju i održavaju IS. To su sistem analitičari, programeri, sistem operateri i drugi. Ukratko, sistem analitičari dizajniraju IS na osnovu korisničkih informacionih zahtjeva,

programeri kreiraju kompjuterske programe oslanjajući se na specifikacije sistem analitičara, a sistem operateri nadziru i održavaju kompjuterski sistem i mrežu.

Hardverski resursi. – Pod hardverom se podrazumijevaju svi fizički uređaji koji se koriste prilikom procesiranja informacija. Ovo ne podrazumijeva samo mašine, kao što su kompjuteri i priključci, već i medijume na kojima su podaci zapisani, počev od papirnih, pa sve do magnetnih i optičkih diskova. Primjeri hardvera u kompjuterski baziranim IS-u su:

- **Kompjuterski sistem** se sastoji od centralne procesorske jedinice (CPU), koja sadrži mikroprocesor, i skupa međusobno povezanih perifernih uređaja. Kompjuterski sistemi mogu biti laptop, desktop, midrang i large mainframe sistemi.
- **Kompjuterske periferije** jesu uređaji kao što su: tastatura ili elektronski miš za unos podataka i komandi, monitor i štampač za izdavanje izlaza, ili magnetni i optički diskovi za skladištenje podataka.

Softverski resursi. – Koncept softverskih resursa podrazumijeva skup instrukcija za procesiranje informacija. Generalno, koncept softvera ne uključuje samo skup operativnih instrukcija računaru, koje nazivamo **programom**, već i instrukcije korisnicima za procesiranje informacija, koje nazivamo **procedurama**. Napomenimo da i IS koji se ne oslanjaju na kompjutere sadrže softversku komponentu. To su obično instrukcije ili uputstva za odgovarajuće čuvanje, obradu i dostavljanje informacija njihovim korisnicima. Primjeri softverskih resursa su:

- **sistemske softver**, kao što je operativni sistem koji kontroliše i podržava operacije u okviru kompjuterskog sistema;
- **aplikativni softver**, koga čine programi namijenjeni procesiranju konkretnih korisničkih zadataka, kao što su program za analizu prodaje, program za obradu teksta i drugi;
- **procedure**, koje su zapravo operativna uputstva za korisnike IS-a, kao na primjer korisničko uputstvo za neki softverski paket.

Resursi podataka. – Podaci su mnogo više od običnog sirovinskog materijala za IS. Podaci predstavljaju jedan od osnovnih organizacionih resursa, koji je bitan menadžerima isto koliko i IS profesionalcima. Dakle,

efikasno upravljanje podacima predstavlja benefit za korisnike informacionog sistema, ali i za kompletnu organizaciju.

Podaci mogu imati različite formate, uključujući tradicionalne *alfanumeričke* podatke, koji se sastoje od brojeva, slova i drugih specijalnih znakova, *tekstualne* podatke u vidu rečenica i pasusa, koji se koriste u pisanim komunikacijama, *image* podatke, kao što su grafički oblici i figure, *audio* podatke, kao što su ljudski glas i ostali zvuci i druge.

Podaci su, u okviru IS, tipično organizovani i uskladišteni nekim od mnogobrojnih sistema za upravljanje podacima u:

- bazama podataka, koje čuvaju procesirane i organizovane podatke i
- bazama znanja, u kojima su smeštena znanja u vidu činjenica, pravila i primjera iz uspešnih poslovnih iskustava.

Na primjer, podaci o prodajnim transakcijama mogu biti akumulirani, procesirani i smešteni u *web* dostupnoj bazi podataka prodaje, koja omogućava kreiranje izvještaja za analizu prodaje od strane menadžera ili marketinga. Baze znanja koriste ekspertni sistemi u automatskom pronalaženju odgovora kao pomoć ekspertu pri rješavanju određenog problema.

Razlika između podatka i informacije. – Sa aspekta IS, podatak i informacija jesu različiti koncepti, iako se ti termini često koriste ravnopravno za jedan ili drugi koncept. **Podaci** jesu sirove činjenice ili posmatranja o nekom fizičkom fenomenu ili poslovnim transakcijama. Preciznije, podaci predstavljaju vrijednosti atributa pojedinih entiteta (kao što su ljudi, mjesta, stvari i događaji). **Informacije** jesu podaci konvertovani u oblik koji ima značenje za specifičnog korisnika. Ovo konvertovanje podataka jeste zapravo procesiranje podataka, koje obično podrazumijeva agregiranje, organizovanje, analiziranje ili evaluaciju podataka u cilju njihovog postavljanja u odgovarajućem kontekstu za korisnika.

Na primjer, nazivi robe, količine i novčani iznosi, sačuvani u okviru prodajne fakture, predstavljaju podatke o prodajnim transakcijama. Međutim, menadžer prodaje ne želi takve pojedinačne podatke o prodajnim transakcijama. Za njega informaciju predstavljaju organizovani i procesirani

podaci u vidu pregleda ukupnog iznosa prodaje prema tipovima proizvoda ili prema prodajnim mjestima.

Mrežni resursi. – Telekomunikacione tehnologije i mreže, kao što su internet, intranet i ekstranet, esencijalne su za uspjeh e-businessa i ostalih poslovnih transakcija neke organizacije, tj. njenog kompjuterski baziranog IS. Računarske mreže sastoje se od kompjutera, komunikacionih procesora i ostalih uređaja povezanih sa komunikacionim medijima i kontrolisanih od strane komunikacionog softvera. Može se reći da komunikacione tehnologije i mreže predstavljaju fundamentalne komponente cijelog IS. Mrežni resursi podrazumijevaju:

- **Komunikacione medije** čine upredena parica, koaksijalni i fiber-optički kablovi, kao i mikrotalasne, celularne i satelitske bežične tehnologije.
- **Mrežna podrška** podrazumijeva razne vrste hardvera, softvera i tehnolgija koje omogućavaju korišćenje komunikacione mreže. Primjeri su komunikacioni procesori kao što je modem, ili komunikacioni kontrolni softver kao što je mrežni operativni sistem ili internet browser.

Već smo rekli da su osnovne aktivnosti IS input, procesiranje, output, skladištenje (memorisanje) i kontrola. Razmotrimo sada detaljnije svaku od ovih aktivnosti IS.

Input. – Podaci o poslovnim transakcijama i ostalim događajima moraju biti uneseni u kompjuterski sistem i pripremljeni za procesiranje, što čini suštinu input aktivnosti. Input tipično podrazumijeva unos podataka (*data entry*) i editovanje. Korisnik obično unosi podatke o transakcijama direktno u kompjuterski sistem ili zapisuje podatke na nekom od fizičkih medija kao što je na primjer papir. Ovo podrazumijeva i niz korektivnih aktivnosti, tj. editovanje podataka kako bi oni bili korektno uneseni. Uneseni podaci se do njihovog procesiranja memorišu na magnetnom disku u mašinski čitljivom formatu.

Tako se podaci o poslovnim transakcijama obično nalaze u okviru izvornog papirnog dokumenta (kao što je na primjer narudžbenica). Ove podatke korisnik direktno unosi u kompjuterski sistem preko tastature ili pomoću uređaja za optičko skeniranje. Korektnost unosa obezbjeđuje se pomoću adekvatnog video displeja, tj. korisničkog interfejsa. Korisnički interfejs zapravo predstavlja metod unosa podataka. Najpoznatiji korisnički interfejsi

jesu sistemi menija, promptova i formi za unos podataka. Sistem menija se koristi za podatke koji imaju mali i ograničeni broj vrijednosti, kao na primjer pol (muški, ženski) i slično. Sistem promptova podrazumijeva unos podataka pomoću pitanja na ekranu. Međutim, ovakav unos ne omogućava efikasno editovanje podataka jer poslije pritiska na taster enter podatak više nije moguće ispravljati. Najbolji korisnički interfejs za input jeste ekranska forma. Forma je ekranski obrazac sa praznim poljima u koja se unose podaci. Polja su obično zaštićena od nekorektnog unosa odgovarajućim pravilima i upozorenjima. Na slici 1.2. dat je primjer forme za unos podataka.

Izvor: Korisničko uputstvo programskog paketa e-računi

Slika 1.2. Ekranska forma za unos podataka

Procesiranje. – Pod procesiranjem podataka podrazumijevaju se operacije kao što su izračunavanje, upoređivanje, sortiranje, klasifikovanje i agregiranje. Pomoću ovih operacija podaci se konvertuju u informacije za korisnike. Upravljanje kvalitetom informacija memorisanih u okviru IS obavlja se pomoću operacija ažuriranja podataka, koje se takođe mogu smatrati procesiranjem podataka.

Na primjer, podaci o nekoj transakciji mogu biti dodati ukupnoj sumi prodaje, upoređeni sa standardima da bi se ispitala mogućnost prodajnog popusta, sortirani po identifikacionom broju proizvoda, klasifikovani po

kategorijama proizvoda, agregirani po kategorijama proizvoda (sumarni iznosi prodaje po kategorijama) ili ažurirani na nivou zapisa o pojedinačnim transakcijama u slučaju nekorektnog unosa.

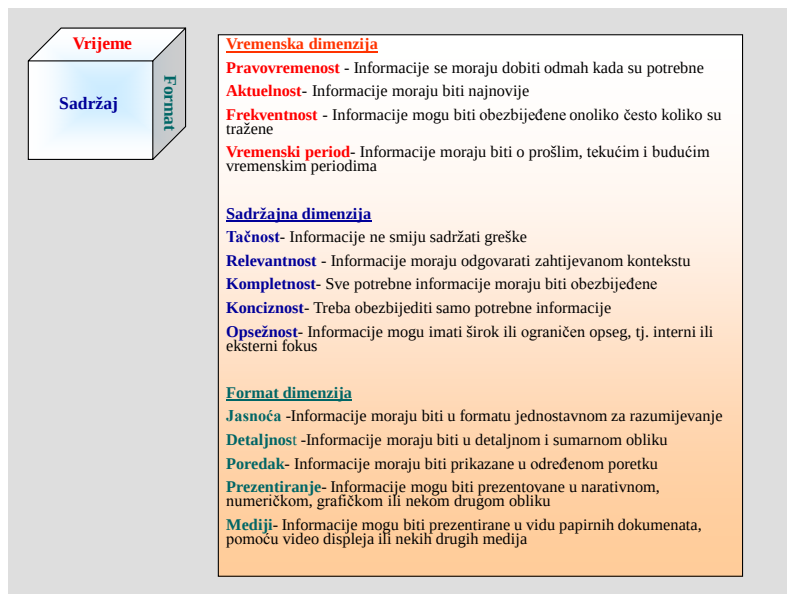
Output. – Suštinu output aktivnosti čine informacije koje se krajnjim korisnicima dostavljaju u različitim formama. Cilj IS jeste proizvodnje odgovarajućih informacija za krajnje korisnike. Informacije mogu biti u vidu poruka, izvještaja i grafičkih prikaza, a prezentiraju se pomoću video displeja, audio uređaja, papirnih proizvoda ili multimedije.

Na primjer, menadžer prodaje pomoću video displeja može da provjeri performanse pojedinih prodajnih mjesta, može primiti kompjuterski generisanu govornu poruku preko telefona ili dobiti štampani izvještaj o rezultatima mjesečne prodaje.

Veoma bitna performansa output-a jeste kvalitet dobijenih informacija. Nepravovremeni, netačni ili teško razumljivi podaci ne mogu služiti korisniku. Korisnici zahtijevaju informacije visokog kvaliteta koje će za njih biti korisne. Stoga informacije moramo posmatrati kroz tri dimenzije – vrijeme, sadržaj i format. Na slici 1.3. prikazani su najvažniji atributi kvaliteta informacija grupisani u ove tri dimenzije.

Memorisanje podataka. – Memorija je bazična komponenta IS. Memorisanje predstavlja aktivnost IS pomoću koje se podaci i informacije čuvaju u organizovanom obliku za kasniju upotrebu. Podaci su hijerarhijski organizovani u vidu karaktera, polja, slogova ili zapisa (*recorda*), datoteka (fajlova) i, na kraju, baza podataka.

- **Karakter** jeste osnovna elementarna jedinica podatka i predstavlja jedan znak na tastaturi (slovo, cifru ili specijalni znak, kao što su tačka, zarez i slično).
- **Polje** jeste grupa karaktera koji predstavljaju jedan atribut (karakteristiku) neke osobe, mjesta, stvari ili događaja (na primjer ime studenta).
- **Slog** predstavlja skup povezanih polja (na primjer, slog student može se sastojati od polja ime, broj indeksa, godina studija, adresa i telefon).
- **Fajl** predstavlja kolekciju međusobno povezanih slogova (na primjer, fajl studenti sastoji se od slogova student za sve studente fakulteta).



Slika 1.3. Tri dimenzije informacija

- **Baza podataka** jeste integrisani skup međusobno povezanih slogova ili fajlova (na primjer, baza podataka fakulteta može da sadrži fajlove studenti, profesori, predmeti i slično).

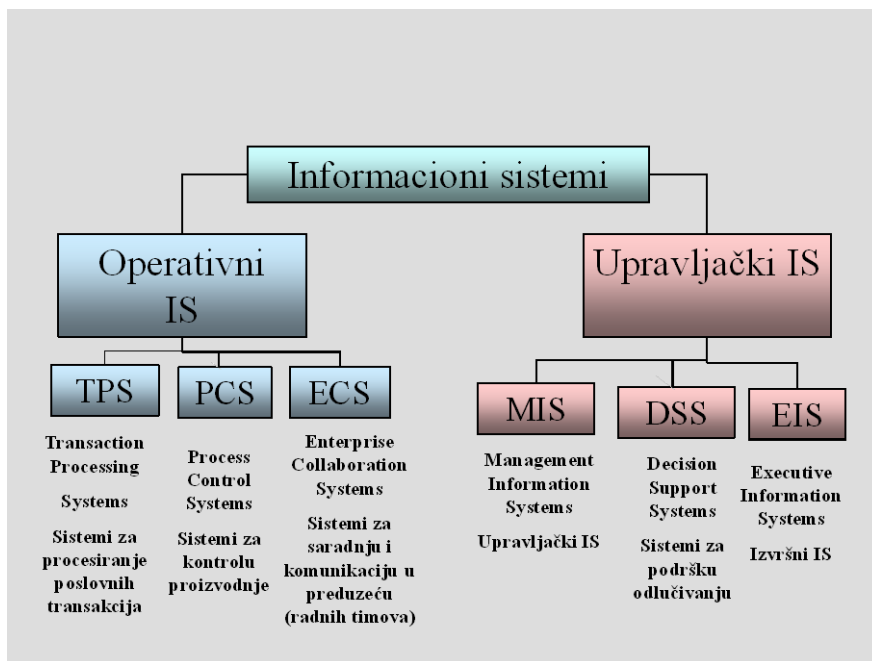
Kontrola. – Veoma važna aktivnost IS jeste kontrola sopstvenih performansi. IS produkuje feedback o input i output aktivnostima, procesiranju i memorisanju. Taj se feedback mora nadzirati i ocjenjivati kako bi se utvrdilo da sistem postiže standardne performanse.

Na taj se način aktivnosti sistema kontrolišu i usmjeravaju da proizvode prave informacije za korisnike.

Na primjer, menadžer prodaje može uočiti da neki suptotal o iznosu prodaje nije dodat na konačni total iznosa prodaje. Ovo može značiti da unos podataka i/ili procedure njihove obrade u IS nisu korektni. Nužno je uraditi odgovarajuće promjene, kako bi se osigurala tačnost informacija koje produkuje ovaj IS prodaje.

1.2.2. VRSTE IS

Informacioni sistemi mogu biti klasifikovani na više načina. Ovdje ćemo prihvatiti generalnu klasifikaciju na **operativne** i **upravljačke informacione sisteme**. Slika 1.4. prikazuje koncept ove klasifikacije.



Slika 1.4. Vrste informacionih sistema

Uloga operativnih IS jeste pružanje podrške u obavljanju poslovnih procesa u okviru firme, što podrazumijeva efikasno procesiranje poslovnih transakcija, praćenje i kontrolu proizvodnih procesa, kao i omogućavanje efikasnih komunikacija i saradnje u preduzeću.

Transaction processing systems (TPS), tj. **sistemi za procesiranje poslovnih transakcija** čuvaju i obrađuju podatke nastale u okviru poslovnih transakcija (prodaje, nabavke, upravljanja zalihama, računovodstva i drugih), ažuriraju operativnu bazu podataka i generišu poslovna dokumenta i izvještaje. Obrada može biti **batch** ili **on-line**. Kod **batch** obrade podaci se akumuliraju neki vremenski period, a onda periodično obrađuju (krajem mjeseca, noću i drugo). Ovakav način obrade podataka naziva se još i **off-line**. Kod **on-line** obrade, koja je poznata i pod imenom **real-time**, podaci se procesiraju odmah u toku same transakcije.

Primjer: POS (point-of-sale) predstavlja on-line sistem u prodajnim objektima kod koga se pomoću elektronskih *cash register* terminala (bar kod) podaci o prodaji odmah, u toku obavljanja prodaje, elektronski snimaju i putem telekomunikacionih kanala odašilju prema centralnom računar.

Upravljački IS obezbjeđuju informacije i podršku odlučivanju za sve nivoe menadžmenta od izvršnog do strateškog.

Management information systems (MIS) ili **upravljački IS** obezbjeđuju informacije u vidu unaprijed definisanih izvještaja na osnovu podataka iz operativne baze, tj. aktuelnih podataka. Ovi izvještaji, za razliku od izvještaja TPS sistema, ne sadrže informacije o pojedinačnim poslovnim transakcijama, već su oni obično agregirani i na neki način logički grupisani (strukturirani). Koristi ih uglavnom izvršni menadžment za strukturirane probleme odlučivanja. Na primjer, menadžer prodaje odlučuje o količini nekog proizvoda koju će naručiti za naredni mjesec. Za ovu vrstu odluke dovoljan je izvještaj o prodaji tog proizvoda za prethodni mjesec.

Decision support systems (DSS) ili **sistemi za podršku odlučivanju** daju kompjutersku podršku menadžerima u procesu odlučivanja. Oni omogućavaju napredne analize podataka i simulacije pomoću odgovarajućih modela kod polustrukturiranih i nestrukturiranih problema. Ovi su sistemi interaktivni i *ad hoc*, što znači da korisnik može po potrebi mijenjati pretpostavke, uključivati nove podatke i drugo.

Koriste uglavnom specijalne višedimenzionalne baze podataka kao i data *warehouse* baze, koje ne sadrže samo tekuće, tj. aktuelne podatke već i istorijske, tj. podatke iz proteklih perioda poslovanja. Ovi podaci mogu biti smješteni u različitim dislociranim operativnim bazama i u različitim formatima, a mogu se koristiti i eksterni podaci kao i meki, tj. subjektivni podaci definisani od strane eksperta. Dok je za transakcione sisteme karakterističan način obrade OLTP, za ove sisteme koristi se **OLAP**, tj. **on-line analitical processing**.

Ekspertni sistemi (ES) čine posebnu kategoriju DS sistema koji automatski generišu sugestije donosiocima odluka za neki određeni (nestrukturirani) problem. Pri tome koriste bazu znanja (koju definiše ekspert iz te oblasti) i tehnike vještačke inteligencije (mašinsko učenje, neuronske mreže itd.) za izvođenje zaključaka. Ovi sistemi još se nazivaju i **sugestivni** ili **inteligentni DSS**. Posebnu kategoriju ekspertnih sistema čine **data mining**

sistemi koji znanja ne izvode iz baze znanja, već iz baze podataka koristeći iste tehnike vještačke inteligencije. Ovi sistemi automatski otkrivaju veze koje postoje između podataka u velikim bazama, tj. otkrivaju skrivene prediktivne informacije, i trenutno su veoma popularni.

Executive information systems (EIS) ili **izvršni informacioni sistemi** obezbjeđuju kritične informacije za izvršni menadžment u formatu jednostavnom za upotrebu. Oni obično upozoravaju na pojavu kritičnih situacija i omogućavaju jednostavne i efikasne analize tih situacija. Mogu se smatrati podskupom DS sistema, s tom razlikom što EIS uglavnom koriste tekuće, tj. aktuelne podatke.

Na primjer, menadžer prodaje koristi izvještaj o rezultatima prodaje proizvoda svoje kompanije razmatrajući iznos prodaje na dnevnom nivou po prodajnim mjestima. Ovdje je riječ o tipičnom MIS izvještaju jer je problem strukturiran, a format izvještaja unaprijed poznat. Međutim, ako menadžer marketinga želi da testira uticaj različitih iznosa budžeta marketinga na predviđanje prodaje novog proizvoda, onda je riječ o jednoj what-if analizi koja nije unaprijed definisana, već se ona mora raditi na jednom specijalno dizajniranom modelu. U ovom slučaju riječ je o DSS-u. Ako na ekranu stoji mapa prodajnih mjesta na kojoj se u trenutku pada iznosa prodaje ispod definisane granice na nekom čvorištu pojavi crvena signalna lampica, onda je riječ o EIS-u. Tada menadžer prodaje može pokrenuti model za višedimenzionalnu analizu prodaje na tom prodajnom mjestu kako bi utvrdio uzroke pada (prema proizvodima, kupcima, vremenskim periodima itd.).

1.3. RAČUNARSKI HARDVER

Danas se računari javljaju u različitim veličinama, oblicima i sa različitim mogućnostima. Rapidni razvoj računarskog hardvera i softvera, kao i promjene u zahtjevima korisnika nužno izazivaju pojavu novih modela kompjutera, počev od malih ručnih PDA uređaja, pa sve do najvećih multiprocesorskih mainframe računara za velika preduzeća.

Kategorije kao što su *mainframe*, *midrange* i *mikrokompjuteri* još uvijek se koriste da bi se izrazila snaga procesiranja i broj korisnika koje podržavaju pojedini tipovi računara. Međutim, ovakva klasifikacija nije precizna jer postoje preklapanja. Zato se koristi i čitav niz drugih imena za pojedine

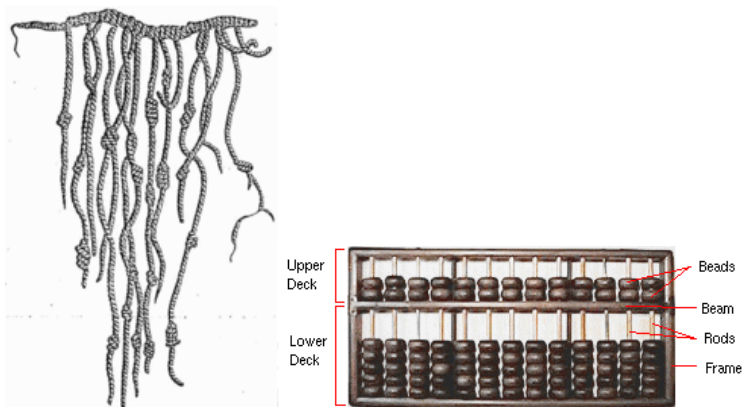
tipove kompjutera, kao što su: personalni računari, mrežni serveri, serveri baza podataka, radne stanice, mrežni kompjuteri i drugo.

Eksperti pokušavaju da predvide kretanja na kompjuterskom tržištu. Mnogi od njih smatraju da su midrange i mainframe bazirani sistemi zastarjeli u odnosu na snažne i mnogostrane klijent/server mreže bazirane na mikrokompjuterima i serverima. Zapravo, u novije vrijeme neki industrijski eksperti predviđaju da će mrežni kompjuteri i drugi informacijski uređaji, koji se koriste kod interneta i kompanijskih intranet mreža, zamijeniti mnoge personalne računare, posebno u velikim organizacijama, kao i na tržištu kućnih računara.

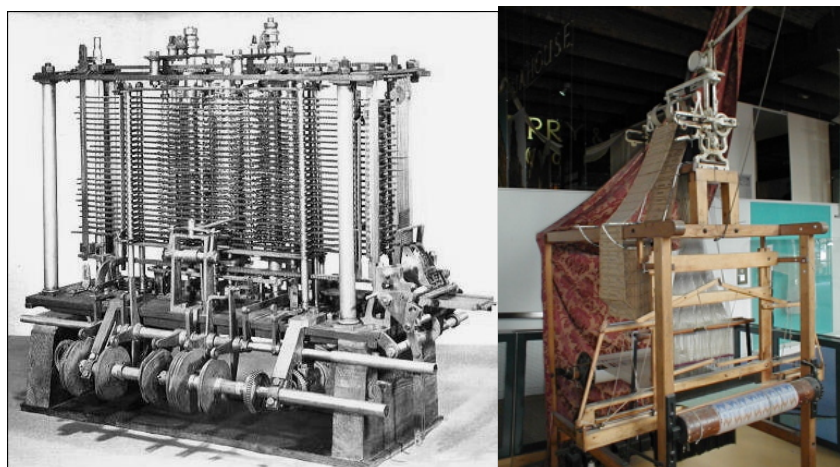
Istorijski razvoj računarskih sistema. – Prva ručna pomagala za memorisanje podataka i računске operacije pojavila su se još prije 5000 godina. Naime, pleme Inka koristilo je QUIPU (slika 1.5. lijevo) kao sredstvo za memorisanje zaliha žita. Riječ je o sistemu pamučnih konopa kojima su predstavljeni brojevi dekadnog brojnog sistema, čije su pozicije kodirane čvorovima na odgovarajućim mjestima. U Aziji su se koristili ručni kalkulatori tzv. ABACUS (slika 1.5. desno) za pomoć pri obavljanju aritmetičkih operacija. Ovi kalkulatori slični su računaljki, i u Japanu se još i danas koriste u školama.

Prve mehaničke mašine za obavljanje računskih operacija konstruisao je engleski matematičar Charles Babbage početkom 19. vijeka. Njegova analitička mašina (slika 1.6. lijevo) predstavlja prvu računsku mašinu kojom su upravljale programske instrukcije zadate na bušenim karticama. Prvi program za njegovu analitičku mašinu napisala je matematičarka Ada Byron (ćerka poznatog engleskog pjesnika). To je zapravo bio plan izračunavanja Bernulijevih brojeva, i to se smatra prvim programom za računar, a Ada Byron – prvim svjetskim programerom. U njenu čast, u Americi je jedan programski jezik nazvan Ada.

Zanimljivo je da se ocem programibilnosti smatra francuski izumitelj i tkalac Joseph Marie Jacquard, koji je stvorio programibilni tkalački razboj (slika 1.6. desno), napravio revoluciju u tkalačkoj industriji i postavio osnove moderne obrade podataka. Ovaj razboj koristio je bušene kartice kao binarno uputstvo za tkanje ćilima. Njegove bušene kartice Charles Babbage samo je adaptirao i iskoristio ih za svoju analitičku mašinu.



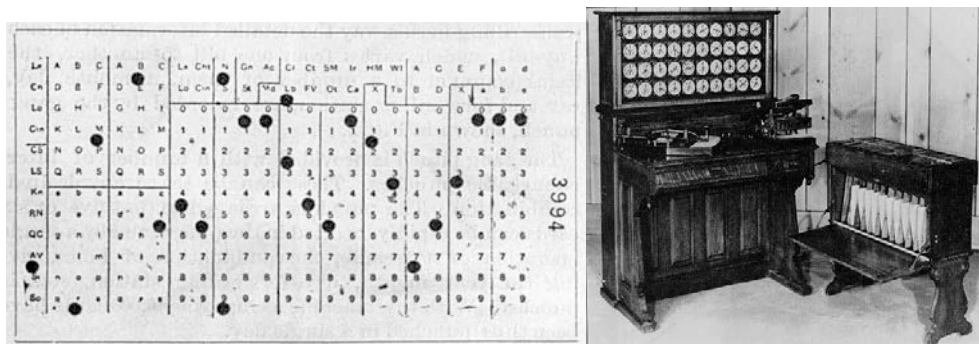
Slika 1.5. QUIPU i ABACUS – prva ručna pomagala za memorisanje i izračunavanje



Slika 1.6. Bebidžova analitička mašina i Žakarov razboj – preteče programibilnih računara

Prve elektromehaničke računske mašine izumio je američki statističar Herman Hollerith za potrebe popisa stanovništva 1890. godine. Ove mašine su koristile bušene kartice (slika 1.7. lijevo) za evidenciju podataka pri popisu, a podatke sa bušenih kartica očitavali su električni uređaji. Poslije uspješno obavljenog popisa, ove mašine su postale popularne, pa je Hollerit osnovao sopstvenu kompaniju za proizvodnju tabelarnih mašina (slika 1.7. desno). Ova kompanija je kasnije, 1924. godine, pod rukovodstvom

Thomasa Watsona postala čuveni IBM, koji je i danas apsolutni lider na tržištu računarskih sistema (slika 1.8.).



Slika 1.7. Hollerithove bušene kartice i tabelarna mašina

Elektronsko doba za razvoj računarskih sistema počinje 40-ih godina dvadesetog vijeka. Prvi elektronski računar bio je ENIAC (Electronic Numerical Integrator and Computer) – slika 1.9. lijevo, koji je konstruisan 1946. godine za potrebe američke vojske. Sastojao se od 18000 vakuumskih cijevi (slika 1.9. desno) i bio težak oko 27 tona, a zauzimao je prostor od 170 m². Računar nije imao mogućnost pohranjivanja programa, a svakog dana otkazivalo je po nekoliko vakuumskih cijevi.

From Computer Desktop Encyclopedia
Reproduced with permission.
© 1996 International Business Machines Corporation
Unauthorized use not permitted.



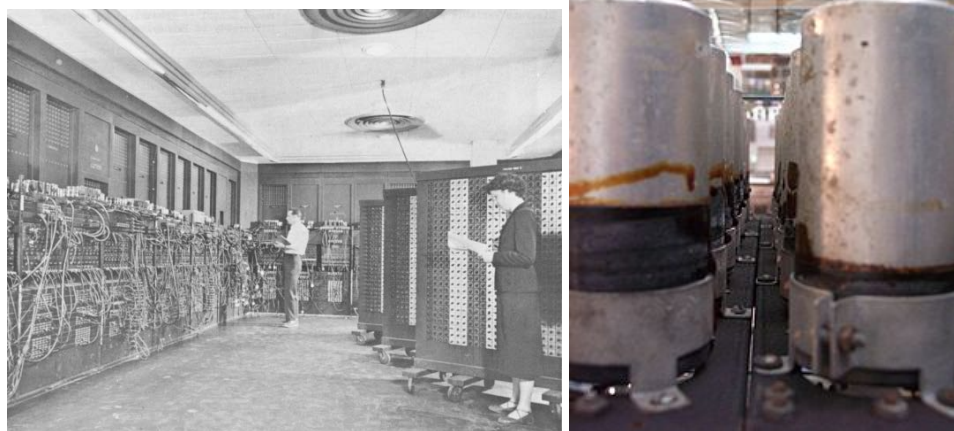
Slika 1.8. Početak IBM-a

Zanimljivo je porijeklo termina za računarsku grešku *bug*, koji potiče iz tog vremena kada su kvarove vakuumskih cijevi obično izazivale bube. Obrada podataka kod ENIAC računara bila je u dekadnom brojnom sistemu. Njegov nasljednik EDVAC bio je prvi digitalni računar, sa pohranjenim programom i magnetnim trakama. Prvi računar opšte namjene za komercijalnu upotrebu bio je UNIVAC i njegov prvi primjerak prodat je američkom popisnom birou 1951. godine. Računar je za unos podataka koristio bušene kartice. Unutrašnje memorisanje podataka obavljano je na magnetnim dobošima, a spoljašnje memorisanje – na bušenim karticama i magnetnim trakama. Za programiranje su korišćeni mašinski i asemblerški programski jezici.

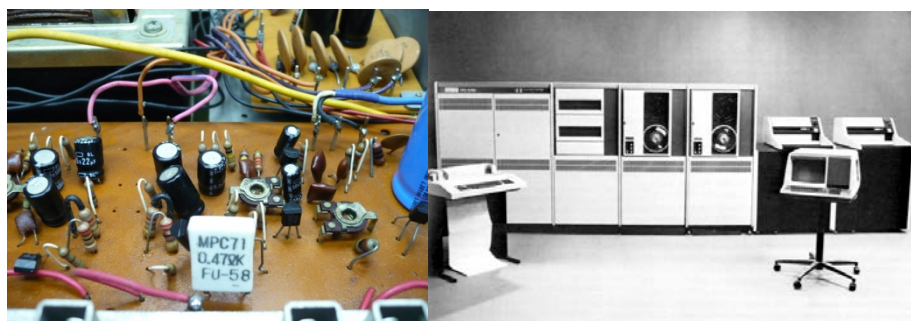
Pojavom tranzistora i njihovom upotrebom kod računara značajno su smanjene dimenzije računarskih sistema (slika 1.10.).

Za unutrašnje memorisanje podataka koriste se magnetna jezgra, a kao spoljašnje memorije pojavljuju se magnetni diskovi. Za programiranje se počinju koristiti viši programski jezici. Pojavljuju se i tastatura i monitor.

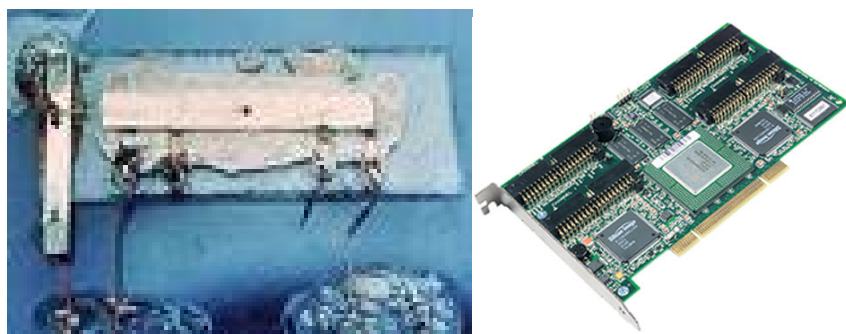
Prekretnicu u razvoju elektronskih digitalnih računara predstavlja pojava mikročipova, tj. integrisanih elektronskih kola.



Slika 1.9. ENIAC i njegove vakuumske cijevi



Slika 1.10. Računari zasnovani na tranzistorima imaju manje dimenzije



Slika 1.11. Kilbyjev prvi mikročip i VLSI mikročip

Pronalazačima mikročipova smatraju se Jack Kilby, koji je 1958. godine konstruisao prvi mikročip na germanijumskoj pločici (slika 1.11. lijevo), i Robert Nojs, koji je germanijumsku ploču zamijenio silicijumskom, a žice samim silicijumom. Dalji razvoj mikročipova doveo je do pojave LSI (Large-Scale Integrated) mikročipova sa gustom od nekoliko desetina hiljada tranzistora po čipu, kao i do pojave VLSI (Very-Large-Scale Integrated) mikročipova, koji danas imaju gustinu i do nekoliko milijardi tranzistora po čipu (slika 1.11. desno). Unutrašnje magnetne memorije zamijenjene su elektrostatičkim memorijama, koje se proizvode u vidu mikročipova. Zahvaljujući malim dimenzijama mikročipova, današnji moćni i snažni računari imaju veoma male dimenzije.

Generacije elektronskih računarskih sistema. Glavne promjene i trendovi elektronskih kompjuterskih sistema odvijale su se kroz pet generacija. Prva

generacija računara razvijena je ranih 50-ih godina, druga se pojavila kasnih 60-ih, treća je vezana za 70-e godine, dok četvrta generacija pripada tehnologiji 80-ih i 90-ih godina. Peta generacija kompjuterskih sistema i uređaja vezana je za tehnologiju početka 21. vijeka. Tabela 1.1. prikazuje karakteristike i mogućnosti kompjutera koji pripadaju ovim generacijama. Primjećuje se da su osnovni trendovi vezani za to da računari postaju manji, brži, pouzdaniji, jeftiniji, lakši za upravljanje i povezivanje na računarske mreže. Promjene u kompjuterskim tehnologijama i dalje će se odvijati ubrzano, posebno pod uticajem internet baziranih tehnologija i primjena.

Tabela 1.1.: Trendovi u razvoju računarskog hardvera

	Prva generacija	Druga generacija	Treća generacija	Četvrta generacija	Peta generacija
VELIČINA	Mainframe (veličine sobe)	Mainframe (veličine ormara)	Minikomputeri (veličine stola)	Mikomkompjuteri (Desktop/Laptop)	Mrežni kompjuteri (u svim veličinama)
UMREŽAVANJE	Nema	Mainframe bazirane mreže video terminala	Mainframe i minicomputer bazirane mreže	LAN i klijent/server mreže	Internet, intranet i extranet mreže.
STRUJNO KOLO	Katodne (vakumske) cijevi	Tranzistori	Integrirana kola	LSI (Large-Scale Integrated) kola	VLSI (Very-Large-Scale Integrated) kola
GUSTINA (kola po komponenti)	Jedno	Stotine	Hiljade	Stotine hiljada	Milioni i milijarde
BRZINA (instrukcija/sec)	Stotine	Hiljade	Milioni	Desetine miliona	Bilioni
POUZDANOST (greška jednom u nekoliko)	Sati	Dana	Nedjelja	Mjeseci	Godina
MEMORIJA (kapacitet u karakterima)	Hiljade	Desetine hiljada	Stotine hiljada	Milioni	Bilioni
TROŠKOVI (za milion instrukcija)	\$10	\$1.00	\$10	\$0.001	\$0.0001

1.3.1. MIKROKOMPJUTERSKI SISTEMI

Mikomkompjuteri su najvažnija kategorija računarskih sistema za poslovne korisnike. Mada se često nazivaju personalni računari, tj. PC računari, mikromkompjuteri su mnogo više od malih kompjutera za ličnu upotrebu. Moć mikromkompjutera danas prevazilazi mainframe kompjutere iz

prethodnih generacija, iako predstavljaju samo djelić njihove cijene. Oni danas postaju moćne mrežne i profesionalne radne stanice za poslovne korisnike.

Mikrokompjuteri se mogu naći u različitim veličinama i oblicima, zavisno od potreba korisnika. Na primjer, PC mogu biti u obliku ručnih uređaja, notebook, laptop, portable, desktop i podnih modela. Ili, na osnovu namjene, mogu biti kućni, lični, profesionalni, radne stanice ili višekorisnički računari. Većina mikroračunara dizajnirana je u vidu desktop modela koji je namijenjen kancelarijskom stolu, ili u vidu laptop modela za one koji žele mali i portabilni PC za svoje radne aktivnosti.

Neki mikrokompjuteri predstavljaju moćne radne stanice koje podržavaju aplikacije za matematička izračunavanja, grafičke displeje, kompjutersko dizajniranje proizvodnje (CAD), investicione i portfolio analize i drugo. Mikrokompjuteri se koriste i kao mrežni serveri. To su obično snažniji mikroračunari koji omogućavaju upravljanje telekomunikacijama i dijeljenjem resursa u malim lokalnim (LAN) mrežama, kao i internet i intranet veb-sajtovima.

Mrežni kompjuteri. – Network computers (NC) smatraju se neizostavnom poslovnom kompjuterskom platformom. NC je poznat kao „tanki klijent” (sa siromašnim hardverskim performansama), za razliku od tradicionalnih „debelih klijenata” – PC računara sa full opremom.

NC predstavlja mikrokompjutersku kategoriju dizajniranu prvenstveno za upotrebu u internet i kompanijskim intranet mrežama, bez disk memorije ili sa minimalnom disk memorijom i niskom cijenom. NC mikrokompjuteri koriste operativni sistem, *web browser*, Java aplikacije i disk memorije, koji se nalaze na internet ili intranet serverima. Osnovna prednost NC računara jesu veoma niski troškovi nabavke, održavanja i podrške u poređenju sa full opremljenim PC računarima. Ostale prednosti u poslovanju ogledaju se u jednostavnosti distribucije i licenciranja softvera, standardizaciji kompjuterske platforme, reduciranju podrške korisničkim zahtjevima, centralizovanom menadžmentu i kontroli kompjuterskih mrežnih resursa.

NC računari razlikuju se od mrežnih PC računara (NetPC) koji obično imaju sopstveni hard disk i mogu raditi sa sopstvenim softverom. Za razliku od standardnih PC računara, NetPC najčešće nemaju floppy i CD diskove, kao

ni serijske i paralelne portove. Sa druge strane, NC računare treba razlikovati i od kompjuterskih terminala (tastatura/monitora) koji su zapravo samo ulazno-izlazni uređaji u mreži. Naime, NC računari mogu izvršavati djelove Java aplikacije, tj. aplete.

Informacioni uređaji. – Ručni mikrokompjuteri, poznati kao **personal digital assistants (PDA)**, najpoznatiji su u kategoriji informacionih uređaja. PDA koriste dodirne ekrane (touch screen) i na olovci bazirano upravljanje (prepoznavanje ručnog ispisivanja) ili tastaturu. Namijenjeni su mobilnim korisnicima za prijem i slanje e-maila, pristup webu, memorisanje i razmjenu informacija o sastancima, obavezama, kontaktima i drugo.

Informacioni uređaji pojavljuju se i u obliku video-game konzole koja se priključuje na kućni TV uređaj. Ovaj uređaj omogućava surfovanje po internetu, slanje i prijem e-mailova, gledanje TV programa i igranje video igara. Primjeri takvih uređaja jesu Sony ili Philips WebTV Plus Receiver, koji koriste Microsoftov web TV mrežni servis. U ostale informacione uređaje spadaju i mobilni celularni i PCS telefoni, kao i žičani telefonski bazirani kućni uređaji koji mogu slati i primati e-maileve i pristupati webu.

Dakle, PC nije jedina opcija. Mnoge pametne sprave kao što su mobilni telefoni i pejdžeri, ručni PC i web bazirane video-game mašine omogućavaju pristup internetu i na taj način obavljaju danas gotovo najvažniju ulogu računara.

Kompjuterski terminali. – *Glupi terminali* sa tastaturom i video monitorom, sa nemogućnošću procesiranja informacija, danas se zamjenjuju tzv. *inteligentnim terminalima*, a to su zapravo standardni ili NetPC računari, mrežni kompjuteri (NC) ili neki drugi „tanki klijenti“. Pojavljuju se u obliku tzv. **mrežnih terminala** koji mogu biti *Windows terminali*, koji zavise od mrežnih servera jer se na njima nalazi i izvršava Windows softver, i disk memorijski resursi ili *internet terminali*, koji zavise od internet ili intranet veb-servera na kojima se nalaze operativni sistem i aplikativni softver.

Inteligentni terminali pojavljuju se u različitim formama i obavljaju unos podataka i određena procesiranja informacija zavisno od zadatka. Tu spadaju i veoma rasprostranjeni **transakcioni terminali**, koji se koriste u bankama, prodajnim mjestima, fabrikama itd. Primjeri ovih terminala jesu **point-of-sale (POS)** terminali, koji koriste tastaturu, dodirni ekran i razne

druge metode prikupljanja podataka i interakcije sa korisnikom u toku prodajne transakcije, dok se za procesiranje transakcije oslanjaju na server ili neki drugi kompjuter u mreži.

1.3.2. MIDRANGE KOMPJUTERSKI SISTEMI

Midrange kompjuteri, u koje spadaju mrežni serveri i minikompjuteri, predstavljaju obično višekorisničke sisteme koji upravljaju mrežom PC računara i terminala.

Midrange kompjuteri se koriste kao moćni mrežni serveri koji omogućavaju upravljanje velikim internet veb-sajtovima, korporacijskim intranet, ekstranet i klijent/server mrežama. Elektronska trgovina i druge poslovne primjene interneta, kao i integrisane aplikacije unutrašnjeg poslovanja (proizvodnja, distribucija, finansije i drugo), upravljanje data warehouseom, data mining i online analytical processing (OLAP) utiču na sve veće zahtjeve u pogledu hardverskih performansi serverskog računara. To su tipične aplikacije koje zahtijevaju *high-end* servere (sa visokim hardverskim performansama), tj. midrange kompjuterske sisteme.

Midrange kompjuteri u početku su se koristili kao minikompjuteri za naučna istraživanja, inženjerske analize i procese praćenja i kontrole proizvodnje. Minikompjuteri su se pokazali povoljnim za ove aplikacije jer su one namjenske i nisu podrazumijevale zahtjevna procesiranja koja obavljaju mainframe sistemi. Oni i danas igraju glavnu ulogu kod computer-aided manufacturing (CAM) i computer-aided design (CAD) aplikacija. Midrange kompjuteri koriste se i kao *front-end* kompjuteri koji pomažu mainframe kompjuterima u upravljanju mrežom.

1.3.3. MAINFRAME KOMPJUTERSKI SISTEMI

Mainframe kompjuteri jesu veliki, brzi i moćni kompjuterski sistemi koji procesiraju stotine miliona instrukcija po sekundi i imaju primarnu memoriju velikog kapaciteta, ranga od stotine megabajta do nekoliko gigabajta. Mainframe sistemi su, u posljednjih nekoliko godina, značajno unaprijeđeni u pogledu reduciranja potrebe za hlađenjem, trošenja električne energije i smještajnih kapaciteta. Na taj način smanjeni su troškovi nabavke i održavanja ovih inače skupih sistema. Ova unapređenja uglavnom su rezultat zamjene na vodi baziranog sistema hlađenja sa vazдушnim hlađenjem.

Mainframe sistemi se još uvijek koriste za potrebe velikih korporacija ili vladinih agencija koje procesiraju veliki broj transakcija ili obavljaju kompleksna izračunavanja. Na primjer, velike internacionalne banke, avio i naftne kompanije i druge velike korporacije procesiraju milione prodajnih transakcija i korisničkih zahtjeva svaki dan zahvaljujući velikim mainframe sistemima. Ovi sistemi se takođe koriste za zahtjevne aplikacije kao što su analize seizmoloških podataka nastalih u oblasti naftnih nalazišta ili za simulacije uslova letjenja pri dizajniranju u avijaciji. Mainframe sistemi se koriste i kao superserveri kod velikih klijent/server mreža, kao i za upravljanje obimnim internet veb-sajtovima velikih kompanija. Takođe, mainframe sistemi predstavljaju popularnu poslovnu platformu za data mining, warehousing i elektronsku trgovinu.

Termin **superkompjuteri** koristi se za jednu kategoriju mainframe sistema sa izuzetnom snagom, specijalno dizajniranu za naučne, inženjerske i poslovne primjene, koje zahtijevaju ekstremno veliku brzinu za masovna numerička izračunavanja. Tržište za superkompjutere predstavljaju vladine istraživačke agencije, veliki univerziteti i velike korporacije. Oni koriste superkompjutere za globalne vremenske prognoze, vojne odbrambene sisteme, kosmološka i astronomska izračunavanja, istraživanje i dizajniranje mikroprocesora, zahtjevni data mining i slično.

Superkompjuteri koriste višeprosorsku arhitekturu, tj. rade sa više međusobno povezanih mikroprocesora, koji paralelno izvršavaju veliki broj instrukcija u jednom vremenskom trenutku. Oni mogu izvršavati aritmetička izračunavanja brzinom od nekoliko biliona operacija sa realnim (float) brojevima u sekundi (gigaflops). Superkompjuteri koji koriste masivno paralelno procesiranje – MPP (sa hiljadama povezanih procesora) mogu procesirati i do jednog trilion float operacija po sekundi (teraflops). Nabavna cijena superkompjuteri kreće se od 5 miliona do 50 miliona dolara.

1.3.4. KONCEPT RAČUNARSKOG SISTEMA

Računar predstavlja mnogo više od skupa elektronskih uređaja koji procesiraju informacije. Računar je sistem, tj. kombinacija međusobno povezanih komponenti koje obavljaju osnovne funkcije ulaza, procesiranja, izlaza, memorisanja i kontrole, obezbjeđujući na taj način korisniku moćan alat za obradu informacija. Za uspješno korišćenje i upravljanje računarom od ključnog značaja jeste shvatanje računara kao računarskog sistema. Na

ovaj način računar se može zamisliti kao najmanji mikroračunarski uređaj, ali i kao velika računarska mreža čije su komponente međusobno povezane telekomunikacionim mrežnim linkovima unutar jednog kompleksa zgrada ili čak šireg geografskog područja.

Slika 1.12. prikazuje da je računar sistem hardverskih uređaja organizovanih prema sljedećim funkcijama sistema:

Ulaz (Input). – U ulazne uređaje računarskog sistema spadaju tastatura, dodirni ekrani (touch screen), olovke, elektronski miš, optički skener i slično. Oni konvertuju podatke u elektronsku formu, bilo da su dobijeni kroz direktni unos ili preko računarske mreže.

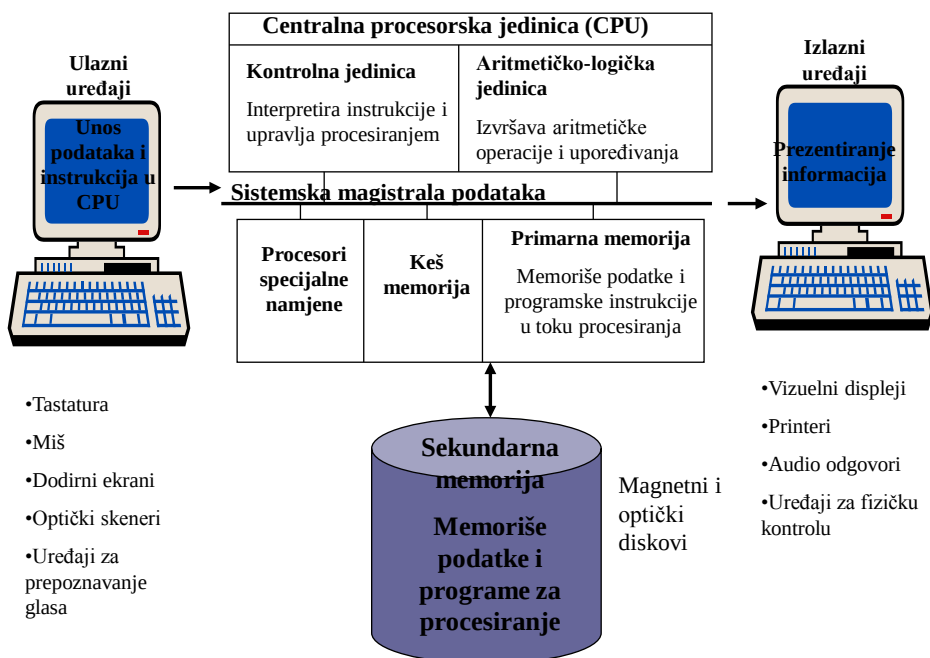
Obrada (Processing). – Centralna procesorska jedinica (Central Processing Unit – CPU) jeste glavna komponenta za obradu podataka u računarskom sistemu. Kod mikroračunara ova jedinica se naziva mikroprocesor. Konceptualno, CPU se može podijeliti u dvije glavne komponente, a to su aritmetičko-logička jedinica i kontrolna jedinica. Elektronsko kolo aritmetičko-logičke jedinice izvršava aritmetičke i logičke funkcije koje su potrebne prilikom izvršavanja softverskih instrukcija. CPU sadrži i **registre** i **keš memoriju**, koji predstavljaju veoma brzu i privremenu memoriju za instrukcije i podatke. Takođe, može sadržati i neke pomoćne procesore specijalne namjene.

Izlaz (Output). – Izlazni uređaji računarskog sistema obuhvataju monitore (uređaje sa video displejom), štampače (printere), audio izlazne uređaje i slično. Oni konvertuju elektronske informacije generisane računarskim sistemom u format pogodan za prezentiranje korisniku.

Memorisanje (Storage). – Uređaj koji memoriše (čuva) podatke i softverske instrukcije u toku obrade jeste **primarna** memorijska jedinica, koja se naziva **radna (operativna) memorija**. Računarski sistem osim primarne memorije ima i **sekundarne** memorijske uređaje, kao što su magnetni ili optički diskovi. Oni trajno memorišu podatke i programe koji se obrađuju na računaru.

Kontrola. – Kontrolna jedinica centralne procesorske jedinice je kontrolna komponenta računarskog sistema. Ovo elektronsko kolo interpretira softverske instrukcije i u skladu sa tim upućuje odgovarajuće komande svim komponentama računarskog sistema.

Brzina procesiranja. – Brzina procesiranja govori nam koliko je brz računarski sistem, tj. koliko brzo obrađuje instrukcije. Nekada se brzina računara mjerila milisekundama (hiljaditi dio sekunde) ili mikrosekundama (milioniti dio sekunde). Danas računari rade u rang nanosekundi (bilioni dio sekunde), pa čak i picosekundi (trilioniti dio sekunde). Ove brzine su gotovo neshvatljive. Kolika je to brzina, najbolje može ilustrovati sljedeći primjer: kada bi se pravio po jedan korak svake nanosekunde, Zemlja bi se obišla 20 puta u toku jedne sekunde.



Slika 1.12. Koncept računarskog sistema

Brzina računara može se izraziti i brojem obavljenih operacija u sekundi. Već smo pomenuli da superkompjuteri mogu imati brzinu od jednog trilion operacija po sekundi (teraflop). Međutim, većina računara danas ima brzinu procesiranja programskih instrukcija izraženu u milionima instrukcija po sekundi (MIPS).

Za mjerenje brzine računarskog sistema koriste se i druge jedinice, kao što su: megaherci (MHz), tj. milion ciklusa po sekundi, ili gigaherci (GHz), tj. bilion ciklusa po sekundi. Ovdje je riječ o vremenskim ciklusima mikroprocesora zasnovanim na internom časovniku. Zato se ova brzina često zove i **clock speed** mikroprocesora.

Međutim, ovako izražene brzine računarskog sistema mogu biti i nedovoljno tačni indikatori stvarne brzine procesiranja mikroprocesora. Naime, brzina procesiranja zavisi i od mnoštva drugih faktora, kao što su: dužina ciklusa, magistrala koja povezuje mikroprocesorske komponente, kapacitet registara za procesiranje instrukcija, korišćenje ultra brze keš memorije ili specijalizovanih pomoćnih procesora, kakav je npr. matematički koprocessor za brza aritmetička izračunavanja i drugo.

1.3.5. RAČUNARSKE PERIFERIJE: ULAZNE, IZLAZNE I MEMORIJSKE TEHNOLOGIJE

Periferije su zapravo generalni naziv za sve ulazne, izlazne i sekundarne memorijske uređaje, koji su sastavni dio računarskog sistema. Periferije se povezuju sa centralnom procesorskom jedinicom direktno ili preko telekomunikacionih linkova. Periferije predstavljaju izdvojene on-line uređaje, koji su elektronski povezani sa CPU i pod njenom su kontrolom. Primjeri računarskih periferija jesu monitori, štampači, skeneri, hard disk uređaji, CD-ROM i DVD drajveri i drugo.

Ulazne tehnologije i trendovi. – Osnovni trend kod ulaznih tehnologija jeste obezbjeđivanje što prirodnijeg i jednostavnijeg korisničkog interfejsa. Nekada su osnovne ulazne medijume predstavljali bušene kartice i papirne trake, kao i unos putem tastature. Danas podatke možemo jednostavno unositi direktno u računar pomoću pokaznih uređaja kao što su elektronski miš i touch pad (male ravne ploče osjetljive na dodir), kao i pomoću tehnologija poput optičkog skeniranja, prepoznavanja rukopisa ili glasa. Ovakav razvoj ulaznih tehnologija eliminisao je nužnost čuvanja podataka na izvornim papirnim dokumentima i njihovog unosa preko tastature.

Pokazni uređaji. – Tastatura je još uvijek najrasprostranjeniji uređaj za unos podataka i teksta. Pokazni uređaji su bolja alternativa za zadavanje komandi, pravljenje izbora i odgovore na promptne displeje na video ekranu. Koriste se kod grafičkog korisničkog interfejsa (GUI), koji je

prezentiran pomoću ikonica, menija, prozora, dugmića i sličnih objekata, koje je moguće selektovati.

Elektronski miš jeste najpopularniji pokazni uređaj koji se koristi za pomjeranje pokazivača po ekranu kako bi se zadala neka komanda ili selektovala ikonica ili opcija menija. Pokretanjem miša po stolu ili podlozi može se pokazivačem naciľjati neka ikonica, dok se pritiskom na taster aktivira neka operacija nad tom ikonicom. Umjesto miša koriste se i **trackball** (kugla za praćenje) ili **touchpad**. Trackball predstavlja stacionarni uređaj u vidu kugle koja se okreće radi pomjeranja kursora na ekranu (slika 1.13. lijevo). Touchpad je mala pravougaona površina osjetljiva na dodir, a obično je smještena na dnu tastature. Kursor se kreće u pravcu kretanja prsta po toj podlozi (slika 1.13. desno). Touchpad i trackball obično se koriste kod laptop računara.



Slika 1.13. Trackball i touchpad

Dodirni ekran (touch screen) (slika 1.14. lijevo) jeste uređaj koji omogućava upotrebu kompjutera dodirivanjem površine ekrana. Neki touch screen emituju IC zrake, zvučne talase ili slabe električne struje, koji bivaju prekinuti u trenutku i na mjestu dodira. Na taj način kompjuter osjeća tačku dodira i prepoznaje i odgovara odgovarajućom akcijom. Na primjer, izbor opcija u meniju obavlja se jednostavno dodirom traćene opcije.

Uređaji u vidu olovke najviše se koriste kod ručnih računara kao što su PDA. Ovi mali računari imaju brze procesore i softver koji prepoznaju i digitalizuju rukopis, kucanje ili crtanje rukom, tj. uređajem u vidu olovke. Oni imaju podlogu osjetljivu na dodir ispod svojih LCD ekrana. Kod ovih

uređaja umjesto tastature koristi se olovka za izbor opcija, slanje e-mailova ili unos rukom ispisanih podataka direktno u uređaj (slika 1.14. desno).



Slika 1.14. Dodirni ekran i PDA računar

Olovka digitajzer jeste još jedan ulazni uređaj u vidu olovke (Slika 1.15. lijevo). Ona se može koristiti kao pokazni uređaj, ali i za crtanje i ispisivanje po površini grafičke bilježnice, koja je osjetljiva na dodir (slika 1.15. desno). Ovaj uređaj digitalizuje rukopis i crteže, prihvata ih kao ulaz, prikazuje na ekranu i unosi u aplikaciju.



Slika 1.15. Olovka digitajzer i grafička bilježnica

Uređaji za prepoznavanje govora. – Prepoznavanje govora obećava najlakši metod unosa podataka, obrade riječi i kompjuterske konverzacije, jer je govor najprirodnije sredstvo ljudske komunikacije. Govorni input danas je tehnički i ekonomski dostupan za mnoge aplikacije. Rano prepoznavanje govora, u kome su se morale praviti pauze između riječi, bilo je diskretnog tipa. Novi softveri za neprekidno prepoznavanje govora (CSR) prepoznaju normalan konverzacioni govor bez pauza. Ovi sistemi

prepoznaju, digitalizuju, analiziraju i klasifikuju govor i kreiraju njegov zvučni uzorak (kod) – slika 1.16. Softver zatim komparira taj zvučni uzorak sa onim koje ima u svojoj bazi zvučnih uzoraka, prepoznaje riječ i prosljeđuje je aplikativnom softveru.

Danas su brojni primjeri upotrebe sistema baziranih na prepoznavanju govora. Na primjer, proizvođači koriste ove sisteme za provjeru zaliha, a avio kompanije za glasom dirigovano sortiranje prtljaga i drugo. Takođe se mogu koristiti kod operativnih sistema za glasovno zadavanje komandi. Na primjer, neki softveri omogućavaju glasom vođeno slanje e-mailova i surfovanje internetom. Postoje i računari – govorni automati, koji omogućavaju korisnicima da verbalno obave neku svoju aktivnost, kao što je plaćanje računa, provjera stanja akcija, upis u školu ili na univerzitet i drugo.



Slika 1.16. Uređaj za digitalizovanje govora

Optičko skeniranje. – Osnovni zadatak optičkog skeniranja jeste premještanje gomile papira sa radnog stola u računar. Ova ulazna tehnologija omogućava transformisanje sadržaja sa papira, koji može biti bilo kojeg formata (poslovni dokument, pismo, logotip, fotografija, grafikon i slično), u digitalnu formu i njihovo memorisanje na računaru.

Osnovni uređaj za optičko skeniranje jeste skener (slika 1.17. lijevo). **Skener** učitava tekst ili grafiku sa papira i konvertuje ih u digitalni ulazni signal za računar. Skeneri rade na principu fotoelektričnih uređaja koji znakove reflektuju u svjetlosne uzorke. Svjetlosni uzorci se dalje pretvaraju (kodiraju) u elektronske impulse koji predstavljaju input u računarski sistem. Javljaju se u vidu kompaktnih stonih skenera, kao i u vidu skenera sa

ravnim ulaganjem slike, koji su brži i omogućavaju skeniranja u boji većih rezolucija.

Tehnologija skeniranja prvenstveno se primjenjuje kod arhiviranja dokumenata. Skenirana dokumenta mogu se razvrstavati u foldere i jednostavnije i brže pretraživati. Skeniranje se koristi i u desktop i veb-izdavaštvu za skeniranje tekstova ili ilustracija, koji se koriste za kompoziciju strane putem DTP (Desktop Publishing) softvera ili softvera za veb-publikovanje.

OCR (Optical Character Recognition) čitači, tj. uređaji za optičko prepoznavanje znakova, učitavaju OCR kodove na etiketama proizvoda, kreditnim karticama, premijama osiguranja, avionskim kartama i slično. Najčešće se javljaju u vidu ručnih štapova ili olovke (slika 1.17. desno) i služe za učitavanje bar-kodova (trakastih, tj. prugastih kodova), koji se koriste za univerzalno kodiranje proizvoda (UPC standard).



Slika 1.17. Skener, OCR štap i bar-kod pen

Prugasti kod pretvara se u odgovarajuće elektronske impulse (binarne kodove) koji se proslijeđuju kao input računarskom sistemu. Na osnovu

binarnog koda pronalazi se u bazi podataka informacija o cijeni proizvoda, prikazuje se na terminalu i štampa na kupčevom računu.

Ostale input tehnologije. – Očitavanje podataka sa **magnetne trake na kreditnim karticama** (slika 1.18. lijevo) jeste input tehnologija koja se koristi na ATM-ovima banaka i terminalima za autorizaciju kreditnih kartica. Magnetna traka predstavlja u stvari sloj od oksida gvožđa na koji se podaci unose namagnetisavanjem. Ona može sadržati oko 200 bajtova informacija o broju računa kupca. **Smart (inteligentne) kartice**, osim magnetne trake, sadrže i mikroprocesorski čip sa memorijom od nekoliko kilobajta (slika 1.18. desno). Ova kartica uz podatke o broju računa memoriše i saldo, a mikročip omogućava ažuriranje salda. Smart kartice se koriste na mjestima gdje ne postoje terminali za autorizaciju (kao što su parkinzi, telefonske govornice i slično). Saldo se uvijek može dopuniti na bankovnim automatima.

Digitalne kamere takođe predstavljaju jednu od input tehnologija. Naime, one omogućavaju unos multimedijalnih sadržaja (fotografija i video snimaka) u računar. Digitalne slike na računaru se mogu preuređivati i koristiti za desktop i veb-izdavaštvo.

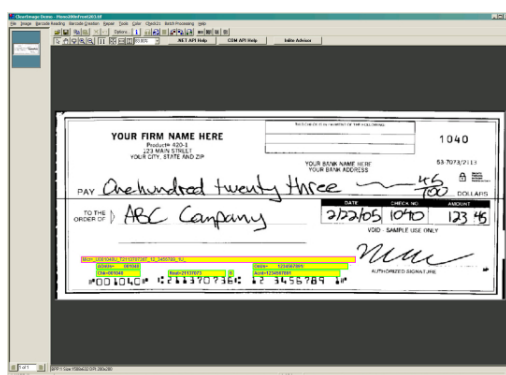
Bankarski sistemi koriste **MICR (Magnetic Inc Character Recognition) input tehnologiju**. Naime, identifikacioni brojevi klijentovih i bankovnih računa, kao i iznos čeka preštampavaju se na dnu čekova mastilom (tj. oksidom gvožđa) koje ima sposobnost namagnetisavanja (slika 1.19. lijevo). Čitači-sorteri čekova prvo magnetišu znakove ispisane magnetnim mastilom, a zatim očitavaju signale indukovane od strane svakog znaka koji prolazi ispod glave za čitanje (slika 1.19. desno). Na ovaj način čekovi se automatski učitavaju i razvrstavaju u bankarskom sistemu.

Izlazne tehnologije i trendovi. – Izlazne tehnologije su se razvijale počev od bušenih kartica, tj. papirnih traka, preko štampanih dokumenata, pa sve do multimedijalnih izlaza. Osnovni trendovi jesu brži, prirodniji i jasniji metodi prezentiranja podataka.

Video prikazi na kompjuterskom ekranu i štampana dokumenta još uvijek su najviše korišćene forme izlaza računarskog sistema. Međutim, danas smo svjedoci sve veće primjene i drugih oblika outputa, kao što su govorne poruke i multimedijalni sadržaji na internet i intranet veb-sajtovima.



Slika 1.18. Magnetna traka i smart kartica



Slika 1.19. Ček ispisan magnetnim mastilom i MICR čitač čekova

Video output. – Video monitor predstavlja komponentu računara kojom se ostvaruje najintenzivnija interakcija, a kvalitetan monitor jeste preduslov za dobro funkcionisanje računara, posebno ako se primjenjuje za rad sa multimedijalnim sadržajima.

Kvalitet slike na ekranu zavisi od grafičke kartice i tipa monitora. Grafičke kartice koje omogućavaju visoku rezoluciju (broj tačkica-piksela, kojima se predstavlja slika na monitoru) daju najkvalitetniju sliku.

Stariji video monitori bazirani su na katodnim cijevima, tzv. CRT ekrani (slika 1.20. lijevo). Danas se sve više koriste monitori sa ravnim ekranima (slika 1.20. desno), koji rade na bazi tečnih kristala – tzv. LCD ekrani. Oni iziskuju manje električne struje i emituju manje električnog zračenja.

Napredne tehnologije, kao što su aktivna matrica, plasma i LED, omogućavaju danas veoma jasnu sliku na ravnim ekranima.

Štampani output. – Većina poslovnih informacionih sistema automatski generiše poslovna dokumenta, kao što su fakture, otpremnice, prijernice, računi i slično, čija je štampanje neophodno. Štampani izlaz još uvijek je najviše korišćena forma izlaza poslije video prikaza.



Slika 1.20. CRT monitori i monitori sa ravnim ekranom

Nekada su se koristili matrični štampači (slika 1.21. lijevo), koji su dosta bučni i spori (štampaju nekoliko stranica u minuti). Radili su na principu iglica koje preko mastiljave trake (ribbon) ostavljaju tragove na papiru. Neki ih još uvijek koriste, jer su dugotrajni i jeftini za nabavku i održavanje.

Danas se koriste laserski štampači (slika 1.21. desno). Oni koriste elektrostatički postupak, a on podrazumijeva lasersko osvjetljavanje materijala koji tako postaje naelektrisan te privlači toner koji se otiskuje na papir. Ovi su štampači veoma brzi, štampaju od nekoliko desetina do nekoliko stotina stranica u minuti, rade gotovo nečujno i imaju visok kvalitet štampe. Imaju i mogućnost štampe u boji.

U kancelarijama se najviše koriste višenamjenski štampači koji imaju mogućnost štampe, faksa, skeniranja i kopiranja.

Memorijske tehnologije – trendovi. – Primarna memorija računara u početku se javljala u vidu magnetnih kalema ili magnetnog jezgra. Počev od

treće generacije, koriste se poluprovodničke memorije. Sa pojavom tehnologije visoke integracije, sa pojavom tj. memorijskih čipova, kapacitet primarne memorije je u stalnom porastu. Sekundarne memorije u početku su bile isključivo magnetne, a javljale su se u vidu magnetnih traka i kalema, pa sve do magnetnih diskova, koji se i danas koriste. Danas se sve više koriste i optičke memorije, čija se prednost ogleda u većoj gustini i kvalitetu memorisanja multimedijalnih podataka.

Poluprovodničke memorije imaju manji kapacitet, ali veću brzinu pristupa podacima od magnetnih i optičkih. Memorije sa bržim pristupom imaju veću cijenu i zato su manjeg kapaciteta. Poluprovodničke memorije nisu stalnog karaktera, tj. gube sačuvane podatke kada izgube napajanje.

Sekundarni medijumi (magnetni i optički) trajnog su karaktera i koriste se za trajno memorisanje podataka. Imaju sporiji pristup podacima u odnosu na poluprovodničke, i jeftinije su, pa, u zavisnosti od potrebe, mogu biti i velikog kapaciteta.

Osnovni trendovi za obje vrste memorija jesu veći kapacitet i brzina pristupa podacima, a manja cijena.



Slika 1.21. Matrični i laserski štampači

Osnove kompjuterskog skladištenja podataka. – Podaci se u kompjuterskom sistemu memorišu i obrađuju u binarnom brojnem sistemu (sistem koji se sastoji od dvije cifre 0 i 1). Ovakav način prezentiranja podataka potiče od svojstava memorijskih medijuma koji mogu da se nađu u

dva stanja. Kod poluprovodničkih memorija riječ je o provodnom i neprovodnom stanju, dok se kod magnetnih memorija ova dva stanja ogledaju u mogućnosti polarizovanja magnetnog polja na pozitivno i negativno.

Dakle, binarne karakteristike memorijskih medijuma jesu osnovni razlog što je binarni sistem osnov za prezentovanje podataka u računar. Stoga provodno stanje predstavlja binarnu jedinicu za elektronska kola, dok neprovodno stanje predstavlja binarnu nulu. Za magnetne medijume, magnetno polje namagnetisane tačke u jednom pravcu predstavlja binarnu jedinicu, a u drugom pravcu – binarnu nulu.

Najmanji dio podatka naziva se **bit** i predstavlja jednu jedinicu ili nulu. Kapacitet memorije se izražava bitima. Osnovna jedinica podataka sa kojom kompjuter radi jeste **bajt (B)** i predstavlja binarni kod jednog karaktera (karakter je slovo, cifra ili specijalni znak). Postoji više standarda za kodiranje karaktera. Opšteprihvaćeni standard jeste ASCII (američki standardni kod za unutrašnju razmjenu informacija), kod koga se svaki karakter kodira sa 8 bita. U tabeli 1.2. predstavljeni su ASCII kodovi nekih karaktera. Većina računara koristi ASCII kodove, pa se može zaključiti da jedan bajt ima 8 bita. S obzirom na to da su danas memorijski medijumi velikih kapaciteta za mjerenje kapaciteta koriste se i veće jedinice, kao što su kilobajt (**KB**), megabajt (**MB**), gigabajt (**GB**) i terabajt (**TB**). Iako kilo u metričkom sistemu znači hiljadu, kompjuterska industrija koristi K da predstavi 1024, odnosno 2^{10} karaktera. Tako na primjer 10 MB predstavlja stvarnih 10485760, a ne 10 miliona karaktera. Ove razlike se često ignorišu zbog jednostavnijeg opisa memorijskih kapaciteta. Tako je megabajt ima oko milion bajta memorije, gigabajt – oko bilion, a terabajt – oko trilion bajta.

Direktni i sekvencijalni pristup. – Primarna memorija poznata je kao memorija sa direktnim pristupom ili kao memorija sa slučajnim pristupom (RAM – Random Acces Memories). Magnetni diskovi poznati su pod nazivom DASD (Direct Acces Storage Devices) i oni predstavljaju memorijski uređaj sa direktnim pristupom. Magnetna traka jeste primjer memorije sa sekvencijalnim pristupom.

Tabela 1.2. Primjeri ASCII kodova

Karakter	ASCII kod
0	00110000
1	00110001
2	00110010
3	00110011
...	...
A	01000001
B	01000010
C	01000011
...	...

Direktni ili slučajni pristup memoriji predstavljaju isti koncept koji podrazumijeva da se jedan karakter može upisati i pročitati direktno na bilo kojoj memorijskoj lokaciji. Ovo podrazumijeva da svaka memorijska lokacija ima jedinstvenu adresu i da se svakoj lokaciji pristupa za isto vrijeme bez prethodnog pristupanja nekim drugim lokacijama.

Sekvencijalni pristup podrazumijeva da su podaci memorisani u sekvenci. Da bi se pristupilo nekom podatku, mora se pristupiti prethodno svim podacima koji su u sekvenci prije njega. Ovakav pristup podacima znatno je sporiji od direktnog.

Poluprovodnička memorija. – Primarna memorija računara sastoji se od poluprovodničkih memorijskih čipova. To je radna memorija u kojoj kompjuter smješta podatke i instrukcije u toku obrade.

Glavne prednosti poluprovodničke memorije jesu mala veličina, velika brzina i otpornost na šokove i temperaturne promjene. Glavni nedostatak poluprovodničke memorije jeste nepostojanost, tj. ukoliko nema neprekidnog strujnog napajanja, sadržaj iz nje se briše. Ovaj problem se prevazilazi upotrebom baterijskog napajanja ili generatora struje. Alternativno rješenje jeste stalno presnimavanje sadržaja iz operativne memorije na neki trajni memorijski medijum i, u slučaju nestanka struje, ponovno vraćanje u radnu memoriju.

Operativna memorija sastavljena je od dva osnovna tipa memorije, tj. dijelom je RAM, a dijelom ROM (Read Only Memory).

RAM – Random Access Memory: RAM memorijski čipovi čine većinu operativne memorije. Svakoj memorijskoj lokaciji može se pristupiti (čitati) i promijeniti (pisati), pa se ove memorije nazivaju i čitaj-piši memorije. Ova vrsta memorije je nepostojana.

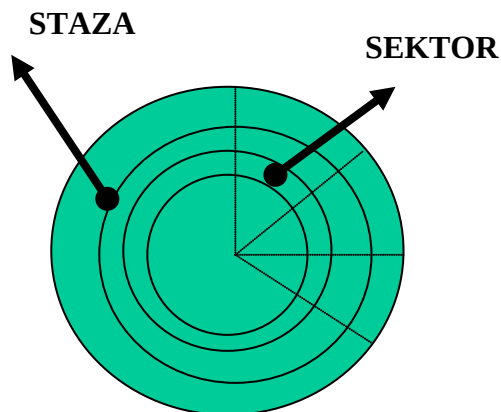
ROM – Read Only Memory: Postojani memorijski čipovi za trajno memorisanje takođe su sastavni dio operativne memorije. ROM jeste memorija samo za čitanje, ne može se brisati niti presnimavati. Obično su u njoj smještene najčešće korišćene kontrolne instrukcije kod kontrolne jedinice, kao i inicijalni dio operativnog sistema (BIOS) kod operativne memorije. Ovi sadržaji nasnimljeni su u toku proizvodnje računara i ne mogu se mijenjati.

Javljaju se i u vidu **PROM (Programmable Read Only Memory) memorija**, u koje je moguće snimiti program (programirati ih) trajno, kao i **EPROM (Erasable Programmable Read Only Memory) memorija**, koje se mogu privremeno programirati, tj. izbrisive su. Ove vrste memorija namijenjene su za direktno programiranje mašinskim instrukcijama u slučaju potrebe što bržeg i efikasnijeg izvršavanja instrukcija.

Magnetne memorije: Magnetne memorije se javljaju u vidu magnetnih diskova i magnetnih traka.

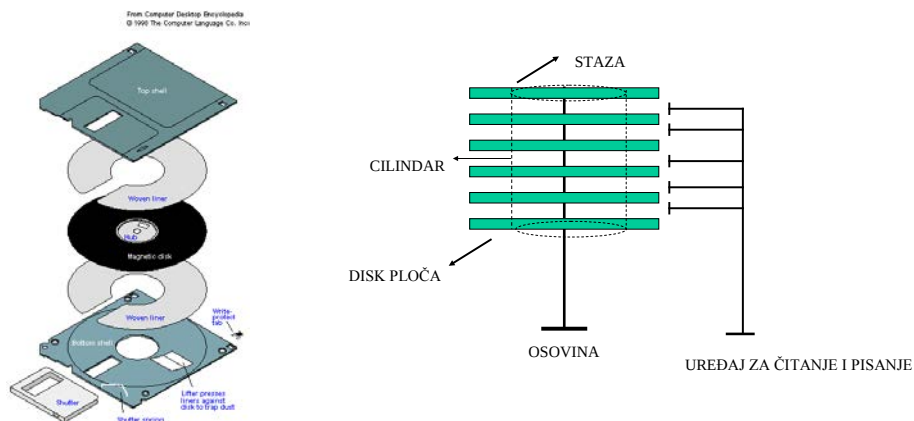
Magnetni diskovi danas imaju kapacitet i do nekoliko desetina gigabajta, što nije čudno ako se uzme u obzir činjenica da multimedijalni sadržaji zauzimaju ogromni memorijski prostor. Magnetni diskovi su uobičajena forma sekundarne memorije za računar.

Magnetni disk jeste metalni disk presvučen oksidom gvožđa, koji ima sposobnost namagnetisavanja. Podaci se zapisuju na koncentričnim stazama (tankim namagnetisanim tragovima) u vidu binarnih cifara, tj. ASCII kodova. Na jednoj traci može biti zapisano nekoliko hiljada bajtova, a na licu diska ima više stotina staza. Disk se rotira oko vertikalne osovine brzinom od 3600 do 7600 rpm-obrtaja u minuti. Podaci se upisuju i čitaju pomoću elektromagnetne R/W glave, kojom upravlja pristupni uređaj. Podacima se pristupa direktno, na osnovu jedinstvene adrese. Adresa se formira pomoću staza i sektora (slika 1.22.).



Slika 1.22. Adresiranje magnetnog diska

Magnetni diskovi mogu biti prenosivi i fiksirani. Osnovni prenosivi disk jeste **floppy disk**. Floppy disk je napravljen od poliestera koji je presvučen oksidom gvožđa. Savitljiv je, pa otuda potiče njegov naziv floppy – meki. Disk se rotira unutar zaštitnog omotača od plastike, koji je kvadratnog oblika. Veličina diska je 3,5 inča a kapacitet 1,44 MB (slika 1.23. lijevo).

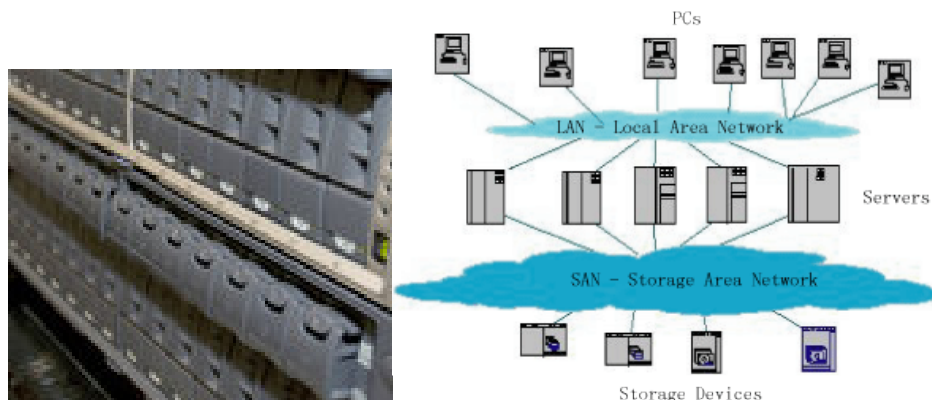


Slika 1.23. Floppy disk – disketa i adresiranje hard diska

Hard disk je fiksiran u kućištu računara. Sastoji se od više magnetnih diskova na vertikalnoj osovini, a između njih se nalaze R/W glave. Kapacitet im se kreće od nekoliko stotina MB do nekoliko desetina GB. Za adresiranje podataka, osim staza i sektora, koriste se i vertikale, tzv. cilindri (slika 1.23. desno).

RAID (Redundant Arrays of Independent Disks) memorija predstavlja niz više međusobno povezanih hard diskova (6 do 100) (slika 1.24. lijevo). Zajedno sa mikroprocesorom, koji njima upravlja, obezbjeđuju ogromne memorijske kapacitete (mnogo gigabajta) i veliku brzinu pristupa podacima (paralelni pristup diskovima). Koristi se i kod sigurnosnih sistema za pravljenje kopija podataka (sigurnosna redundanca).

Povezivanjem više dislociranih RAID memorija optičkim kablom nastaju **SAN (Storage Area Network) memorije** koje predstavljaju ogromne distribuirane memorijske kapacitete kojim upravljaju mrežni serveri (slika 1.24. desno).



Slika 1.24. RAID i SAN memorije

Magnetna traka. – Ova vrsta magnetne memorije ima sekvencijalni pristup i povoljna je za batch (paketno) memorisanje. Ima dosta nižu cijenu od magnetnog diska. Javlja se u vidu kalema ili kertridža (kaseta) (slika 1.25.). Grupisanjem nekoliko stotina kasete kojim upravlja robot tehnologija postiže se velika brzina pristupa podacima i veliki kapacitet. Danas se

koristi uglavnom za arhiviranje i backup (rezervne kopije podataka), isključivo zbog niže cijene od ostalih memorijskih medijuma.



Slika 1.25. Magnetna traka u vidu kalema i kasete

Optičke memorije se javljaju u vidu optičkih diskova.

Optički disk jeste metalni disk na kome se podaci snimaju laserski tako što se prave mikroskopska udubljenja na spiralnim trakama na površini diska. Laserski uređaj čita binarne kodove podataka prezentovane tim udubljenjima, tj. odgovarajućom refleksijom svjetlosti koju emituju ta udubljenja. Ovi diskovi memorišu podatke gustinom mnogo većom nego magnetni diskovi, ali imaju manju brzinu upisivanja podataka, i skuplji su. Koriste se najviše za image i multimedijalne podatke (katalozi, prezentacije, enciklopedije itd.). Javljaju se u sljedećim vidovima:

CD-ROM (Compact Disk-Read Only Memory). – To su diskovi prečnika 12 cm, a kapacitet im je više od 600 MB (kao 400 floppy diskova ili 300000 stranica teksta). Kupuju se nasnimljeni i ne mogu se brisati niti presnimavati. Na njima su najčešće nasnimljeni muzički albumi, enciklopedije, katalozi, adresari, priručnici, programske aplikacije i slično.

CD-R (Compact Disk – Recordable). – Ovi optički diskovi omogućavaju snimanje podataka (narezivanje), ali samo jednom, dok se čitati mogu više puta – WORM (write once, read many) tehnologija. Najčešće se koriste za snimanje digitalne muzike ili fotografija (slika 1.26. lijevo).

CD-RW (CD – Rewritable). – Ova vrsta diskova omogućava snimanje i brisanje podataka. Sadržaj se briše laserskim topljenjem udubljenja na površini diska, a magnetno optičkim tehnikama moguće je izmijeniti reflektivne osobine udubljenja iz jednog pravca u drugi, mijenjajući tako 1 u 0 i obrnuto, tj. mijenjajući podatke (zato su se ovi diskovi ranije nalazili i sa nazivom CD-MO) – slika 1.26. desno.



Slika 1.26. CD-R i CD-RW optički diskovi

Kapacitet i mogućnosti optičkih diskova znatno su povećani kod **DVD (Digital Video Disk ili Digital Versatile Disk)** diskova. Kapacitet im se kreće od 4,7 GB (na jednoj strani) do 8,5 GB (na obje strane). Imaju veoma visok kvalitet memorisanja image, audio i video podataka. Javljaju se u vidu DVD-ROM i DVD-RAM tehnologije. **DVD-ROM** diskovi se kupuju sa nasnimljenim multimedijalnim sadržajima. U filmskoj reprodukciji danas su potpuno zamijenili magnetne video kasete kao i CD-ove. **DVD-RAM** diskovi (slika 1.27. lijevo) koriste se za arhiviranje i backup masivnih i multimedijalnih podataka. Mogu biti **DVD-R** i **DVD-RW**, sa istim značenjem kao kod CD diskova.

U novije vrijeme koriste se i **Blu-ray** optički diskovi, koji imaju veliki kapacitet od 25 GB (jednostrani) do 50 GB (dvostrani). Naziv potiče od upotrebe plavog lasera koji omogućava pet puta gušće smještanje podataka od običnog DVD diska (slika 1.27. desno).



Slika 1.27. DVD i Blue-ray optički diskovi

Optičke memorije imaju veći kapacitet od magnetnih, ali su znatno sporije i skuplje od magnetnih, pa se ne očekuje da će u potpunosti zamijeniti standardne magnetne diskove u poslovnim primjenama.

USB Flash memorije. – Ovo je vrsta trajne spoljašnje memorije koja se može elektronski brisati i reprogramirati. Spada u EEPROM (Electrically Erasable Programmable Read-Only Memory). Za razliku od EPROM memorije, koja se javlja kao sastavni dio primarne memorije, gdje se brisanje i programiranje obavlja na nivou bajta, kod ove memorije se to radi na nivou velikih blokova, zbog čega su one mnogo brže. Ove memorije su veoma popularne kod portable uređaja koji nemaju floppy, CD i DVD čitače. Kapacitet im se kreće od 512 MB do 8 GB. Imaju neka ograničenja u pogledu sigurnosti podataka. Naime, ne mogu se neograničeno presnimavati (oko 100 000 puta), a upravljanje blokovima može dovesti do nemogućnosti korišćenja punog kapaciteta memorije. Osim toga, cijena po GB je veća nego kod magnetnih memorija, pa se ne očekuje da će zamijeniti masivne magnetne memorije (slika 1.28.).



Slika 1.28. Flash memorije

1.4. RAČUNARSKE MREŽE

Računarske mreže omogućavaju razmjenu informacija različitih formata (glas, tekst, grafika, zvuk i video) posredstvom međusobno povezanih računara.

Računarske mreže, u koje spadaju globalna svjetska mreža internet, mreža unutar jedne organizacije – intranet i mreža organizacije sa poslovnim partnerima – ekstranet, predstavljaju osnovu za elektronsko komuniciranje i saradnju radnih timova, organizacija i poslovnih partnera, za efikasno upravljanje poslovanjem, dijeljenje poslovnih resursa i za konkurentno učešće na globalnom svjetskom tržištu.

1.4.1. TRENDovi U TELEKOMUNIKACIJAMA I RAČUNARSKIM MREŽAMA

Telekomunikaciona industrija transformisala se od monopola regulisanog vladom u slobodno tržište telekomunikacionih usluga. Tako se danas veliki broj kompanija nudi mnoštvo različitih telekomunikacionih usluga. Lokalne i globalne telefonske usluge (u vidu celularnih ili PCS sistema), kablovska televizija, video konferencije, interaktivna televizija, video na zahtjev, učenje na daljinu, desktop multimedija, pristup internetu i slično – samo su neke od njih.

Osnovni tehnološki trend računarskih mreža jeste konvergencija klasičnih klijent/server sistema ka otvorenim sistemima. Otvoreni sistemi

podrazumijevaju neograničenu mogućnost povezivanja, pri čemu koriste opšte standarde tj. internet mrežne tehnologije, kao što su web browseri, TCP/IP internet mrežni protokoli, mrežne sigurnosne zaštite (fire walls) i drugo. Otvoreni sistemi, kao što su internet i organizacijski intranetovi i ektranetovi, predstavljaju kompjutersko okruženje koje omogućava jednostavne pristupe krajnjih korisnika i njihovih računarskih mreža. Ovi sistemi omogućavaju vršenje više različitih aplikacija koje pokreću razni tipovi računarskih sistema, zasnovani na različitim bazama podataka i oslonjeni na različite međusobno povezane mreže. Drugi revolucionarni tehnološki trend jeste prelazak na digitalne mrežne tehnologije. Raniji telekomunikacioni sistemi zasnivali su se na analognom prenosu, tj. prenosu električnih frekvencija generisanih zvučnim talasima ljudskog glasa. Digitalne tehnologije prenose informacije u obliku diskretnih kompjuterskih signala, što omogućava znatno veću brzinu prenosa, prenos većeg broja informacija, veću ekonomičnost i mnogo manje grešaka u prenosu. Još jedan tehnološki trend jeste prelazak sa bakarnih medija na fiber-optičke, kao i prelazak sa zemaljskih mikrotalasnih sistema na satelitske i druge bežične tehnologije. Osnovni trend predstavlja i povezivanje celularnih i PCS mobilnih telefona, kao i PDA-ova na internet i na organizacione mreže.

Promjene u telekomunikacionoj industriji i tehnologiji rapidno su povećale broj poslovnih primjena. Računarske mreže igraju glavnu ulogu u oblasti elektronske trgovine, u interorganizacionoj saradnji i u drugim poslovnim primjenama koje podržavaju operativnost i upravljanje preduzećem.

1.4.2. PREDNOSTI PRIMJENE RAČUNARSKIH MREŽA U POSLOVANJU

Informacione tehnologije, posebno u poslovnim primjenama baziranim na telekomunikacijama, omogućavaju prevazilaženje geografskih, vremenskih, troškovnih i strukturalnih barijera na putu ka poslovnom uspjehu.

- Prevazilaženje geografskih barijera podrazumijeva mogućnost saradnje sa poslovnim partnerima i vršenje transakcija na udaljenim lokacijama. Narudžba nekog kupca, putem interneta i ektraneta, prosljeđuje se sa udaljenog prodajnog mjesta kako bi se u organizacionom centru ova narudžba obradila. Na ovaj način smanjuje se vrijeme čekanja kupca za ispunjenje narudžbe i ubrzava se naplata narudžbe.

- Prevazilaženje vremenskih barijera podrazumijeva mogućnost trenutnog obezbjeđivanja informacija na udaljenim lokacijama. Provjera kreditne kartice na online POS prodajnom mjestu obavlja se u sekundama.
- Prevazilaženje troškovnih barijera podrazumijeva smanjenje troškova tradicionalne komunikacije (npr. poštarine, tel. računa, poslovnih putovanja). Desktop video konferencija između organizacije i njenih poslovnih partnera putem interneta, intranetova i ekstranetova smanjuje visoke troškove poslovnih putovanja i povećati efikasnost i kvalitet donijetih odluka.
- Prevazilaženje strukturalnih barijera podrazumijeva sinhronizaciju sa poslovnim partnerima u smislu poštovanja standarda. Korišćenjem EDI (Electronic Data Interchange) razmjene podataka između poslovnih partnera, komunikacija se standardizuje bez obzira na njihovu organizacionu, hardversku, softversku ili mrežnu strukturu.

1.4.3. INTERNET

Eksplozivni rast interneta predstavlja revolucionarni fenomen u oblasti računarstva i telekomunikacija. Internet je danas najveća i najvažnija mreža koja se konstantno širi na sve veći broj poslovnih i drugih organizacija, njihovih korisnika, kompjutera i mreža, sjedinjujući ih u jedan globalni Web. Hiljade poslovnih, edukativnih i istraživačkih mreža sada povezuju milione kompjuterskih sistema i korisnika u više od 200 zemalja širom svijeta. Internet postaje ključna platforma za razne poslovne aplikacije, kao što su saradnja između preduzeća, elektronska trgovina i drugo.

Ne postoji saglasnost u vezi sa datiranjem nastanka interneta. Neki smatraju da je to početak šezdesetih, kada je otkrivena packet-switching tehnologija za prenos podataka. Drugi pretečom interneta smatraju mrežu ARPANET, koju je krajem šezdesetih godina kao vojni eksperiment oformilo američko ministarstvo odbrane. Kasnije je ova mreža postala javni istraživački eksperiment i obuhvatala je nekoliko stotina računara. Ipak, može se reći da je internet doživio eksponencijalni rast zahvaljujući pojavi TCP/IP tehnologije početkom osamdesetih godina.

Ova mreža nema centralni kompjuterski sistem ili telekomunikacioni centar. Svaka poruka ima svoju jedinstvenu adresu i bilo koji internet server može je proslijediti do njene destinacije. Internet nema upravljačko tijelo, već samo internacionalna udruženja za propisivanje opštih standarda i njihovo promovisanje. Ovi opšti standardi su ključni za slobodan protok informacija između međusobno različitih kompjuterskih i mrežnih sistema organizacija i internet provajdera.

Najpopularnije primjene Interneta jesu elektronska pošta (*e-mail*), pretraživanje sajtova na *webu* pomoću browsera i participacija u diskusionim grupama (*newsgroups*) i čatovanje (*chat rooms*). Internet e-mail poruke u vidu podataka, teksta, faksa, grafike, video fajlova i drugog, stižu za nekoliko sekundi ili minuta na bilo koju lokaciju u svijetu. Internet browser softver kao što je Internet Explorer omogućavaju milionima korisnika surfovanje *webom* klikom na multimedijalne informacione resurse (*websites*) smještene na hiperlinkovima povezanim stranama u oblasti poslovanja, istraživanja itd. Internet obezbjeđuje elektronske diskusione forume kojima upravljaju različite interesne grupe. Bilo koji korisnik na internetu može učestvovati u diskusiji tako što čitaj i odgovara na poruke. Još jedna popularna primjena jeste down-loading fajlova i softvera, pri čemu se oni automatski kopiraju na računar korisnika, kao i pristup nekim specijalnim bazama podataka. Automatsko pretraživanje informacija na veb-sajtovima obavljaju pretraživači kao što su Yahoo!, Google i Fast Search. Čatovanje podrazumijeva konverzaciju u realnom vremenu (on-line) sa drugim internet korisnicima.

Poslovne primjene interneta proširene su sa elektronske razmjene podataka na široku platformu za strateške poslovne primjene, kao što su saradnja između poslovnih partnera, saradnja sa korisnicima, elektronska trgovina, interaktivni marketing, prodaja i drugo.

Koristeći internet, organizacije publikuju veb-sajtove za interaktivni marketing i elektronsku trgovinu i ostvaruju komunikaciju i razmjenu dokumenata sa kupcima i poslovnim partnerima. Ekstranetovi elektronske trgovine, koji su realizovani preko osiguranih internet linkova, omogućavaju dobavljačima da provjeravaju i dopunjuju zalihe i inventar i da šalju poslovna dokumenta. Kupci koriste veb-sajt elektronske trgovine za interaktivnu kupovinu proizvoda i usluga. Poslovni partneri koriste e-mail, diskusione forume, video konferencije za saradnju, ali mogu imati i ekstranet pristup resursima intraneta. Udaljeni odjeli organizacije, kao i

članovi virtuelnih timova koriste intranet linkove preko interneta za interaktivnu komunikaciju, saradnju kao i za pristup podacima i programima.

Primjena interneta i internet baziranih tehnologija kod intranetova i ekstranetova, prije svega značajno smanjuje troškove jer su razvoj, rad i upravljanje njima znatno jeftiniji od tradicionalnih sistema. Na primjer, za avio kompaniju mnogo je jeftinije da korisnici koriste njihov veb-sajt nego telefonski sistem. Druga primarna prednost jeste privlačenje novih korisnika inovativnim marketingom, kao i zadržavanje postojećih korisnika sa naprednim korisničkim servisima. Efikasno stvaranje prihoda putem elektronske trgovine donosi organizaciji nesumnjivo najveće prednosti u poslovnom primjenjivanju interneta, a to su:

- stvaranje novih prihoda putem on-line prodaje;
- redukovanje troškova on-line trgovinom i korisničkom podrškom;
- privlačenje novih korisnika putem veb-marketinga i reklamiranja i on-line prodaje;
- povećanje lojalnosti postojećih korisnika pomoću naprednih veb-korisničkih servisa i podrške;
- razvoj novih veb-tržišta i distribucionih kanala za sopstvene proizvode i
- razvoj novih informacionih proizvoda dostupnih na webu.

1.4.4. POJAM I ULOGA INTRANETA U POSLOVANJU

Intranet jeste mreža unutar jedne organizacije koja koristi internet tehnologije (web browser, TCP/IP mrežni protokol, HTML (Hypertext Markup Language) hipermedijalni format dokumenata i podataka i slično). Ovakva mreža omogućava dijeljenje informacija i hardverskih i softverskih resursa, komunikacije i saradnju, kao i podršku poslovnim procesima u okruženju sličnom internetu. Intranet posjeduje sigurnosne sisteme zaštite kao što su password, enkripcija, fire walls i drugo, pa mu autorizovani korisnici mogu pristupiti putem interneta. Intranetu takođe mogu pristupiti sa svog intraneta kupci, dobavljači i ostali poslovni partneri putem ekstranet linkova.

Intranet aplikacije grupišu se konceptualno u nekoliko kategorija korisničkih servisa i tako formiraju **intranet portale**. Ovi korisnički servisi

realizuju se pomoću browsera, aplikacionog servera i odgovarajućeg softvera. Intranet portal podržava komunikaciju i saradnju, veb-izdavaštvo, poslovne operacije i menadžment, kao i upravljanje samim portalom. Osnovne komponente intranet portala jesu sljedeće:

Komunikacija i saradnja. – Ova komponenta sastoji se od skupa servisa koji omogućavaju efikasnu komunikaciju i saradnju radnih timova u organizaciji. Na primjer, član radnog tima može, koristeći servise ove komponente, da šalje i prima e-maile, glasovne poruke, faksove i da na taj način komunicira sa ostalima u organizaciji, ili van nje preko interneta ili ekstranetova. Saradnja tima i usaglašavanje u vezi sa projektima može se realizovati putem audio i video konferencija, čatovanjem i drugim.

Veb-publikovanje. – U okviru intraneta vrši se i kreiranje i publikovanje hiperlinkovanih multimedijalnih dokumenata poput onih koji su dostupni na *webu*. Ova komponenta intranet portala obezbjeđuje za to potrebne servise. Prednosti kao što su lakoća, atraktivnost i niski troškovi publikovanja i pristupa multimedijalnim poslovnim informacijama putem intranet veb-sajtova, najviše su zaslužne za eksplozivni rast primjene intraneta u poslovanju. Tako na primjer, organizacija može publikovati katalog proizvoda, tehničke crteže i drugo, a zaposleni će, koristeći intranet browser, na efikasan i jednostavan način locirati potrebne poslovne informacije.

Poslovna operativa i menadžment. – Ova komponenta sadrži skup servisa koji omogućavaju razvoj, realizaciju i upotrebu poslovnih aplikacija koje podržavaju poslovne procese i upravljačko odlučivanje (obrada narudžbenica, kontrola zaliha, upravljanje prodajom i drugo). Većina ovih aplikacija je povezana i pristupa organizacionim bazama podataka i standardnom IS, a instalirane su na intranet veb-serverima. Takve aplikacije se mogu implementirati kako na intranetu, tako i na ekstranetovima i internetu. Ovo znači da zaposleni ili poslovni partneri mogu pokrenuti ove aplikacije koristeći web browsere sa bilo kog mjesta u mreži, kad god su im potrebne.

Upravljanje intranet portalom. – Ovu komponentu čini skup servisa koji omogućavaju upravljanje portalom, tj. administraciju klijenata i servera u mreži, realizaciju sigurnosnih aspekata i slično.

1.4.5. POJAM I ULOGA EKSTRANETA U POSLOVANJU

Ekstranetovi predstavljaju mrežne linkove koji koriste internet tehnologije za povezivanje intraneta jedne organizacije sa intranetovima njenih poslovnih partnera. Organizacije mogu uspostaviti direktne privatne mrežne linkove među sobom ili kreirati privatne osigurane internet linkove, tj. virtuelnu privatnu mrežu. Takođe se mogu koristiti i neosigurani internet linkovi, ali uz enkripciju (kodiranje) osjetljivih podataka i uz firewall sisteme kako bi se obezbijedila adekvatna sigurnost.

Dakle, ekstranet omogućava kupcima, dobavljačima, konsultantima, učesnicima u ugovorima, poslovnim planerima i ostalima da pristupaju odabranim intranet veb-sajtovima i organizacionim bazama podataka. Tako na primjer, preko ekstranet linka kupci imaju mogućnost korišćenja online marketinga, naručivanja i kupovine, tj. plaćanja proizvoda. Dobavljači i distributeri mogu, zahvaljujući ekstranetu, efikasno upravljati nabavkom i distribucijom, a poslovni partneri provjeriti ugovore, nadgledati realizaciju zajedničkih projekata i slično.

Web browser tehnologije omogućavaju jednostavan i efikasan način pristupa intranet resursima za kupce, dobavljače i ostale poslovne partnere organizacije. Ovaj, kao i veliki broj novih web baziranih servisa namijenjenih poslovnim partnerima, opravdava tvrdnju da ekstranetovi grade i jačaju strateške odnose sa poslovnim okruženjem. Osim toga, ekstranetovi unapređuju saradnju i komunikaciju sa poslovnim partnerima i omogućavaju online interaktivni razvoj proizvoda i marketing, koji utiču na bržu pojavu bolje dizajniranih proizvoda na tržištu.

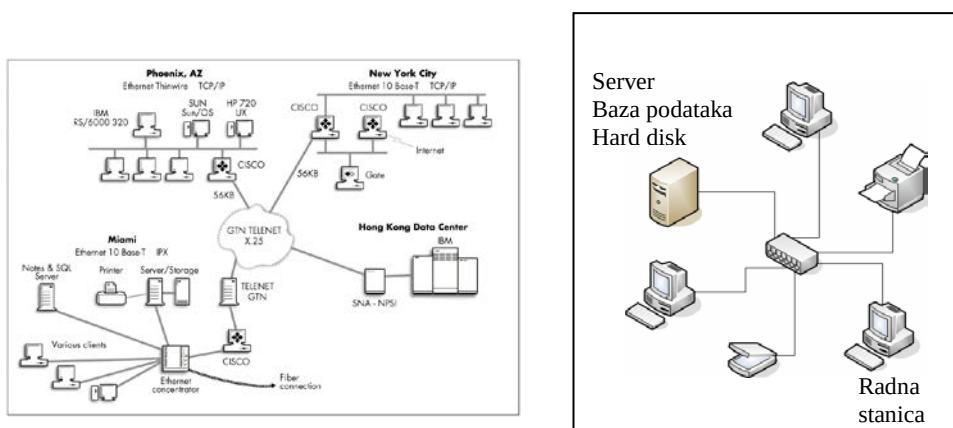
1.4.6. TIPOVI TELEKOMUNIKACIONIH MREŽA

Telekomunikaciona infrastruktura interneta, intranetova i ekstranetova javlja se u vidu mnoštva različitih tipova mreža. Sa aspekta korisnika, možemo ih svrstati u nekoliko osnovnih tipova.

WAN (Wide Area Networks) mreže jesu mreže širokog opsega, tj. mreže koje pokrivaju veliko geografsko područje (slika 1.29. lijevo). U njih spadaju i tzv. gradske mreže (MAN), koje pokrivaju područje jednog grada ili metropole. WAN mreže koriste, na primjer, multinacionalne kompanije za komunikacije sa zaposlenima, kupcima, dobavljačima i drugim

organizacijama, koji su locirani u udaljenim gradovima, regijama i zemljama širom svijeta.

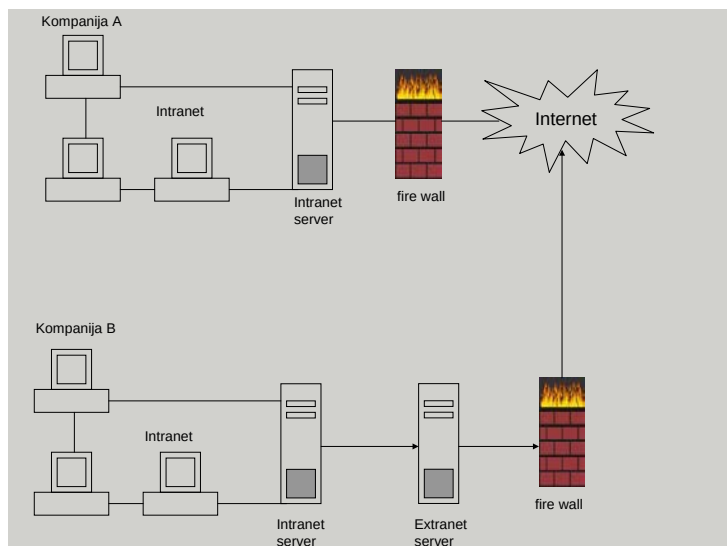
LAN (Local Area Networks) mreže povezuju kompjutere i druge uređaje za procesiranje informacija u okviru ograničenog fizičkog prostora kao što je kancelarija, učionica, zgrada i dr. LAN mreže u organizaciji povezuju korisnike u kancelarijama, organizacionim odjelima i ostalim radnim grupama. LAN predstavlja skup različitih telekomunikacionih medija kao što su standardni telefoni, koaksijalni kablovi, ili čak bežični radio sistem, koji međusobno povezuju mikrokompjuterske radne stanice i kompjuterske periferije. Da bi komunicirao preko mreže, svaki PC mora imati mrežnu karticu. Većina LAN-ova koristi jedan moćniji mikrokompjuter sa velikim hard diskom kao mrežni server. Na njemu se nalazi mrežni operativni sistem – softver, koji kontroliše telekomunikacije i omogućava upotrebu i dijeljenje mrežnih resursa. Na slici 1.29. prikazana je LAN mreža koja omogućava dijeljenje baze podataka, hard diska, štampača i skenera.



Slika 1.29. WAN i LAN mreža

VPN (Virtual Private Network) mreže koriste se da bi se osigurali intranetovi i ektranetovi. Može se reći da su to osigurane mreže koje kao bazičnu mrežu koriste internet. Prilikom konekcije sopstvenog intraneta na internet ili neki drugi intranet, koriste sigurnosne servise poput fire walls (vatrene zida). Primjer upotrebe VPN mreže jeste kada organizacija koristi internet uz sigurnosne mehanizme za ostvarivanje intraneta sa svojim

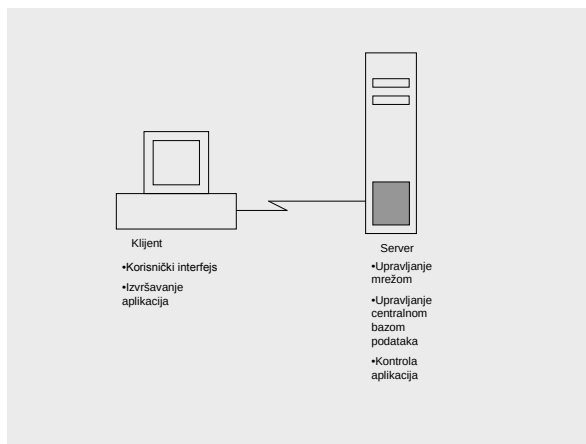
udaljenim kancelarijama, kao i za realizaciju ekstraneta sa poslovnim partnerima (slika 1.30.).



Slika 1.30. VPN mreža

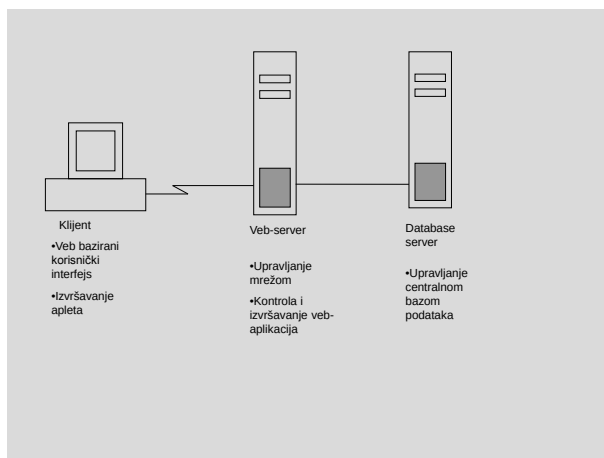
Klijent/server mreže jesu najrasprostranjeniji tip mreže u preduzećima i organizacijama. Klijenti ove mreže jesu korisničke PC (ili neke druge) radne stanice, i oni mogu samo da obezbjeđuju korisnički interfejs ili da dijelom ili u potpunosti izvršavaju korisničke aplikacije. Oni su povezani sa mrežnim serverom koji upravlja mrežom, bazom podataka i izvršava korisničke aplikacije. Ovaj tip mreže poznat je pod nazivom **dvoslojna klijent/server arhitektura** (slika 1.31.).

Osamdesetih godina veliki mainframe bazirani sistemi zamijenjeni su klijent/server mrežama koje se sastoje od više međusobno povezanih LAN-ova. Međutim, ova zamjena podrazumijevala je često i kompleksnu i skupu instalaciju novog aplikativnog softvera umjesto klasičnog mainframe baziranog poslovnog informacionog sistema. Sa druge strane, klijent/server mreže su, sa stanovišta potreba korisnika, radnih grupa i poslovnih jedinica, mnogo ekonomičnije i fleksibilnije od mainframe mreža.



Slika 1.31. Klasična dvoslojna Klijent/Server mreža

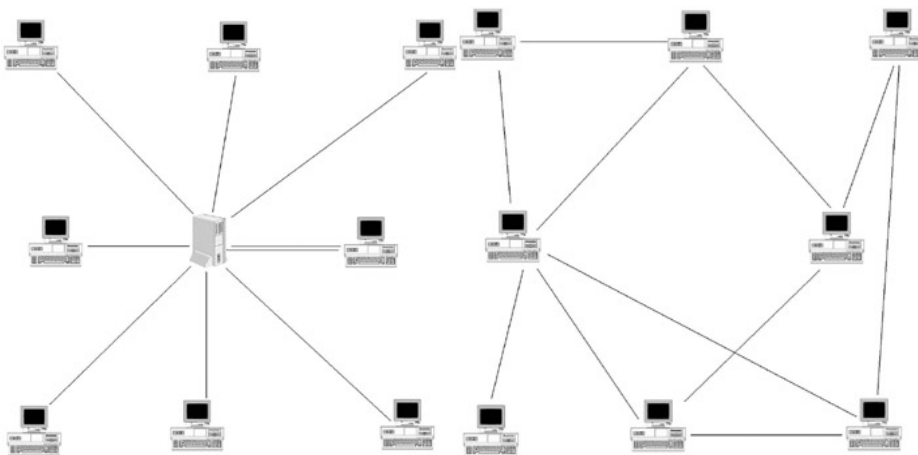
Troslojna klijent/server arhitektura (IP) predstavlja tip mreže koji se sastoji od „tankih klijenata“, aplikativnog veb-servera i servera baze podataka. „Tanki klijenti“ su mrežni PC kompjuteri sa niskom cijenom (tj. siromašnim hardverskim performansama) ili neki drugi uređaji koji obezbjeđuju browser bazirani korisnički interfejs i izvršavanje apleta – malih aplikativnih programa. Aplikativni veb-server i server baze podataka obezbjeđuju mrežni operativni sistem, izvršavanje veb aplikativnog softvera i upravljanje bazom podataka (slika 1.32.). Ovaj tip mreže danas predstavlja standard u poslovnim primjenama.



Slika 1.32. Troslojna klijent/server arhitektura (IP)

P2P (Peer-to-Peer) mreže jesu mreže istog prioriteta, u kojima svi računari u mreži imaju isti prioritet.

Slika 1.33. prikazuje dva osnovna modela P2P mreža.



Slika 1.33. Napster arhitektura (lijevo) i Gnutella arhitektura (desno)

Kod Napster arhitekture P2P softver povezuje svaki računar sa centralnim serverom, koji sadrži direktorijum svih ostalih korisnika u mreži. Kada neki korisnik zatraži neku datoteku, softver pretražuje taj direktorijum i pronalazi sve trenutno aktivne korisnike na mreži koji imaju zahtijevanu datoteku. Pronađenu listu korisnika, zatim, prosljeđuje računar koji je uputio zahtjev. Ova lista zapravo predstavlja skup linkova. Klikom na neki od tih linkova softver ostvaruje direktnu konekciju sa odabranim linkom i automatski prenosi zahtijevanu datoteku sa njegovog hard diska na hard disk korisnika koji je inicirao zahtjev.

Gnutella arhitektura jeste potpuna P2P mreža, u kojoj je svaki računar u mreži povezan sa svima ostalim direktno, a ne preko centralnog servera. Računar može uspostaviti vezu sa bilo kojim drugim aktivnim računarom u mreži. Svaki korisnik ima listu aktivnih linkova i kada se zahtijeva određena datoteka, softver mora pretražiti svakog aktivnog korisnika u mreži.

Može se reći da centralni server kod napster arhitekture predstavlja i prednost i nedostatak. S jedne strane, on može biti usporen ili pretrpan od

premnogo korisnika i na taj način zakočiti čitavu mrežu, ali, sa druge strane, obezbjeđuje bolji integritet i sigurnost prenosa podataka u mreži. Što se tiče gnutela P2P mreža, njihov nedostatak jeste sporost i pojava grešaka prilikom prenosa.

1.4.7. MREŽNE TEHNOLOGIJE

Komunikacione mreže predstavljaju tehnički najviše usmjeren dio informacionih tehnologija, i to sa rapidnim i brzim promjenama. Gledajući iz ugla poslovanja, može se reći da nisu potrebna detaljna znanja o tehničkim karakteristikama ovih mreža, ali je neophodno znanje o nekim važnim karakteristikama najbitnijih komponenti, koje će pomoći u efikasnom donošenju odluka u vezi sa mrežnim alternativama.

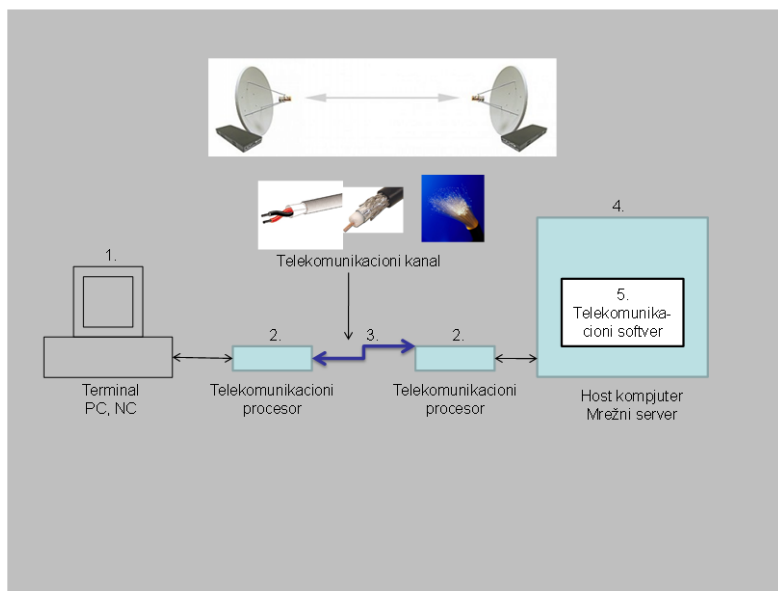
Komponente komunikacionih mreža. – Uopšteno rečeno, komunikaciona mreža jeste sistem u kome pošiljalac šalje poruku primaocu preko kanala sastavljenog od nekoliko tipova medijuma. Slika 1.34. prikazuje jednostavan konceptualni model komunikacione mreže, koji se sastoji od 5 osnovnih kategorija komponenti:

- **Terminali** su obično umreženi personalni kompjuteri. Međutim, i bilo koji ulazno-izlazni uređaj, koji koristi mrežu da šalje i prima podatke, jeste terminal (npr. telefon).
- **Telekomunikacioni procesori** podržavaju slanje i prijem podataka između terminala i kompjutera. Ovi uređaji, kao što su modemi, svičeri i ruteri, obavljaju razne kontrolne i druge funkcije u telekomunikacionoj mreži. Na primjer, konvertuju podatke iz digitalnih u analogne i obrnuto, kodiraju i dekodiraju podatke, kontrolišu brzinu, tačnost i efikasnost komunikacionih tokova i dr.
- **Telekomunikacioni kanali.** – Preko njih se podaci šalju i primaju u mreži. Telekomunikacioni kanali mogu koristiti medije poput bakarne žice, koaksijalnog kabla ili fiber-optičkog kabla, kao i bežične sisteme kao što su mikrotalasni, komunikacioni sateliti, radio i ćelijski sistem i dr.
- **Kompjuteri** različitih karakteristika i veličina koriste se u mreži za upravljanje procesiranjem informacija. Tako se kod većih mreža kao *host* kompjuter može koristiti *mainframe*, a *midrange* kao *front-end* procesor, dok se kod manjih mreža kao kompletan mrežni server može koristiti i mikroračunar.

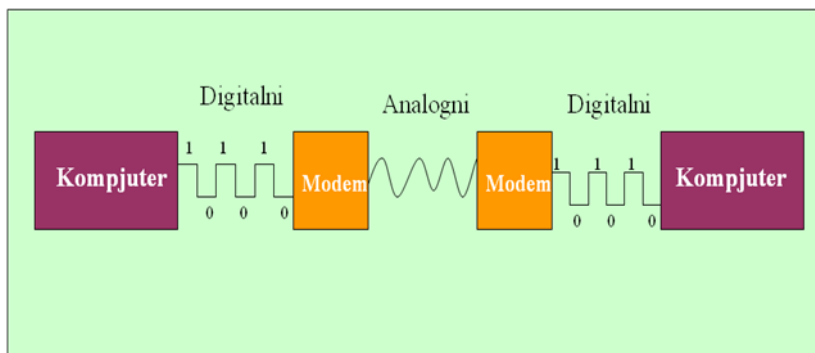
- **Telekomunikacioni kontrolni softver** sastoji se od programa koji kontrolišu telekomunikacione aktivnosti i upravljaju funkcijama mreže. Primjeri za to su telekomunikacioni monitori kod mainframe host kompjutera, mrežni operativni sistemi kod mrežnog servera i web browseri kod mikrokompjutera.

Telekomunikacioni procesori. – U telekomunikacione procesore spadaju modemi, multiplekseri i međumrežni procesori kao što su svičeri i ruteri.

Modemi konvertuju digitalne kompjuterske signale koje emituje pošiljalac u analogne frekvencije kako bi se oni mogli prenijeti putem standardne telefonske linije. Još jedan modem na drugom kraju komunikacione linije konvertuje poslate podatke obrnuto u digitalni signal za terminal primaoca (slika 1.35.). Ovaj proces se, dakle, sastoji od modulacije i demodulacije, pa sam naziv modem potiče od ove dvije riječi. Postoje različiti formati modema, uključujući samostojeće jedinice, ali i modemske kartice za laptop PC računare. Osim osnovne funkcije modulacije i demodulacije, mnogi modemi danas imaju i druge funkcije, kao što su kontrola greške u prenosu, automatsko pozivanje telefonskog broja i odgovaranje, rad sa faksovima i drugo.



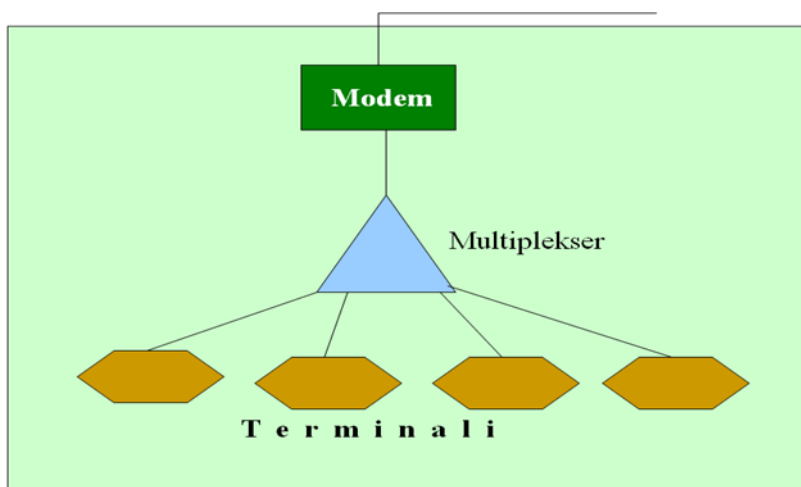
Slika 1.34. Konceptualni model komunikacione mreže



Slika 1.35. Uloga modema

Upotreba modema je nužna jer su standardne telefonske mreže u početku dizajnirane tako da prenose kontinuirane analogne signale (elektromagnetne frekvencije), koje generiše ljudski glas preko telefona. Sa druge strane, kompjuterski podaci su u digitalnoj formi (voltažni signali). Dakle, potreban je uređaj za konvertovanje digitalnih signala u odgovarajuće analogne frekvencije za prenos, i obrnuto. Međutim, danas je u svijetu sve više digitalnih mreža koje prenose samo digitalne signale i analogno-digitalna konverzija im nije potrebna. Ipak, one i dalje koriste uređaje poznate pod nazivom digitalni modemi za podršku nekih drugih telekomunikacionih funkcija.

Multiplekser jeste komunikacioni procesor koji omogućava simultano slanje podataka sa više terminala kroz jedan komunikacioni kanal (slika 1.36.). Multiplekser spaja pošiljke više terminala na jednom kraju komunikacionog kanala, dok sličan uređaj na drugom kraju ove pošiljke razdvaja. Multiplekser ovo obavlja na dva načina. Jedan način odnosi se na dijeljenje jednog brzog kanala na nekoliko sporih kanala, što predstavlja dijeljenje frekvencija. Drugi način ostvaruje se podjelom vremena, pri čemu multiplekser svakom terminalu dodjeljuje brzi komunikacioni kanal za veoma kratki vremenski trenutak. Veoma je popularan statistički multiplekser, koji umjesto da svakom terminalu dodjeljuje podjednak vremenski slot, dinamički alokira vremena, zavisno od prioriteta koji se određuje na osnovu nekih statistika (broja pristupa, brzine terminala i dr.) i/ili od strane telekomunikacionog menadžera.



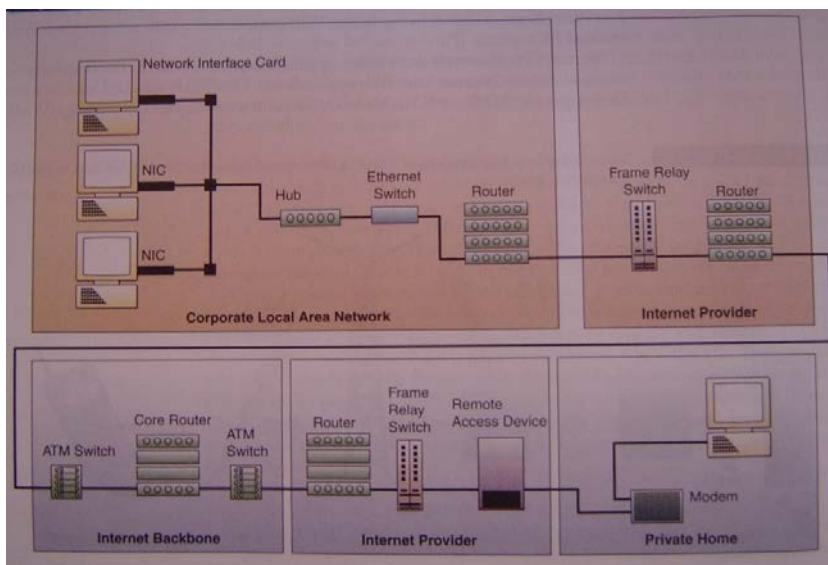
Slika 1.36. Uloga multipleksera

Međumrežni procesori predstavljaju uređaje koji povezuju različite mreže i njihove djelove. Tu spadaju svičeri, ruteri, hubovi i gateveyovi. Svič je komunikacioni procesor koji povezuje komunikacione cikluse unutar mreže kako bi telekomunikaciona poruka stigla do zahtijevane destinacije. Ruteri su inteligentni komunikacioni procesori koji povezuju mreže bazirane na različitim protokolima kako bi poruka bila usmjerena prema njoj destinaciji. Hub je komunikacioni procesor koji obavlja svičing između više konekcija (portova) za jedan mrežni resurs. Mreže koje koriste različite komunikacione arhitekture povezuju se pomoću gatewaya. Slika 1.37. prikazuje upotrebu međumrežnih procesora kod tipične internet konekcije iz privatne kuće na udaljeni organizacioni LAN.

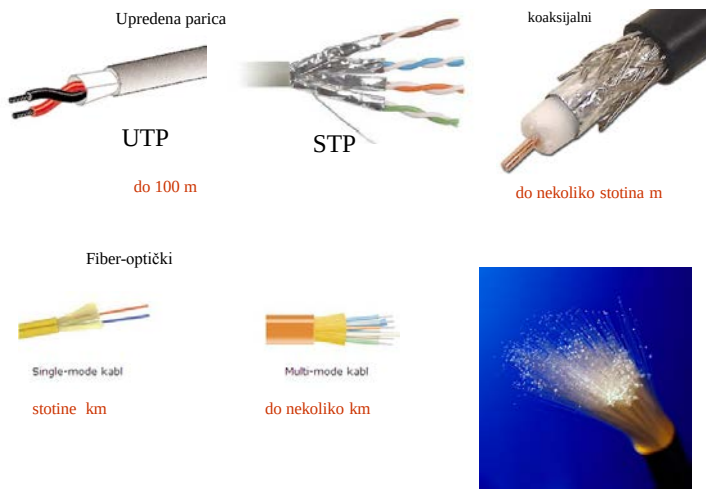
Telekomunikacioni mediji. –Telekomunikacioni kanali koriste različite telekomunikacione medije, kao što su upredena parica, koaksijalni kabal, fiber-optički kabal i drugo, za fizičko povezivanje uređaja u mreži.

Upredena parica jeste standardna telefonska žica, koja se sastoji od bakarne žice usukane u parovima. Signali se prenose putem elektromagnetnih talasa. Upredanje se obavlja da bi se spriječio električni šum, tj. miješanje signala. Korišćena je za kućne i kancelarijske telefonske sisteme, ali i za mnoge lokalne mreže. Veoma je osjetljiva na spoljašnje uticaje i sa povećanjem rastojanja javlja se slabljenje signala, pa se moraju

koristiti pojačivači na svakih 100 metara. Iz tog razloga koristi se samo za manja rastojanja. Javlja se u vidu UTP i STP kabla (slika 1.38.).



Slika1.37. Upotreba međumrežnih procesora kod internet konekcije



Slika 1.38. Vrste telekomunikacionih medija

Koaksijalni kabal jeste debela bakarna žica obmotana izolatorom, koja prenosi podatke u vidu elektromagnetnih talasa (slika 1.38.). Izolator minimizira miješanje i deformisanje signala prolaskom kroz žicu. Sa rastojanjem signal slabi, tako da se moraju koristiti pojačivači signala na svakih 500 m. Namijenjeni su za manja rastojanja. Mogu prenositi podatke u analognom i digitalnom formatu. Više koaksijalnih kablova može biti objedinjeno u jedan radi lakše instalacije. Teški su, nesavitljivi i kompleksni za ugradnju. Imaju brzinu prenosa podataka od oko 200M BPS i koriste se za kablovsku televiziju, glavne vodove lokalne mreže i drugo.

Fiber-optički kabal predstavlja revolucionarno otkriće u oblasti mreža. Zahvaljujući njemu, mreže danas povezuju velika geografska rastojanja.

Sastoji se od nekoliko hiljada staklenih vlakana (fibera) u zaštitnom omotaču. Podaci se putem njega prenose u vidu svjetlosnih signala generisanih laserom ili LED diodom. Dimenzije i težina fiber-optičkog kabla su manje a brzina prenosa podataka je oko 640 puta veća od brzine koaksijalnog kabla. Fiber-optički kablovi su otporni na elektromagnetna zračenja i ne generišu ih, imaju manju potrebu za retransmisijom signala, prenos je manje podložan greškama i otporniji su na udare nego drugi mediji. Zbog svojih pozitivnih karakteristika ovaj komunikacioni medijum je u svijetu zamijenio druge u mnogim aplikacijama. Jedina negativna strana jeste njegova visoka cijena, pa se kod nas uglavnom koristi samo za glavni vod mreže.

Pojavljaju se u vidu single-mode i multi-mode kabla (slika 1.38.). Single-mod kabal je jeftiniji, ali se kod njega za emitovanje svjetlosnih signala koristi laserska tehnologija, koja je skuplja od tehnologije zasnovane na LED diodama, a koristi se kod multi-mode optike. Laserska tehnika ima mogućnost mnogo bržeg emitovanja svjetlosnih impulsa od LED diode, pa je prenos signala brži i može se prenositi na veoma velika rastojanja. Iz tog razloga single-mode optika se koristi za rastojanja na nivou gradova, država, pa čak i kontinenata, dok je multi-mode kabal povoljniji za rastojanja od nekoliko kilometara.

Bežične tehnologije. – Bežične telekomunikacione tehnologije zasnovane na radio-talasima, mikrotalasima, infracrvenim i svjetlosnim signalima – omogućavaju digitalne komunikacije između komunikacionih uređaja bez njihovog povezivanja žicom. Bežične tehnologije obuhvataju zemaljske

mikrotalasne sisteme, satelitske komunikacione sisteme, celularne i PCS telefonske sisteme, bežične LAN-ove i bežične internet tehnologije.

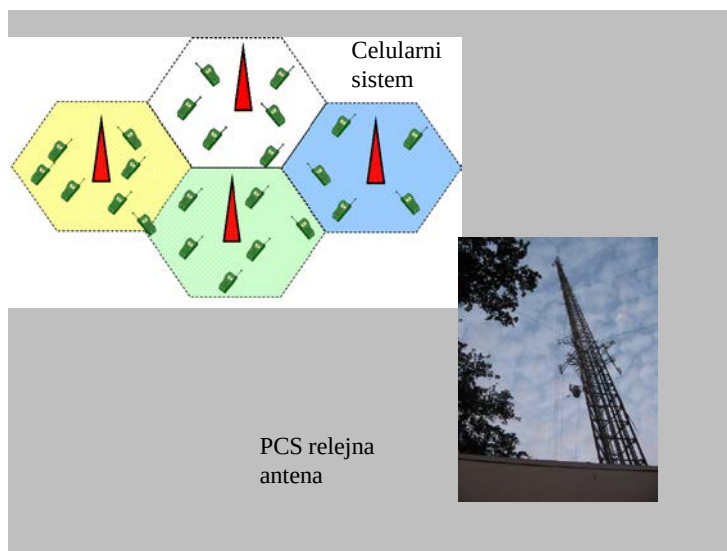
Zemaljski mikrotalasni sistemi prenose veoma brze radio signale između relejskih stanica udaljenih međusobno aproksimativno 50 km. Mikrotalasne antene su obično smještene na krovovima visokih zgrada, tornjevima, brdima ili planinskim vrhovima. Još uvijek su popularan medijum kako u mrežama dalekog dometa, tako i u gradskim metropola mrežama.

Satelitski komunikacioni sistem takođe koristi mikrotalase kao komunikacioni medij. Većina komunikacionih satelita smještena je u Zemljinoj orbiti, koja je od Ekvadora udaljena aproksimativno 35000 km. Sateliti se snabdijevaju solarnom energijom i odašilju signale brzinom od nekoliko stotina miliona bita po sekundi. Zemaljske stanice pomoću satelitskih antena usmjeravaju mikrotalasne signale prema satelitima, koji šire i šalju ove signale drugim zemaljskim stanicama hiljadama kilometara naokolo. U početku se satelitska komunikacija koristila samo za glasovne i video prenose. Danas se koristi i za veoma brzi prenos velike količine podataka u bilo kom formatu. S obzirom na to da je riječ o komunikacijama na veoma veliku daljinu i da zahtijevaju određeno vremensko kašnjenje, one nisu pogodne za real-time procesiranje podataka. Satelitske tehnologije se u poslovnim komunikacijama koriste na različite načine. Neke kompanije koriste mreže malih satelitskih antena da povežu svoja skladišta ili prodavnice kao i druga udaljena radna mjesta pomoću satelita. Takođe se za uspješne poslovne komunikacije koriste i satelitske mreže u nižim orbitama (oko 800 km udaljenim od zemlje).

Celularni i PCS telefonski sistemi koriste različite radio-komunikacione tehnologije. Sve one dijele geografski prostor na mala područja – ćelije od oko nekoliko kvadratnih kilometara. Svaka ćelija ima svoj odašiljač ili relejnu radio-antenu za prenos poziva od jedne ćelije do druge (slika 1.39.). Ukoliko se korisnici mobilnih telefona kreću iz jednog područja u drugo, prenos njihovih poziva kontrolišu i koordiniraju kompjuteri i drugi komunikacioni procesori. Celularni telefonski sistemi dugo su koristili analogne komunikacione tehnologije. Noviji celularni sistemi mobilne telefonije koriste digitalne tehnologije koje obezbjeđuju veći kapacitet i sigurnost, kao i dodatne napredne servise. PCS (Personal Communications Services) sistem jeste digitalna telefonska mreža sa visokim frekvencijama radio-talasa (oko 1900 MHz, za razliku od obične mobilne telefonije sa 800 do 900 MHz) i manjom snagom. Njena antenska mreža je gušća od

celularne, što omogućava korišćenje telefona i u tunelima, metroima, vozovima i drugdje, a troškovi funkcionisanja i korišćenja su manji. Omogućava vezu sa internetom kao i video i multimedijalne komunikacije.

Bežični LAN. – Sprovođenje kablova u kancelarijama i zgradama za lokalne mreže često je veoma težak i skup posao. Stare zgrade najčešće nemaju kanale za koaksijalne kablove, a kod novih neadekvatno postavljeni kanali mogu izazvati nužnost potpune relokacije LAN radnih stanica i drugih komponenti. Jedno od rješenja ovakvih problema jeste bežični LAN uz upotrebu neke od bežičnih tehnologija. Najčešće su korišćene digitalne radio-tehnologije nižih i visokih frekvencija, kao i infracrveni zraci. Najpopularniji su wi-fi (wireless fidelity) LAN-ovi, koji su brži i jeftiniji od standardnih žičnih LAN-ova. Pri upotrebi ove tehnologije PC i laptop računari povezuju se na internet i druge mreže preko wi-fi modema.



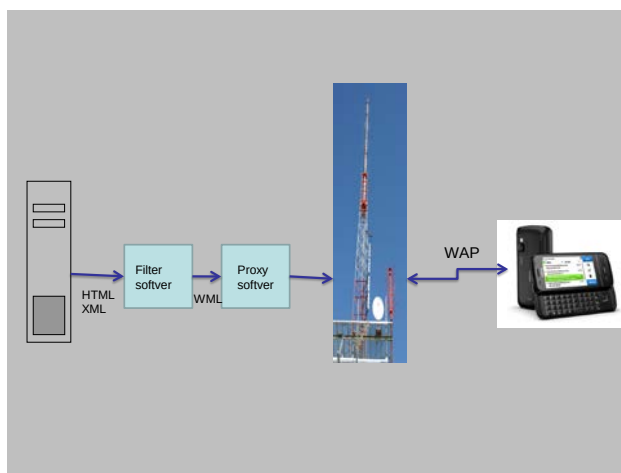
Slika 1.39. Celularni sistem i PCS relejna antena

Bežične internet tehnologije. – Bežični pristup internetu, intranetovima i ektranetovima raste uporedo sa razvojem sredstava za pristup veb-informacijama. Pametni telefoni, pejdžeri, PDA i drugi portabilni komunikacioni uređaji predstavljaju veoma tanke klijente u bežičnim mrežama. Razvoj standarda WAP (wireless access protocol) uticao je na razvoj velikog broja bežičnih veb-aplikacija i servisa.

Tehnologija 3G dostiže brzinu prenosa koja omogućava izvršavanje video i multimedijalnih aplikacija na mobilnim telefonima, tj. multimedijalni pristup webu.

PDA (Personal Digital Assistant), može da šalje i prima elektronsku poštu i pristupa webu pomoću specijalne tehnologije koja generiše korisnički dizajnirane veb-strane za mnoge popularne veb-sajtove.

Slika 1.40. ilustruje način na koji se standardne veb-strane (u HTML ili XML formatu) procesiraju kako bi bile dostupne nekom bežičnom uređaju. Vidimo da se one prvo prevode u WML (wireless markup language) pomoću filter softvera. Zatim se preprocesiraju pomoću proxy softvera u format pogodan za bežični prenos. Na kraju se u tom formatu prenose do bežičnog uređaja po WAP standardu.



Slika 1.40. Bežični pristup webu

Način i brzina prenosa podataka. – Prenos podataka putem telekomunikacionih kanala može biti analogni i digitalni, i o tome je već bilo riječi. Digitalni prenos može biti asinhroni i sinhroni. Kod asinhronog prenosa podaci se prenose karakter po karakter, dok se kod sinhronog prenose blokovi karaktera. Dalje, prenos može biti jednostruki, pri kome se podaci prenose samo u jednom smjeru, poludvostruki – podaci se prenose u oba smjera, ali u različitim vremenima i dvostruki, kod koga podaci kroz prenosni medij prolaze u oba smjera u isto vrijeme. Brzina prenosa podataka se mjeri bitovima po sekundi **BPS**, dok kapacitet telekomunikacionih kanala

predstavlja razliku najviših i najnižih frekvencija u kanalu, i od njega zavisi opseg, tj. **bandwidth** kanala (uskopojasni i širokopojasni kanali).

Osim osnovne jedinice BPS, koriste se i veće jedinice, kao što su:

KBPS (kilobita po sekundi) = hiljadu bita po sekundi,

MBPS (megabita po sekundi) = milion bita po sekundi i

GBPS (gigabita po sekundi) = bilion bita po sekundi.

Telekomunikacioni softver. – Softver predstavlja vitalnu komponentu računarskih mreža. Nalazi se na serverima, PC radnim stanicama i nekim komunikacionim procesorima kao što su multiplekseri i ruteri. U telekomunikacioni softver spadaju mrežni operativni sistemi i telekomunikacioni monitori. Oni određuju prioritete u prenosu, rutiraju i preusmjeravaju poruke, biraju odgovarajuće terminale i redove čekanja pri zahtjevu za prenos, otkrivaju i ispravljaju greške pri prenosu, bilježe statistiku o mrežnim aktivnostima i štite mrežu od neautorizovanih pristupa. Osnovne funkcije upravljanja mrežom su:

- **Upravljanje saobraćajem.** – Softver upravlja mrežnim resursima i saobraćajem u mreži kako bi se izbjegla gužva i istovremeno optimizuje nivoe mrežnih usluga za korisnike.
- **Sigurnost.** – Softver obezbjeđuje autentifikaciju, enkripciju i realizuje druge sigurnosne pretpostavke.
- **Nadgledanje mreže.** – Softver nadgleda cjelokupnu mrežu i informiše administratore o potencijalnim problemima prije nego se oni pojave.
- **Planiranje kapaciteta.** – Softver razmatra mrežne resurse, saobraćajne obrasce i korisničke potrebe u cilju efikasnijeg zadovoljenja tih potreba, ukoliko se one mijenjaju.

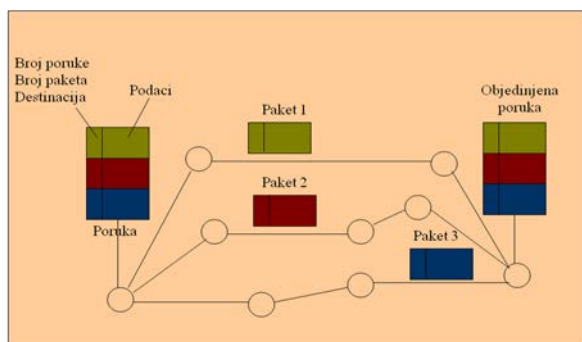
Mrežne arhitekture i protokoli. – Nedostatak standarda za interfejsse između hardvera, softvera i komunikacionih kanala telekomunikacionih mreža, povećava troškove, a smanjuje efektivnost i efektivnost. Telekomunikacioni proizvođači i nacionalne i internacionalne organizacije razvili su standarde, poznate pod nazivom protokoli, i generalne planove, tj. mrežne arhitekture, u cilju razvoja naprednih komunikacionih mreža.

Protokoli su standardni skupovi pravila i procedura koji kontrolišu komunikacije u mreži. Zadatak mrežnih arhitektura jeste standardizacija i

kompatibilnost ovih komunikacionih protokola. Jedan primjer protokola jeste standard za fizičke karakteristike kablova i konektora između terminala, kompjutera, modema i komunikacionih linija. Drugi protokoli kontrolišu prijem podataka u mreži, svič tehnike, međumrežno povezivanje i drugo.

Mrežne arhitekture promovišu otvoreno, jednostavno, fleksibilno i efikasno telekomunikaciono okruženje. Ovo se postiže upotrebom standardnih protokola, standardnim hardverskim i softverskim interfejsima i dizajnom standardnog interfejsa na više nivoa između krajnjih korisnika i kompjuterskih sistema.

Mrežna arhitektura i komunikacioni tokovi u mreži predstavljaju veoma kompleksan skup aktivnosti (slika 1.41.). Prvo, računar pošiljalac, tj. njegova korisnička aplikacija inicira zahtjev za prenos podataka. Podaci se dijele u pakete, a zatim formatiraju i kodiraju za prenos. Svakom paketu se određuje putanja u mreži, tj. dodjeljuje mu se odgovarajuća adresa. Paketi se transportuju između čvorova u mreži, pri čemu se vodi računa o njihovoj sigurnosti i ispravnosti. Kod računara primaoca paketi se prihvataju, oslobađaju pridruženih podataka (kao što su adresa, kontrola greške i drugo), povezuju se u logičku cjelinu i predaju korisničkoj aplikaciji.



Slika 1.41. Prenos poruka i paketa u mreži

ISO je definisala **OSI** (Open Systems Interconnection) model sa sedam nivoa kao standardni model mrežne arhitekture. OSI standard dijeli komunikacione funkcije u sedam nivoa i na taj način omogućava modularni razvoj mrežne arhitekture.

1. **Aplikativni nivo** obezbjeđuje komunikacione servise za korisničke aplikacije (iniciranje i prihvatanje zahtjeva).
2. **Prezentacioni nivo** obezbjeđuje odgovarajući format podataka za prenos (formatiranje, kodiranje, dekodiranje).
3. **Nivo sesije** podržava izvršavanje telekomunikacione sesije (kontrola protoka saobraćaja i razmjene podataka).
4. **Transportni nivo** podržava organizaciju transfera podataka između čvorova u mreži (pouzdanost, sigurnost i kvalitet transporta podataka, ispravke greške).
5. **Mrežni nivo** kontroliše putanju podataka obezbjeđujući konekciju između mrežnih linkova (dodjeljivanje adrese paketima, dodjeljivanje prioriteta i drugo).
6. **Nivo veze** vrši pakovanje i raspakivanje podataka u pakete, kao i kontrolu greške.
7. **Fizički nivo** obezbjeđuje fizički prenos podataka preko telekomunikacionih medija u mreži (prenos paketa u vidu niza bitova).

TCP/IP (Transmission Control Protocol / Internet Protocol) jeste standardni telekomunikacioni protokol kojeg koristi internet.

Nastao je kao dio OSI arhitekture, ali se kasnije razvio kao samostalni skup protokola od 5 nivoa, koji pokrivaju funkcionalnosti svih sedam nivoa OSI arhitekture. Prije pojave TCP protokola, ukoliko bi se neki paket izgubio pri prenosu, komunikacije u mreži su se prekidale. TCP protokol je uveo kontrolu greške i ponovnu transmisiju izgubljenog paketa bez prekida komunikacija u mreži. IP protokol je uveo globalno adresiranje računara u mreži, što je omogućilo priključivanje velikog broja računara na internet.

TCP/IP zapravo predstavlja mrežnu arhitekturu interneta, a koriste ga i intranetovi i ektranetovi, pa se može reći da je to danas standardna mrežna arhitektura. Mnoge organizacije transformišu svoje klijent/server mreže u TCP/IP tehnologiju, tj. u mrežnu arhitekturu. Ovakve mreže zovu se IP mreže.

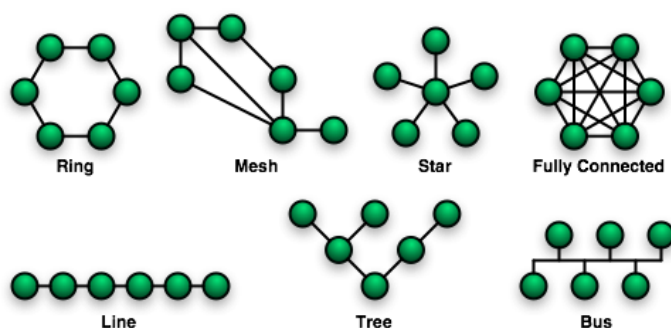
1.4.8. MREŽNE TOPOLOGIJE

Mrežne topologije su grafovi koji predstavljaju način povezivanja računara u mreži. Postoje tri osnovne topologije koje se koriste u WAN i LAN mrežama.

Topologija zvijezde (star) povezuje sve korisničke kompjutere sa jednim centralnim kompjuterom, preko koga se prenose sve komunikacione poruke. **Topologija prstena (ring)** povezuje korisničke kompjutere u krug, a poruke se prenose tako što prolaze kroz svaki računar redom u jednom smjeru. Može se prenositi više poruka odjednom kroz mrežu, ali obično samo u jednom smjeru. **Topologija magistrale (bus) ili linijska topologija (line)** jeste mreža u kojoj korisnički kompjuteri koriste zajedničku magistralu, tj. zajednički komunikacioni kanal. Poruke se kroz ovakvu mrežu prenose magistralom u oba smjera. Samo jedna poruka se može prenositi u jednom trenutku, a ukoliko dva kompjutera simultano razmjenjuju poruke može doći do kolizije.

Jedna varijacija prstenaste topologije jeste **mesh topologija**, u kojoj se koriste direktne komunikacione linije za povezivanje nekih ili svih kompjutera (**fully connected**) u prstenu sa svima ostalima. Još jedna varijacija jeste **drvo topologija (tree)**, koja spaja više linijskih topologija.

Ove topologije prikazane su na slici 1.42.



1.42. Mrežne topologije

Klijent/server mreže najčešće koriste kombinaciju ovih topologija. Po pravilu zvjezdasta topologija je više centralizovana, dok prstenasta i linijska imaju više decentralizovani pristup. Međutim, ovo ne mora da važi uvijek. Na primjer, centralni računar kod zvjezdaste topologije može da služi samo za upravljanje komunikacijom između lokalnih kompjutera. Mrežne topologije se razlikuju po svojim performansama, pouzdanosti i cijeni.

Kod topologije magistrale dodavanje novih čvorova ne prekida rad mreže, ali gust saobraćaj u mreži prouzrokuje loše performanse, pa je broj čvorova ograničen. Kvarovi na mrežnim karticama mogu dovesti do blokiranja mreže. Ova topologija gubi popularnost u praksi zbog takvih problema.

Kod zvjezdaste veze svi računari u mreži zavise od centralnog. Ako se on pokvari, raskidaju se veze između svih ostalih u mreži, a podaci iz mreže se mogu zauvijek izgubiti. Zato je važno da centralni računar (host computer) bude visoko pouzdan. Multiprocesorska arhitektura koja obezbjeđuje toleranciju otkaza jednog procesora jeste standardno rješenje za ovu topologiju. Međutim, u slučaju kvara nekog računara (osim centralnog), mreža nesmetano nastavlja sa radom. Proširivanje mreže ne dovodi do prekida rada, ali je ograničeno i skupo jer svaki novi korisnik zahtijeva posebni kabal.

U slučaju kvara jednog kompjutera u **prstenastoj mreži**, ostali kompjuteri ne mogu da nastave sa radom i da komuniciraju sa ostalim računarima u mreži. Dodavanje novih čvorova takođe zahtijeva prekid rada mreže. Ovi nedostaci prstenaste topologije danas se prevazilaze upotrebom softvera koji omogućava zaobilazanje pokvarenog čvora u informacionom toku. Za dodavanje novog čvora bez prekida rada mogu se koristiti specijalni konektori. Prstenasta topologija, za razliku od linijske, ima stabilne performanse bez obzira na broj korisnika u mreži.

1.4.9. MREŽNI SERVISI

Dial-up jeste tradicionalni analogni telefonski servis, u kome je obavezna direktna veza između dva čvora u mreži u toku transmisije podataka (circuit switching – komutirana veza). Ima malu brzinu prenosa oko 56 KBPS (slika 1.43.).

Packet switching jeste osnovna tehnika upravljanja transmisijom podataka, kod koje se dugački blokovi karaktera dijele na male pakete (kod X.25 standarda od 128 bajtova) koji sadrže i adresu primaoca i informaciju o greški, a šalju ih preko nekoliko različitih komunikacionih kanala, tj. rutiraju različitim putanjama u mreži. Svaki paket kroz mrežu putuje nezavisno. Kada dođu do predviđene destinacije ponovo se objedinjuju u jednu poruku (slika 1.41.).



1.43. Dial-up servis

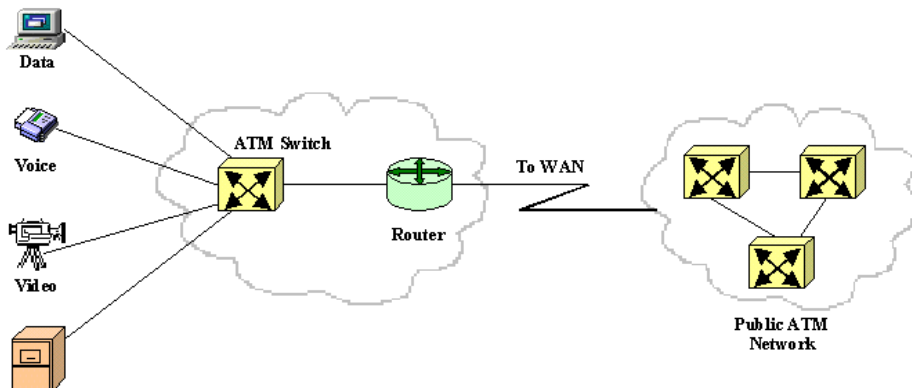
Frame relay jeste mrežni servis sličan paket switchingu, s tom razlikom što je brži od njega (iznad 1.544 MBPS). Ovaj servis dijeli podatke u frameove (okvire) slične paketima, ali se ne vrši kontrola greške. Koristi se kod pouzdanih kanala koji ne zahtijevaju čestu retransmisiju podataka zbog greške. Internet provajderi koriste ovu tehnologiju (slika 1.37.).

Većina kompanija danas koristi odvojene mreže za glasovne komunikacije, razne sopstvene servise i pristup i transmisiju podataka. Svaka od tih mreža zasnovana je na različitim tehnologijama (telefoni, pejdžeri i drugi bežični uređaji, IC komunikacija, klijent/server mreže, internet itd.). Mrežni servis **ATM (Asynchronous Transfer Mode)** može prevazići ovakve probleme, jer on omogućava efikasno upravljanje različitim vidovima komunikacije – glasovnim, video, grafičkim, podacima i dr. – u okviru jedne mreže (slika 1.44. lijevo).

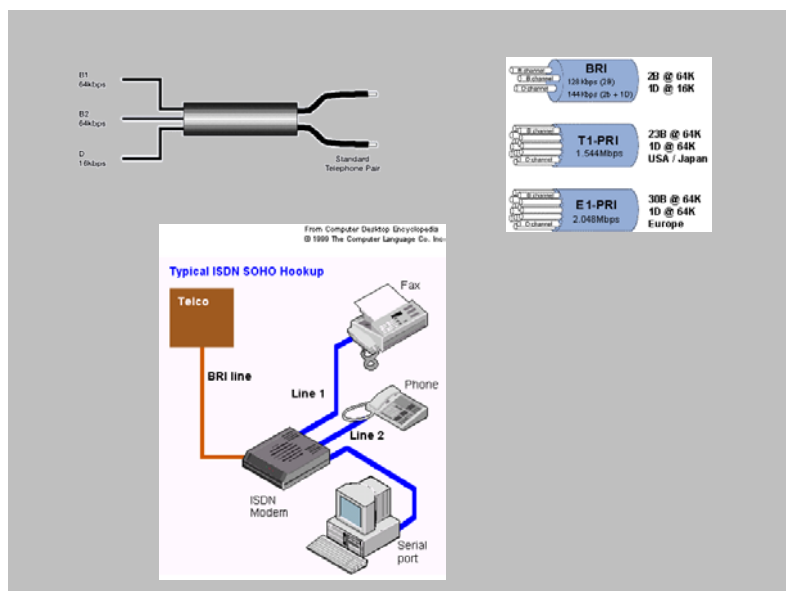
ATM tehnologija dijeli informacije u uniformne ćelije sa 53 grupe po 8 bajtova i prenosi podatke bez protokolarne konverzije. Može prenositi podatke između različitih hardverskih platformi, kao i između mreža različitih brzina. Obično povezuje LAN-ov, koji imaju protokole manjih brzina, sa WAN-ovima, koji operišu većim brzinama. ATM može prenositi više od 2.5 biliona bitova po sekundi. Koristi se kod internet kičme (slika 1.37. i slika 1.44. desno).

ISDN (Integrated Services Digital Network) jeste servis koji omogućava istovremeni prenos više različitih sadržaja (glas, podaci, grafika, video) putem jedne telefonske linije. Na primjer, organizacije ili pojedinci koji koriste ovaj servis mogu putem iste telefonske linije koristiti i internet i

telefon. Ovaj servis koristi odvojenu grupu B kanala za prenos informacija od D grupe kanala za kontrolne informacije. Brzina prenosa je od 128 KBPS do 2 MBPS, zavisno od broja B kanala (slika 1.45.).



Slika 1.44. Uloga ATM servisa kod LAN mreže i javne mreže – interneta



Slika 1.45. ISDN servis

DSL (Digital Subscriber Line) servis, kao i ISDN, koristi telefonsku liniju za istovremeni prenos glasa, podataka i videa, s tom razlikom što ima veći kapacitet prenosa podataka. Postoje dvije vrste DSL linija. Asimetrična DSL linija (ADSL) ima veliku brzinu prijema podataka (1.5 do 9 MBPS), ali manju brzinu slanja podataka (oko 640 KBPS). Simetrična DSL linija (SDSL) ima istu brzinu i prijema i slanja podataka (oko 3 MBPS).

T1 i T3 linije jesu specijalizovane telefonske linije koje se sastoje od nekoliko desetina do nekoliko stotina kanala, od kojih svaki može biti konfigurisan da prenosi glas ili podatke. Riječ je o linijama sa veoma velikim kapacitetom i brzinom prenosa podataka (oko 1.54 MBPS – za T1 i 45 MBPS – za T3 linije), koje se često koriste za internet konekcije velikog kapaciteta.

Servisi, tj. tehnologije, kao što su ISDN, DSL, T1 i T3, koje imaju veliku brzinu i kapacitet prenosa podataka i imaju mogućnost simultanog prenosa različitih vrsta podataka putem više kanala u okviru jednog komunikacionog medija, poznate su pod nazivom **broadband** (širokopojasni prenos).

1.4.10. MREŽNE ALTERNATIVE

Odlučivanje u oblasti računarskih mreža zahtijeva poznavanje mrežnih alternativa. Slika 1.46. prikazuje mrežne alternative koje su razjašnjene u ovom poglavlju.

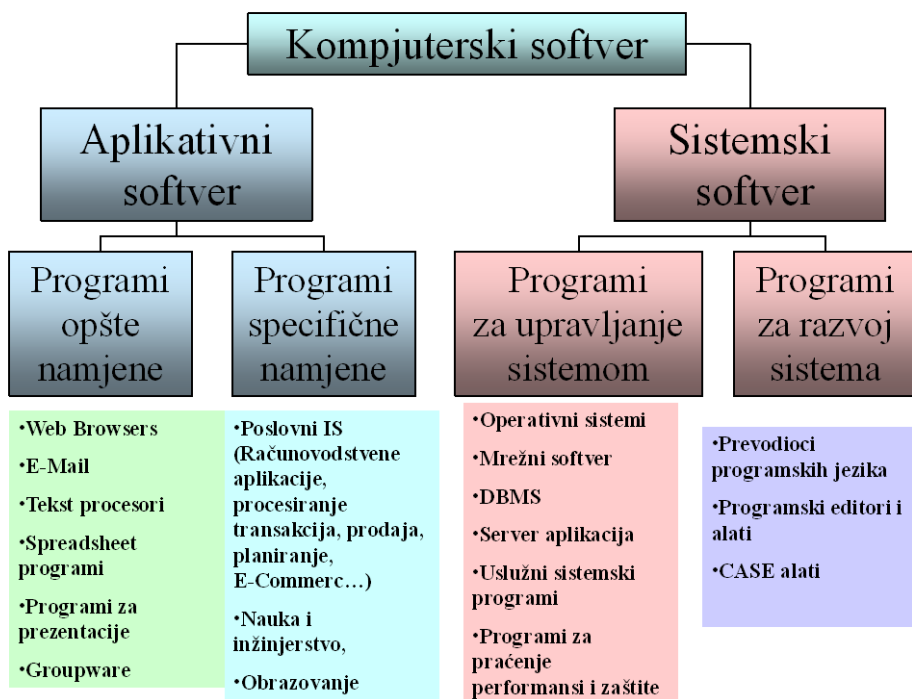
Mreže: Internet, intranet, extranet, WAN, LAN, klijent/server, troslojna klijent/server arhitektura, P2P
Mediji: Upređena parica, koaksijalni kabal, fiber-optički kabal, radio-mikrotalasi, komunikacioni sateliti, ćelularni i PCS sistemi, bežični mobilni i LAN sistemi
Procesori: Modemi, multiplekseri, svičeri, ruteri, hubovi, gateways, front-end procesori
Softver: Mrežni operativni sistemi, telekomunikacioni monitori, Web browseri
Kanali: Analogni/digitalni, svičovani/nesvičovani, svičing ciklusa, poruka, paketa, okvira, ćelija, bandwidth tehnologije
Topologije/Arhitekture : Zvijezda, prsten, magistrala; OSI I TCP/IP arhitekture i protokoli.

Slika 1.46. Mrežne alternative

1.5. RAČUNARSKI SOFTVER

Softver predstavlja skup programa koji se izvršavaju na računaru. U okviru ovog poglavlja biće razmatrane karakteristike, svrha i primjene glavnih tipova softvera.

Slika 1.47. prikazuje osnovne tipove i funkcije aplikativnog i sistemskog softvera. Aplikativni softver predstavlja programe namijenjene za rješavanje konkretnih problema, aktivnosti i zadataka korisnika računarskog sistema. Sistemski softver čini klasu programa koja obezbjeđuje efikasno funkcionisanje računarskog sistema i koordiniranje aktivnosti hardverskih resursa u zavisnosti od zahtjeva aplikativnog softvera. Sistemski softver predstavlja međusloj, tj. vezu između hardvera i aplikativnog softvera.



Slika 1.47. Podjela kompjuterskog softvera

1.5.1. SOFTVERSKI TRENDОВИ

U početku je glavni trend bio softver razvijen od strane profesionalnih programera iz same organizacije, tj. program koji su pisali sami korisnici. Zatim se trendovi kreću u pravcu softverskih paketa koji se kupuju od prodavaca softvera. Ovaj trend je doživio nagli rast sa razvojem manje skupih i za upotrebu lakih aplikativnih softverskih paketa i višenamjenskih softverskih paketa za mikroracunare. Najnoviji trendovi su softverski paketi dizajnirani za veb-tehnologije i kolaborativne aktivnosti. Ovakve veb orijentisane aplikativne pakete upotrebljavaju korisnici i radni timovi na internetu, kao i na kompanijskim intranet i ekstranet mrežama. Takođe, danas se mnogi softverski paketi mogu downloadovati, updateovati, iznajmiti ili zakupiti od aplikativnih servis provajdera (ASP) putem interneta i kompanijskog intraneta.

Softverske trendove prate i trendovi programskih jezika i alata. Oni se kreću od tehničkih mašinski zavisnih jezika koji koriste binarno bazirano ili simboličko kodiranje, preko proceduralnih jezika, koji koriste kratke naredbe i matematičke izraze za specificiranje programskih instrukcija. Dalji trendovi se kreću u pravcu vizuelnog grafičkog interfejsa za objektno orijentisano programiranje. Ovaj trend je značajno ubrzan pojavom neproceduralnih jezika četvrte generacije (4GL), koji su jednostavni za upotrebu.

Najnoviji trendovi u oblasti programskih jezika i alata jesu vještačka inteligencija kao i grafički korisnički interfejs (GUI – Graphical User Interface) i interfejs zasnovan na prirodnom jeziku, što čini softverske pakete lakšim za upotrebu. Neke mogućnosti vještačke inteligencije se koriste u mnogim tipovima softverskih paketa. Na primjer, softveri imaju inteligentne helpove poznate pod nazivom *wizard*, koji automatski izvršavaju neke softverske funkcije, kao što su generisanje grafikona kod spreadsheet programa ili generisanje izvještaja iz baze podataka.

Dakle, svi ovi trendovi konvergiraju ka softverskim paketima pete generacije, koji su višenamjenski, ekspertski, veb orijentisani, sa grafičkim interfejsom zasnovanim na prirodnom jeziku i koji treba da podrže produktivnost i saradnju i krajnjih korisnika i IS profesionalaca.

1.5.2. APLIKATIVNI SOFTVER – SOFTVER ZA KRAJNJE KORISNIKE

Slika 1.47. prikazuje da aplikativni softver podrazumijeva skup programa koji se mogu podijeliti na programe opšte namjene i programe specifične namjene.

Na tržištu postoje hiljade različitih **softverskih paketa specifične namjene** koji podržavaju poslovne i druge aktivnosti korisnika. Na primjer, poslovni paketi podržavaju upravljačke i operativne aktivnosti kao što su procesiranje transakcija, podrška odlučivanju, upravljanje prodajom, analiza investicija i elektronska trgovina. Softver specifične namjene u oblasti nauke i inženjerstva ima glavnu ulogu u istraživačkim i razvojnim programima, u dizajniranju i proizvodnji visoko kvalitetnih proizvoda i dr. Tu su takođe paketi namijenjeni obrazovanju, zabavi i sl.

Poslovni aplikativni softver. – Poslovni informacioni sistemi predstavljaju integrisanu kombinaciju kros-funkcionalnih sistema ili odvojenih funkcionalnih poslovnih sistema koji pokrivaju samo određenu poslovnu kategoriju.

Glavni kros-funkcionalni poslovni sistemi su:

ERP (Enterprise Resource Planing) jesu sistemi za planiranje i upravljanje resursima preduzeća (finansijskim, ljudskim, materijalnim i dr.).

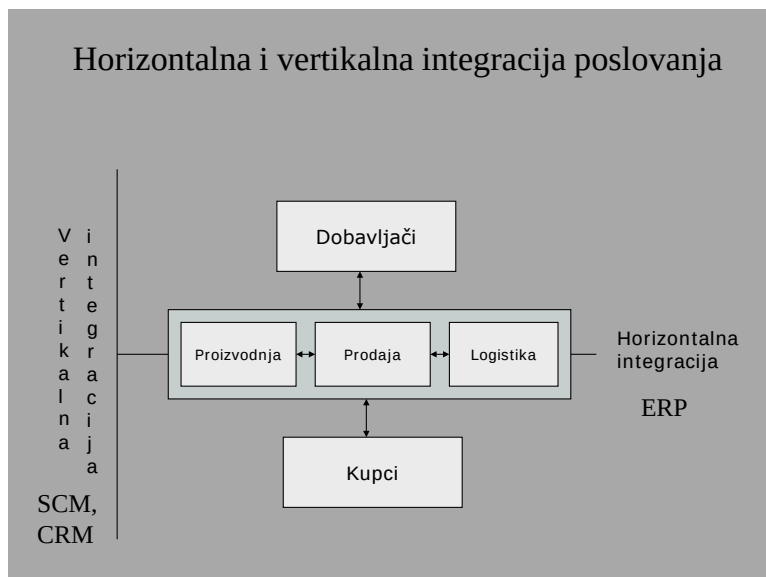
CRM (Customer Relationship Management) jesu sistemi za odnose sa korisnicima, tj. kupcima.

SCM (Supply Chain Management) predstavljaju sisteme koji omogućavaju elektronsko poslovanje (e-commerce i komunikacije) sa dobavljačima, distributerima i prodavcima koji čine aktere poslovnih aktivnosti u **lancima nabavke**.

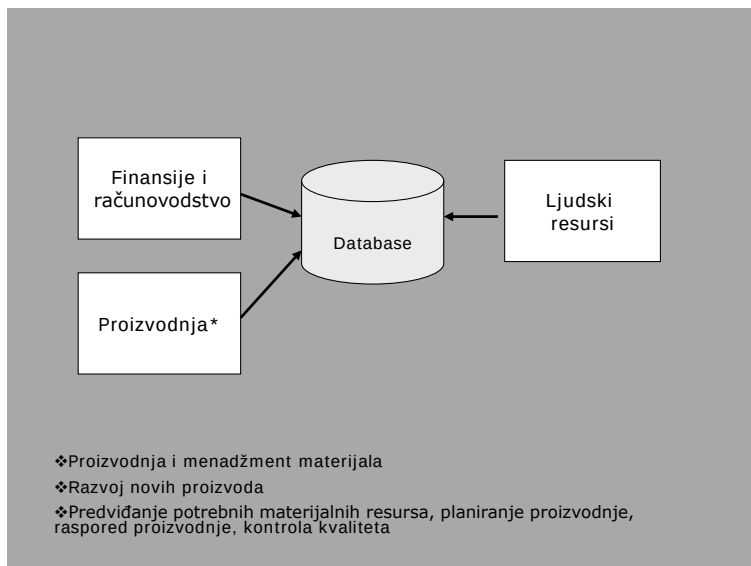
Kros-funkcionalni sistemi vrše horizontalnu, odnosno vertikalnu integraciju poslovanja, kao na slici 1.48.

Istorijski ERP sistemi nastali su od sistema za planiranje materijalnih i proizvodnih resursa MRP (*Material Requirements Planning*). Danas ERP sistemi automatizuju i integrišu poslovne procese u proizvodnji (planiranje

proizvodnje, logistika, prodaja, zalihe, narudžbe, distribucija), u finansijama i računovodstvu, kao i u upravljanju ljudskim resursima (back-office) – slika 1.49.

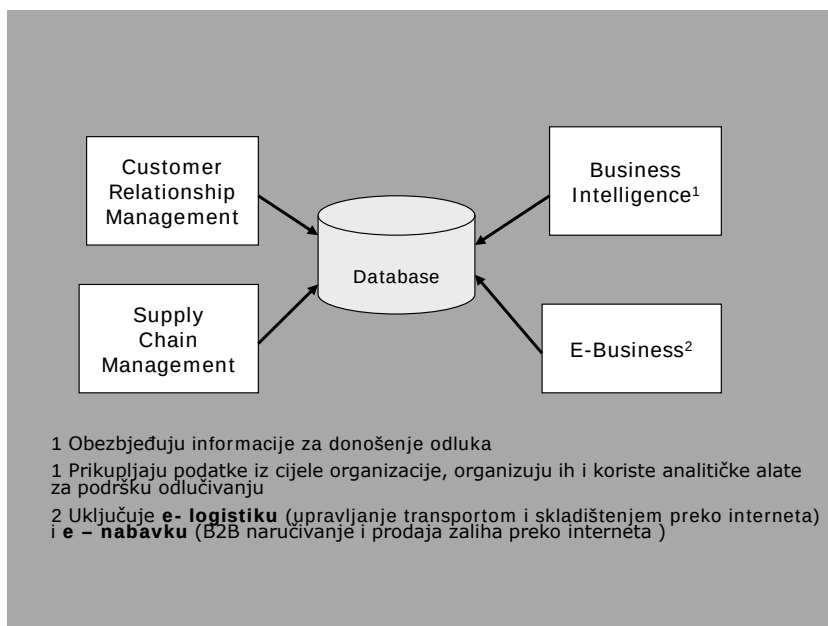


Slika 1.48. Uloga kros-funkcionalnih sistema u integraciji poslovanja



Slika 1.49. Moduli ERP sistema

Extended (prošireni) ERP sistemi danas obuhvataju i CRM i SCM sisteme, kao i sisteme poslovne inteligencije (business intelligence BI) i e-poslovanja, i tako čine univerzalni kros-funkcionalni softver koji se koristi u poslovanju. Slika 1.50. prikazuje komponente ovakvih ERP sistema, kao i ulogu BI komponente i komponente e-poslovanja. Ovakvi sistemi, osim unutrašnjeg, obuhvataju i spoljašnje poslovanje (front-end).



Slika 1.50. Moduli Extended ERP sistema

Extended ERP sistemi su obično gotovi standardizovani softverski paketi koji se pomoću parametara prilagođavaju pojedinačnoj organizaciji.

Najpoznatiji ERP paketi na tržištu su:

- **SAP – BusinessOne**, AiO, R/3 (zauzimaju oko 28,7% tržišta)
- **ORACLE – People Soft** (10,2%)
- **BAAN**
- **Microsoft Dynamics sistemi – NAV (Navision), AX (Axapta), GP (Great Plains), SL (Solomon)** (3,7%)
- **ASW – asw: dominus**

Uvođenje ERP sistema najčešće podrazumijeva spoljne konsultante koji vrše prilagođavanja sistema i obuku osoblja. Nekad je potreban i reinženjering, tj. prilagođavanje poslovnih procesa. Implementacija traje od nekoliko mjeseci do nekoliko godina, obavlja se korak po korak, što omogućava modularna struktura ERP sistema. Uvođenje ovih sistema donosi preduzeću niz prednosti.

Tako se, na primjer, velika svjetska kompanija Colgate Palmolive odlučila za uvođenje ERP sistema SAP R/3. Na taj korak se odlučila jer joj je bilo potrebno da radi lokalno, na velikom broju lokacija, a da upravlja globalno. Sistem je prvobitno uveden kao podrška Colgate lancu nabavke u najprofitabilnijim odjelima. Godine 2001. SAP je uveden u svim Colgate odjelima širom svijeta. Efekti su bili sljedeći:

- Prije ERP-a prijem i obrada narudžbenice trajali su od 2 do 7 dana, a sa uvođenjem sistema traju oko 4 sata. Kompletan proces od narudžbe do isporuke traje upola manje.
- Prije ERP-a broj pravovremenih isporuka bio je 91,5%, a poslije 97,5%.
- Broj korektnih isporuka iznosio je 97,5% prije ERP-a, a 99% poslije.
- Poslije uvođenja ERP-a zalihe su se smanjile za jednu trećinu.
- Potraživanja su sa 31,4 dana spala na 22,4 dana.
- Obrtni kapital kao procenat prodaje pao je sa 11,3% na 6,3%.
- Ukupni troškovi isporuke reducirani su za 10%.

Razlozi za uvođenje CRM sistema mogu se opravdati sljedećim činjenicama:

- Prodaja novom kupcu košta 6 puta više nego starom!
- Nezadovoljan kupac to svoje iskustvo prenijeće deseterici ljudi!
- Kompanija može povećati profit za 85% povećavajući zadržavanje kupaca za 5% na godišnjem nivou!
- Procenat od 70% kupaca koji se žale opet će kupovati kod kompanije koja se brzo pobrine da popravi grešku!
- Danas preduzeća posluju na principu **customer-centric** i **one to one** marketing strategije.
- Osnova za ovakav princip poslovanja jesu CRM sistemi.

CRM sistemi automatizuju i integrišu korisničke servise u prodaji, marketingu i servisiranju proizvoda. Omogućavaju analize i predviđanja. Sastoje se od sljedećih cjelina (modula):

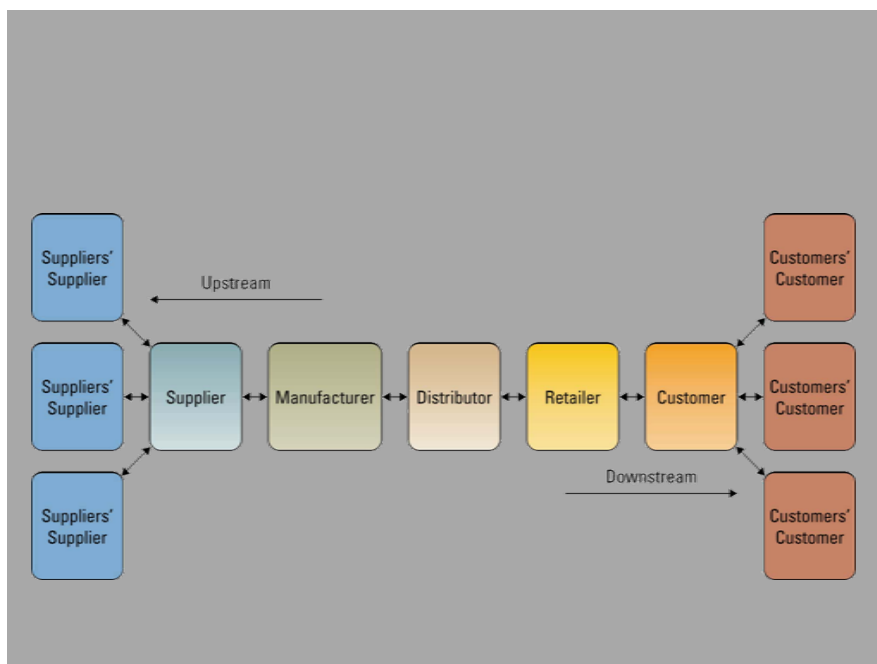
- **Prodaja.** – CRM memoriše sve kontakte i druge aktivnosti kupaca prilikom prodaje (identifikovanje i praćenje najboljih kupaca, njihovih potreba, kupovnih navika, tačaka kontakta i sl.).
- **Marketing.** – CRM omogućava memorisanje i mjerenje različitih vrsta marketinških kampanja, a mjere su broj klikova na sajt, odgovori na kampanju, aktivnosti, prihodi i slično.
- **Servisiranje proizvoda.** – CRM omogućava efikasno prihvatanje i rješavanje zahtjeva korisnika, ukoliko oni imaju problema sa proizvodom ili uslugom (Help desk).
- **Analiza i predviđanja.** – CRM omogućava analize i predviđanja u domenu prodaje, marketinga i usluga. Koristi podatke koji su prikupljeni u okviru prethodna tri modula. Realizuje se obično linkom na business intelligence.

Već smo rekli da današnja preduzeća posluju na principu **customer-centric** i **one to one**, odnosno na principu **ciljane** marketinške strategije. Zato gotovo najvažnija uloga CRM sistema jeste analiza i predviđanje na osnovu prikupljenih podataka. Ovakve analize omogućavaju BI sistemi pomoću OLAP i data mining alata. Tipične analize i predviđanja koje ovaj sistem omogućava jesu:

- **Profitabilnost kupaca:** Zašto su neki kupci profitabilni, a neki ne? Da li će neki novi komitent biti profitabilan?
- **Segmentacija tržišta:** Formiranje grupa kupaca sa zajedničkim karakteristikama (obično demografskim i psihografskim). Buduće marketinške aktivnosti i interakcija sa komitentom kroje se po mjerama grupe (segmenta).
- **Target (ciljni) marketing:** Kakva je sklonost kupaca iz ciljnog tržišnog segmenta za kupovinu nekih postojećih ili novih proizvoda?
- **Lojalnost kupaca (churn prediction):** Zašto su neki kupci otkazali lojalnost? Da li će neki kupac ostati lojalan ili ne?
- **Afinitet kupaca (market-basket analiza):** Koje proizvode kupac obično kupuje zajedno (npr. klijenti koji uzimaju Visa karticu obično uzimaju i Visa Electron karticu, klijenti koji uzimaju stambeni kredit obično uzimaju i kredit za auto itd.).

- **Unakrsna prodaja (cross selling):** Otkrivanje novih proizvoda koje će vjerovatno tražiti neki komitent, i koji će mu biti ponuđeni u prvom narednom kontaktu.

Mreža poslovnih relacija sa preduzećima od kojih se roba nabavlja, kao i sa onim koja robu prodaju – naziva se lanac nabavke (slika 1.51.).



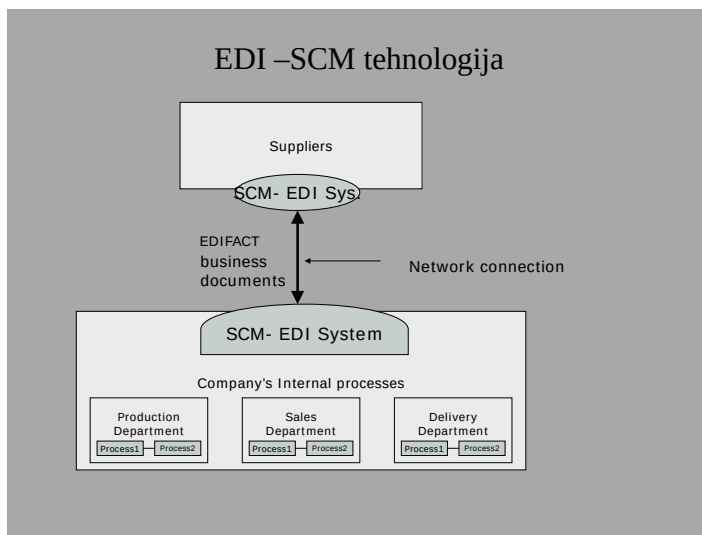
Slika 1.51. Lanac nabavke

Uloga SCM sistema jeste automatizacija lanaca nabavke. Osnovne funkcije SCM sistema, a u skladu sa tim i modul, jesu:

- planiranje (usklađuje zahtjeve za proizvodima sa nabavkom materijala);
- sourcing (utvrđuje od koga će se materijal najefikasnije i najpovoljnije nabaviti)
- proizvodnja (utvrđuje raspored proizvodnje i brine se da uvijek bude dovoljno materijala)
- isporuka proizvoda (omogućava obradu zahtjeva kupca i upravljanje magacinima)

- povraćaj (omogućava procesiranje reklamacije materijala i proizvoda).

Najvažnije aktivnosti u okviru svakog modula predstavlja razmjena poslovne dokumentacije. Ova razmjena realizuje se poštovanjem EDI ili XML standarda za prenos dokumenata. Slika 1.52. prikazuje dio SCM konekcije koja je zasnovana na EDI standardu.



Slika 1.52. Razmjena EDIFACT dokumenata kod SCM sistema

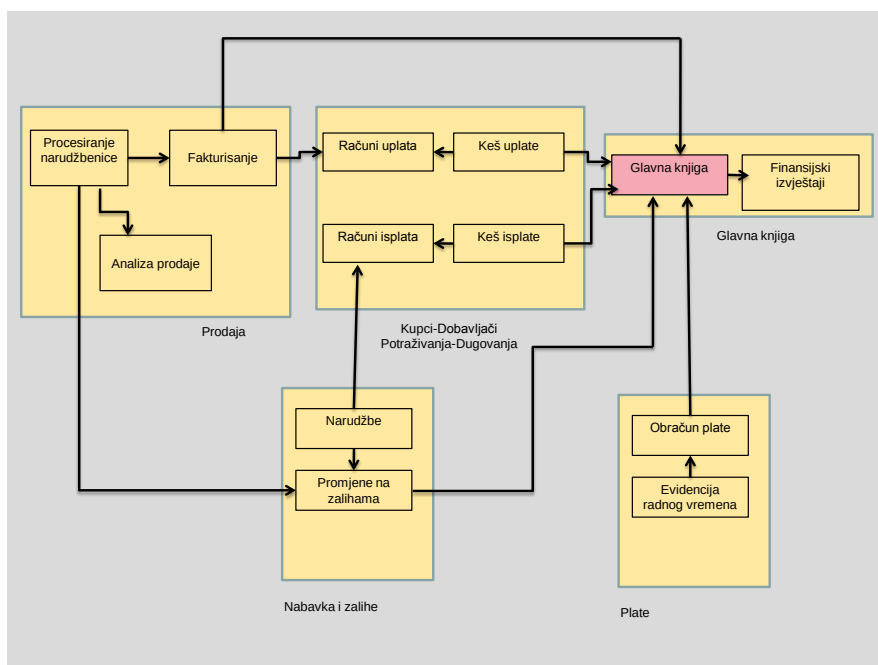
Glavni funkcionalni poslovni sistemi jesu sistemi za marketing (planiranje, promocija i prodaja postojećih proizvoda, razvoj novih proizvoda i novih tržišta), za proizvodnju (planiranje, automatizacija i kontrola procesa proizvodnje, CAD/CAM i CIM sistemi), za ljudske resurse (planiranje potreba, platni izvještaji, analiza učinka personala u poslovnim funkcijama, upravljanje ličnim podacima zaposlenih), za računovodstvo (procesiranje narudžbenica, kontrola zaliha, računi kupaca, računi dobavljača, obračun plata i glavna knjiga) i za finansijski menadžment (upravljanje gotovinom i investicijama, budžet, finansijsko planiranje i predviđanje).

Računovodstveni sistemi jesu najstariji i najrasprostranjeniji sistemi u poslovanju. Oni memorišu i izvještavaju o poslovnim transakcijama i drugim ekonomskim događajima. Većina ostalih poslovnih funkcija oslanja

se na njihove podatke. Zasnivaju se na klasičnom konceptu dvojnog knjiženja.

Kompjuterski zasnovani računovodstveni sistemi memorišu aktuelne i istorijske zapise i generišu važne finansijske izvještaje, kao što su bilanci stanja i uspjeha. Sistemi upravljačkog računovodstva fokusiraju se na planiranje i kontrolu poslovnih operacija. Oni posebno koriste analitičke izvještaje troškova, projekcije finansijskih izvještaja i druge analitičke izvještaje za komparaciju aktuelnih i predviđenih performansi.

Obično se računovodstveni sistemi sastoje od sljedećih 6 podsistema – modula (slika 1.53. prikazuje ove podsisteme i odnose među njima):



Slika 1.53. Moduli računovodstvenog sistema

- **Procesiranje narudžbenica (prodaja).** – Memorišu se i procesiraju narudžbe kupaca i produkuju se podaci za kontrolu zaliha i računi uplata. Fakturišu se narudžbenice i knjiže se podaci o prodaji u glavnoj knjizi.

- **Kontrola zaliha.** – Procesiraju se podaci koji reflektuju promjene na zalihama. Generišu se otpremnice. Knjiže se promjene na zalihama po otpremnici u glavnoj knjizi. Knjiže se podaci o nabavci po prijemnici u glavnoj knjizi.
- **Računi uplata (kupci-dugovanja).** – Memorišu se razne uplate, uplate kupaca i generišu se fakture za kupce. Generišu se mjesečni izvještaji o kupcima i dugovanjima. Knjiže se keš uplate u glavnoj knjizi.
- **Računi isplata (dobavljači-potraživanja).** – Memorišu se nabavke, njihovi iznosi, isplate dobavljačima i druge isplate. Generišu se keš menadžment izvještaji. Knjiže se keš isplate u glavnoj knjizi.
- **Plate.** – Memorišu se podaci o radu zaposlenih (sa vremenskih kartica) i obračunava se plata i generišu platni spiskovi. Knjiže se podaci o platama u glavnoj knjizi.
- **Glavna knjiga.** – Konsoliduju se podaci iz drugih podsistema, omogućavaju se zaključna knjiženja i generišu se periodični finansijski izvještaji.

Programi opšte namjene izvršavaju poslove standardnog procesiranja informacija za krajnje korisnike, kao što su na primjer obrada teksta, obrada tabela, izrada prezentacija i drugo. Oni značajno povećavaju produktivnost krajnjih korisnika. Oni takođe uključuju i programe koji podržavaju komunikacije i saradnju između radnih grupa i timova, kao što su web browseri, e-mail programi i groupware programi.

Programi opšte namjene su obično integrisani u pakete i poznati su pod nazivom **integrisani paketi**. Na tržištu postoje ovakvi integrisani paketi koje nude različiti proizvođači (najpoznatiji je Microsoft Office). Najpoznatiji proizvođači i proizvodi koje sadrže njihovi integrisani paketi dati su u tabeli 1.3.

Cijena integrisanog paketa znatno je manja od ukupne cijene pojedinačnih softverskih paketa. Svi programi koji čine integrisani paket imaju sličan GUI (ikone, toolbar i statusbar, menije i slično) koji daje isti izgled i osjećaj za rad i čini ih jednostavnijim za savladavanje i upotrebu. Programi u okviru integrisanog paketa imaju standardne alate kao što su help wizard i drugo. Njihova velika prednost jeste i mogućnost kopiranja i prenošenja djelova fajlova iz jednog u drugi, kao i međusobno importovanje fajlova.

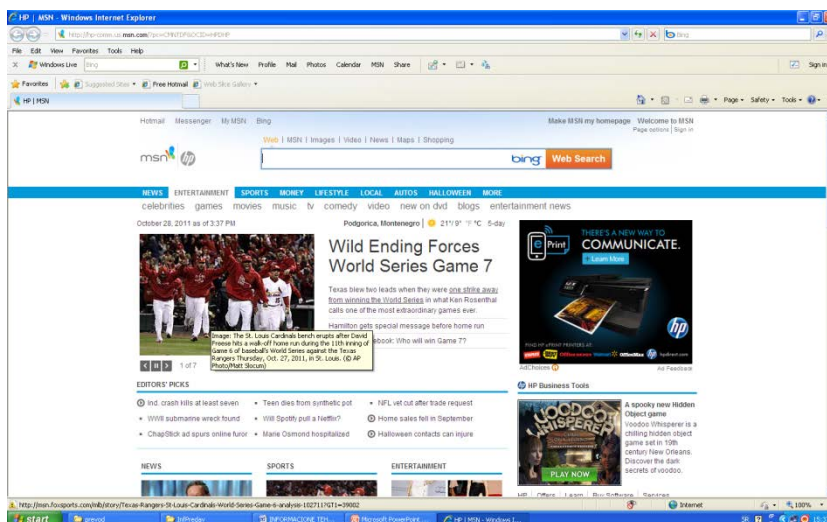
Tabela 1.3.: Integrirani paketi

Programs	Microsoft Office	Lotus SmartSuite	Corel WordPerfect Office	Sun StarOffice
Word Procesor	Word	WordPro	WordPerfekt	StarWriter
Spreadsheet	Excel	1-2-3	Quattro Pro	StarCalc
Presentation Graphics	PowerPoint	Freelance	Presentations	StarImpress
Database Manager	Access	Approach	Paradox	StarBase
PIM	Outlook	Organizer	Corel Central	StarSchedule

Naravno, integrisanje softverskih paketa ima i svoje nedostatke. Osnovni nedostatak jeste taj što zauzimaju mnogo prostora na disku (preko 150 MB), a mnogi korisnici neke programe i prednosti integrisanog paketa nikada ne upotrebljavaju. Nije zanemarljiva ni cijena koja se kreće u rang u od nekoliko stotina dolara (preko 700\$ za kompletne verzije).

Web browseri. – Gotovo najvažnija softverska komponenta za mnoge korisnike danas jeste jednostavni i namjenski ograničeni, ali moćni alat web browser. Browser programi, kao što su Netscape Navigator i Microsoft Explorer (slika 1.54.), predstavljaju ključni softverski interfejs koji koristimo za *point and click* putovanje po hiperlinkovanim veb-resursima na internetu, kao i na intranet i ekstranet mrežama.

Iako su proizvedeni radi surfovanja po internetu, može se reći da danas browseri postaju univerzalna softverska platforma za pretraživanje informacija, za elektronsku poštu, transfer multimedijalnih fajlova, učešće u diskusionim grupama i za još mnogo drugih internet, intranet i ekstranet primjena. Tako, ukoliko želite da gledate video film, telefonirate, downloadujete neki softver, održite video konferenciju sa poslovnim partnerima ili svojim zaposlenima, da provjerite vaš e-mail, pregledate cjenovnik vaših proizvoda ili plan rada projektnog tima, ili da upotrijebite neki program koji podržava upravljanje i operativnost u organizaciji (npr. da pregledate izvještaj o prodaji, da pošaljete narudžbenicu dobavljaču itd.) – vi ćete to uraditi pomoću browsera. Dakle, browser je univerzalni klijent, tj. univerzalni softver koji se instalira na klijentima intranet mreže.

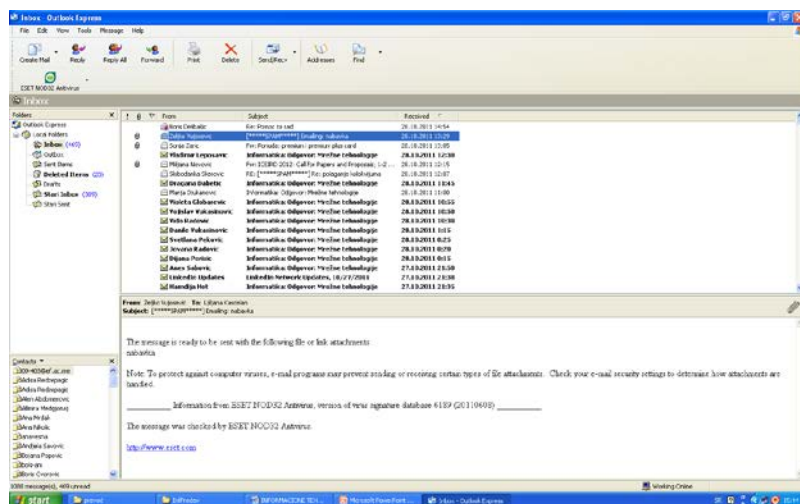


Slika 1.54. Web browser MS Explorer

E-mail procesori. – Prva stvar koju mnogi zaposleni širom svijeta svaki radni dan urade, jeste da provjere svoj e-mail. Elektronska pošta je promijenila način na koji ljudi rade i komuniciraju. Milioni korisnika sada koriste e-mail softver da bi ostvarili komunikaciju slanjem i primanjem elektronskih poruka preko interneta ili njihovih organizacijskih intranet i ektranet mreža.

E-mail se čuva na mrežnim serverima sve do trenutka dok ga korisnik ne primi. Korisnik, kada bude htio, može pročitati njegov sadržaj na ekranu svoje radne stanice. Dakle, za samo nekoliko minuta truda (plus nekoliko mikrosekundi ili minuta za prenos poruke) poruka za jednog ili više korisnika može biti sastavljena, poslana i primljena.

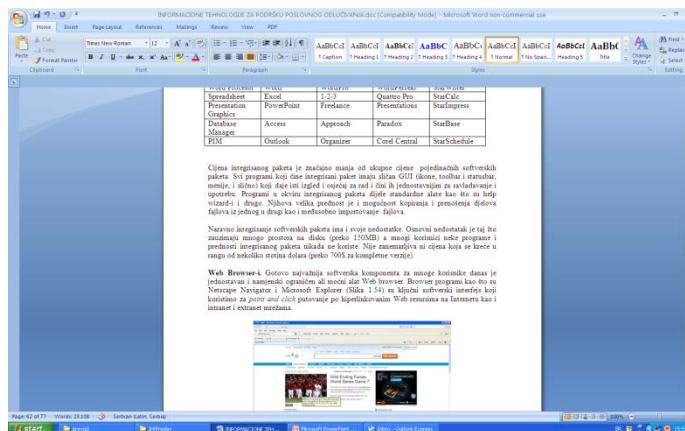
E-mail softver predstavlja komponentu nekog integrisanog paketa (kao na primjer Outlook Express, slika 1.55.) ili web browsera. Originalne verzije ovog softvera imaju mogućnost upućivanja poruke za više korisnika istovremeno (na osnovu *predefined mailing lists*), zatim obezbjeđuju sigurnosne zaštite u vidu passworda, automatsko forwardovanje poruka i drugo. Oni takođe omogućavaju čuvanje poruka u folderima, dodavanje atačmenta fajlu poruke, editovanje i slanje multimedijalnih poruka. Na kraju e-mail softver automatski filtrira i sortira pristigle poruke, raspoređujući ih u odgovarajuće mailboxove i foldere.



Slika 1.55. E-mail procesor MS Internet Explorer

Tekst procesori i desktop izdavaštvo (publikovanje). – Tekst procesori omogućavaju kompjutersko kreiranje, editovanje, reviziju i štampanje dokumenata (kao što su pisma, zapisnici, izvještaji itd.) elektronskim procesiranjem tekstualnih podataka (riječi, rečenica i paragrafa). Najpoznatiji tekst procesori, kao što su Microsoft Word (slika 1.56.), Lotus WordPro, i Corel WordPerfect, zahvaljujući svojim mogućnostima desktop izdavaštva, omogućavaju generisanje i štampanje različitih vrsta dokumenata u atraktivnom formatu. Ovi paketi takođe omogućavaju konvertovanje svih dokumenata u HTML format i njihovo publikovanje u vidu veb-strana na organizacijskom intranetu ili na webu.

Tekst procesori takođe obezbjeđuju neke napredne servise kao što je spell cheking, koji koriguje greške u spelovanju kod engleskog jezika i predlaže najbolji izbor riječi za izražavanje neke ideje. Oni takođe identifikuju i ispravljaju gramatičke greške i daju sugestije za poboljšanje stila pisanja. S obzirom na to da posjeduju mogućnost konvertovanja dokumenata u HTML format, ovi paketi se mogu koristiti i za kreiranje veb-strana za internet ili intranet veb-sajt.

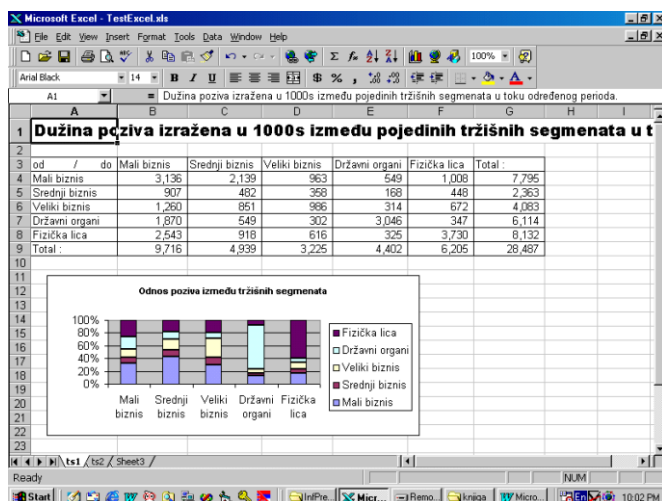


Slika 1.56. Tekst procesor MS Word

Softver za desktop publikovanje (DTP – desktop publishing) omogućava korisnicima i njihovim organizacijama da sami kreiraju štampane materijale za profesionalno publikovanje. Dakle oni mogu sami dizajnirati i štampati sopstvene časopise, brošure, uputstva i knjige, koristeći različite stilove, grafike, fotografije i boje. Najpoznatiji DTP softveri su Adobe Page Maker i QuarkXPress. Obično se tekstualni i grafički materijali generišu pomoću tekst procesora i grafičkih paketa i importuju se kao grafički i tekstualni fajlovi ili se pak skeniraju sa odštampanih materijala. Dakle, suština desktop publikovanja jeste dizajniranje stranice (*page makup* ili *page composition*). Pomoću DTP softvera dizajnira se format strane, a tekstualni materijali i ilustracije se uklapaju u ovaj format. Prebacivanje teksta u drugu kolonu ili stranicu obavlja se automatski. DTP pomaže pri određivanju veličine i mjesta ilustracija, headingsa i drugog.

Spreadsheet programi. – Programi za obradu elektronskih tabela (spreadsheet) kao što su Microsoft Excel, Lotus 1-2-3 i Corel QuattroPro koriste se za poslovne analize, planiranje i modelovanje. Oni omogućavaju kreiranje elektronske tabele (podaci raspoređeni po vrstama i kolonama) koja može biti sačuvana na korisnikovom PC računaru ili serveru ili konvertovana u HTML format i sačuvana kao veb-strana na organizacijskom intranetu ili na webu. Kreiranje radne tabele podrazumijeva kako njeno formatiranje tako i upotrebu formula i funkcija za izračunavanja. Kao odgovor na korisnikov unos ili editovanje podataka, program na osnovu formule ili funkcije definisane od strane korisnika izvršava potrebna

izračunavanja i odmah prikazuje rezultat. Većina ovih paketa omogućava i grafičko prezentiranje radnih tabela (slika 1.57.).



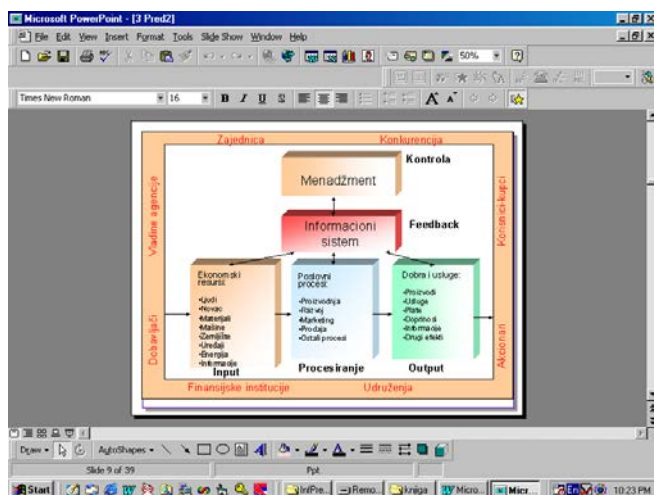
Slika 1.57. Program za obradu elektronskih tabela MS Excel

Tako na primjer, pomoću ovih programa može se kreirati radna tabela za analizu prošlih i sadašnjih performansi i efekata reklamiranja. Sada ovaj program postaje alat za podršku odlučivanju jer omogućava what-if analizu reklamiranja. Tako se na primjer može analizirati šta će se desiti sa tržišnim učešćem ukoliko se troškovi reklamiranja povećaju za 10%. Korisnik će kod ovakve analize promijeniti formulu za troškove reklamiranja. Softver će automatski izvršiti ponovna izračunavanja za sve tabele i grafikone koji se tiču tržišnog učešća, a koje zavise od troškova reklamiranja. Efekti odluka u oblasti reklamiranja i njihov uticaj na tržišno učešće preduzeća sada će biti jasnije uočljivi. Ovo zapažanje se može podijeliti sa ostatkom radnog marketinškog tima tako što se na intranet veb-sajtu postavi hiperlink na ovu radnu tabelu.

Programi za grafičke i multimedijalne prezentacije. – Ovi programi omogućavaju konvertovanje numeričkih podataka u grafičke prikaze, kao što su linijski i kružni grafikoni, grafikoni sa pravougaonicima i razni drugi tipovi grafikona. Većina ovih programa omogućava i pripremu multimedijalnih prezentacija sa tekstom, grafikonima, fotografijama, audio i video animacijama, kao i njihovo publikovanje na webu (konvertovanje u

HTML format). Grafički i multimedijalni prikazi nisu samo lakši za shvatanje od numeričkih podataka nego i bolje ističu ključne tačke, strateške razlike i važne trendove u vezi sa podacima. Grafičke prezentacije su znatno efikasnije od tabelarnih prezentacija numeričkih podataka za izvještavanja u marketinškim medijima, menadžmentu i ostalim poslovnim domenima.

Poznati paketi za grafičke prezentacije jesu Microsoft PowerPoint (slika 1.58.), Lotus Freelance i Corel Presentation. Iako većina paketa omogućava automatski generisani slide show sa multimedijalnim prikazima, paketi za grafičke prezentacije imaju mnogo veće mogućnosti. Oni omogućavaju korišćenje predizajniranih obrazaca (template) za poslovne prezentacije, pripremu i editovanje crtica i napomena vezanih za prezentaciju i upravljanje multimedijalnim fajlovima. Najveća prednost ovih paketa jeste mogućnost transfera multimedijalnih prezentacija u HTML format za intranet i internet veb-sajtove.



Slika 1.58. Program za grafičke prezentacije MS PowerPoint

Već smo više puta pominjali HTML, hypertext i hypermedia tehnologije. Hypertext i hypermedia su **softverske tehnologije za multimedijalne prezentacije**. Prema definiciji, hypertext sadrži samo tekst i ograničenu količinu grafike. Hypermedia je elektronski dokument koji sadrži različite formate, uključujući tekst, grafikone, video i audio zapise i drugo. Ključne

teme u hypertext i hypermedia dokumentima jesu softverski linkovi (hiperlinkovi) na neke druge povezane displeje.

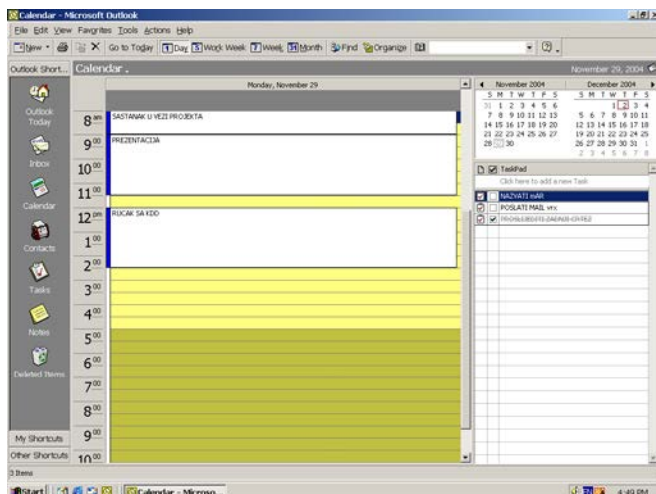
Ovi linkovi su obično podvučeni tekstovi mada mogu biti i slike, grafikoni ili neki drugi format podatka. Klikom miša na ove podvučene termine automatski se prikazuje novi sadržaj povezan sa tim terminom. Kada se završi sa tim dijelom prezentacije, uvijek se možemo vratiti nazad na stranu hypermedia dokumenta koju smo razmatrali prije klika.

Hypertext i hypermedia dokumenta kreiraju se pomoću softverskih paketa koji se oslanjaju na specijalne programske jezike Java i HTML (Hypertext Markup Language). Obično se kod ovih tehnologija kreiraju hiperlinkovi na druge djelove dokumenta ili na druge dokumente i multimedijalne fajlove. Hypertext i hypermedia dokumenta mogu se koristiti i za pretraživanje multimedijalnih baza podataka, slijedeći lanac hiperlinkova na različite multimedijalne fajlove. Veb-sajtovi na internetu predstavljaju popularni primjer ovih tehnologija. Dakle, može se reći da upotreba hypertext i hypermedia softvera pomoću web browsera ili nekog drugog programa predstavlja okruženje za on-line interaktivne multimedijalne prezentacije.

PIM (Personal Information Manager) je popularni softverski paket koji omogućava bolju organizovanost i produktivnost korisnika a najpopularnija mu je primjena kod PDA uređaja. PIM softverski paketi kao što su Microsoft Outlook (Slika 1.59) i Lotus Organizer pomažu korisnicima prilikom čuvanja, organizovanja i pretraživanja informacija o korisnicima, klijentima i obavezama, planiranja i upravljanja sastancima, sjednicama i radnim zadacima. PIM paket organizuje unesene podatke, pretražuje i prikazuje informacije u različitoj formi, zavisno od stila i strukture PIM-a i traženih informacija. Na primjer, informacije mogu biti prikazane u obliku elektronskog kalendara ili kao lista termina za sastanke i sjednice, zatim u obliku vremenske tabele za neki projekat ili kao prikaz ključnih činjenica i finansijskih podataka o korisnicima, klijentima i drugo.

PIM programi se prodaju kao nezavisni programi ili su uključeni u integrisane pakete i razlikuju se u svom stilu, strukturi i osobinama. Na primjer, Lotus Organizer koristi notebok format sa tabovima, dok Microsoft Outlook organizuje podatke o ljudima u vidu neprekidne A–Z liste. Većina PIM paketa posebno održava kontakt listu klijenata, korisnika i drugo. Planiranje sastanaka, sjednica i radnih zadataka takođe predstavljaju važne primjene PIM paketa. Uključivanje mogućnosti pristupa webu kao i

moгуćnosti korišćenja e-mail u PIM programe je trenutno aktuelni trend. Novi PIM-ovi dakle podržavaju timski rad i saradnju



Slika 1.59. PIM softver MS Outlook

korišćenjem interneta i e-maila, tj. dijeljenjem informacija kao što su kontakt liste, liste radnih obaveza i planovi sa ostalim umreženim PIM korisnicima – članovima tima.

Groupware programi jesu programi za saradnju koji pomažu radnim grupama i timovima da rade zajedno i izvršavaju grupne zadatke. Groupware pripada kategoriji softvera opšte namjene sa veoma brzim porastom koja kombinuje različite softverske tehnike i funkcije sa ciljem unapređenja saradnje. Na primjer, groupware programi, kao što su Lotus Notes, Microsoft Exchange i drugi, podržavaju saradnju pomoću e-maila, diskusionih grupa, rasporeda, audio i video konferencija, konferencija putem podataka, baza podataka i slično.

Groupware proizvodi neprestano se mijenjaju u težnji za boljom i efikasnijom saradnjom. Danas se ovi programi dizajniraju tako da koriste internet i organizacijske intranet i ekstranet mreže, omogućavajući na taj način saradnju na globalnom nivou, odnosno saradnju sa virtuelnim timovima lociranim bilo gdje u svijetu. Tako na primjer član tima, ma gdje se trenutno nalazio, može koristiti internet za e-mail, za diskusione forume o projektu i za dizajniranje i publikovanje zajedničkih veb-strana. On takođe

može koristiti organizacijski intranet za publikovanje projektnih novosti, izvještaja o napredovanju, kao i za zajednički rad na dokumentima koji se nalaze na veb-serveru.

1.5.3. SISTEMSKI SOFTVER

Sistemska softver sastoji se od programa koji upravljaju i podržavaju kompjuterski sistem i njegovo procesiranje informacija. Slika 1.47. prikazuje da se sistemska softver može grupisati u dvije kategorije, koje obuhvataju programe za podršku sistema i programe za razvoj sistema.

Programi za podršku sistema upravljaju hardverom, softverom, mrežom i resursima podataka prilikom izvršavanja korisničkih zadataka. Najvažniji programi iz ove kategorije jesu operativni sistemi, programi za upravljanje mrežom, programi za upravljanje bazama podataka i uslužni sistemska programi.

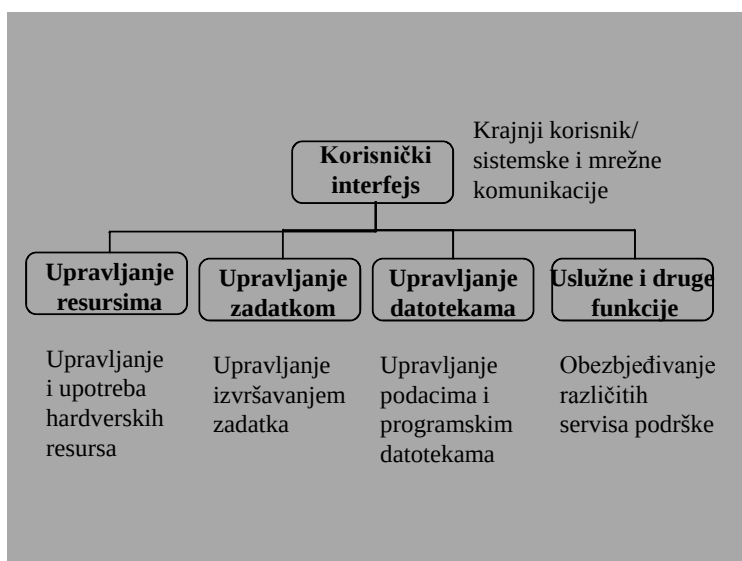
Operativni sistemi. – Najvažniji sistemska softverski paket za računar jeste operativni sistem. Operativni sistem predstavlja integrisani sistem programa koji upravljaju operacijama CPU, kontrolišu ulaz/izlaz i memorijske resurse, kao i aktivnosti računarskog sistema i obezbjeđuju veliki broj servisa podrške za izvršavanje korisničkih aplikativnih programa.

Osnovni zadatak operativnog sistema jeste maksimiziranje produktivnosti kompjuterskog sistema uz upravljanje njime na najefikasniji mogući način. Ovi programi minimiziraju potrebu korisnikovih intervencija prilikom procesiranja. Oni omogućavaju aplikativnim programima da izvršavaju standardne operacije kao što su pristup mreži, unos podataka, čuvanje i pretraživanje fajlova i štampanje i prikazivanje izlaza. Operativni sistem se nalazi na sekundarnoj memoriji i mora se učitati u radnu memoriju i aktivirati prije nego počne izvršavanje korisničkih aplikacija.

Operativni sistem obavlja pet osnovnih funkcija u računarskom sistemu, a to su: obezbjeđivanje korisničkog interfejsa, upravljanje resursima, upravljanje zadacima (procesima), upravljanje datotekama, obezbjeđivanje uslužnih servisa i servisa podrške (slika 1.60.).

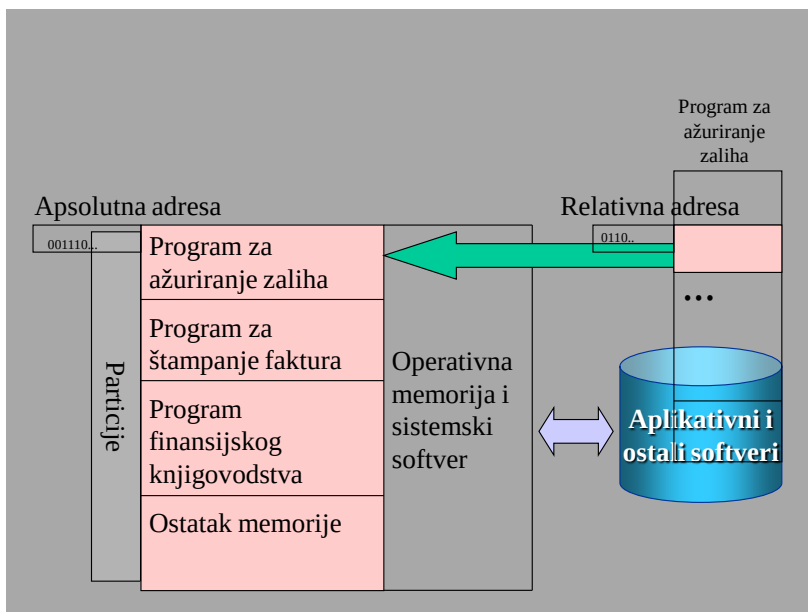
Korisnički interfejs je dio operativnog sistema koji omogućava komunikaciju korisnika sa sistemom u smislu pokretanja programa, pristupa datotekama i izvršavanja raznih drugih zadataka. Tri osnovna tipa

korisničkog interfejsa jesu komandni, interfejs preko sistema menija i grafički interfejs. Osnovni trendovi kod korisničkih interfejsa jesu izbjegavanje pamćenja i unošenja komandi ili čak i izbora opcija menija. Umjesto toga većina operativnih sistema obezbjeđuje grafički korisnički interfejs (GUI – Graphical User Interface), koji jednostavan za upotrebu, a koristi ikone, barove, komandnu dugmad, boksove i drugo. GUI se zasnivaju na ulaznim uređajima (poput miša) koji omogućavaju selekciju objekata pokazivačem ili na dodir.



Slika 1.60. Funkcije operativnog sistema

Upravljanje resursima podrazumijeva upravljanje hardverskim i mrežnim resursima, uključujući CPU, radnu memoriju, sekundarnu memoriju, telekomunikacione procesore i input/output periferije. Na primjer, programi za upravljanje memorijom čuvaju trag o tome gdje su podaci i programi smješteni u memoriji (adrese). Oni dijele memoriju na particije i omogućavaju razmjenu djelova programa i podataka između memorije i magnetnih diskova ili nekih drugih uređaja sekundarne memorije. Drugim riječima, realizuju koncept **virtuelne memorije** koji omogućava sistemu da izvršava programe i obrađuje podatke veće nego što su realni kapaciteti osnovne (radne) memorije (slika 1.61.).



Slika 1.61. Realizacija koncepta virtuelne memorije

Upravljanje datotekama omogućava kreiranje, brisanje i pristup datotekama podataka i programa. Programi za upravljanje datotekama čuvaju fizičke lokacije (adrese) datoteka na magnetnim diskovima ili nekim drugim uređajima spoljašnje memorije. Operativni sistem, dakle, održava direktorijume informacija o lokacijama i karakteristikama datoteka sačuvanih na uređajima sekundarne memorije.

Upravljanje zadacima (procesima) podrazumijeva upravljanje izvršavanjem zadataka (procesa) korisničkih aplikacija na centralnom procesoru. Operativni sistemi svakom procesu dodjeljuju mali trenutak (slot) CPU vremena i prekidaju CPU operacije da bi ga dodijelili drugom zadatku. Na ovaj način više zadataka se izvršava istovremeno, i to se označava terminom **multitasking**. Ukoliko CPU izvršava zadatke više programa istovremeno, onda je to **multiprogramiranje**, a ukoliko se istovremeno izvršavaju programi od strane više korisnika, riječ je o **timesharingu**. Efikasnost multitasking-a zavisi od snage procesiranja CPU, virtuelne memorije i mogućnosti operativnog sistema.

Većina operativnih sistema ima mogućnosti multitasking-a. Tako korisnik može istovremeno obavljati dvije ili više operacija jednim programom (na

primjer kucati tekst nekog dokumenta dok se neki drugi štampa) ili raditi sa više aplikacija, odnosno programa (na primjer kucati tekst i pretraživati web). Multitasking na mikrokompjuterima omogućen je razvojem moćnijih mikroprocesora sa sposobnošću adresiranja memorije većeg kapaciteta. Ovo omogućava operativnom sistemu da podijeli radnu memoriju na particije u kojima se mogu učitavati i izvršavati djelovi različitih aplikativnih programa. Broj programa koji se mogu konkurentno izvršavati zavisi od kapaciteta radne memorije, kao i od broja zahtjeva za procesiranje. Ukoliko je CPU preopterećen, tj. ima prevelik broj zahtjeva, može doći do sporog odziva, odnosno dugog čekanja. Ako je kapacitet memorije i snaga CPU adekvatna, koncept multitasking-a omogućava korisniku da nesmetano prelazi sa jedne aplikacije na drugu, da koristi datoteke u više aplikacija istovremeno i da izvršava zadatke u pozadini, kao što su dugotrajno štampanje, matematička izračunavanja ili telekomunikacione sesije.

Prvi operativni sistem mikroračunara bio je MS-DOS (Microsoft Disk Operating System). To je bio jednokorisnički i single-tasking sistem sa komandnim korisničkim interfejsom. U kombinaciji sa Microsoft Windows-om imao je GUI i limitirane multitasking mogućnosti. Microsoft je 1995. godine zamijenio ovu DOS/Windows verziju operativnim sistemom Windows 95. Ovaj sistem je imao GUI, stvarni multitasking, mogućnost upravljanja mrežom, multimedijalne i druge opcije. Kasnije verzije ovog operativnog sistema jesu Windows 98, Windows 2000 i Windows XP.

Svoju ediciju mrežnih operativnih sistema Microsoft je započeo 1995. godine sa Windows-om NT (New Technology). Ovo je bio moćni multitasking, višekorisnički sistem koji je instaliran na mnogim mrežnim serverima da upravlja klijent/server mrežama, ali i na PC računarima sa zahtjevima visokih performansi. Microsoft je svoje proizvode 98 i NT spojio u novom proizvodu Windows 2000 koji predstavlja moćan operativni sistem za mrežne i veb-servere kao i za radne stanice, tj. PC i laptop računare (*Professional* verzija).

Kod mainframe i midrange kompjutera, ali i kod mikrokompjutera popularni su multitasking, višekorisnički mrežni operativni sistemi UNIX i Linux. I pored dosta visoke cijene, UNIX je najčešći izbor za veb-servere. Linux je jeftinija verzija UNIX operativnog sistema, koja se i danas može nabaviti besplatno ili po veoma niskoj cijeni preko interneta.

Programi za upravljanje mrežom. – Danas se informacioni sistemi u potpunosti oslanjaju na internet, intranet i ekstranet, LAN i druge telekomunikacione mreže da bi povezali korisničke radne stanice, mrežne servere i druge kompjuterske sisteme. Ovo zahtijeva razne vrste sistemskog softvera za upravljanje mrežom, uključujući mrežne operativne sisteme, monitore mrežnih performansi, telekomunikacione monitore i drugo. Ovi programi se nalaze na mrežnim serverima, ali i na drugim kompjuterima i uređajima u mreži (radnim stanicama, telekomunikacionim procesorima itd.). Oni kontrolišu input/output aktivnosti radnih stanica i terminala i dodjeljuju prioritete njihovim komunikacionim zahtjevima, detektuju i koriguju greške prilikom prenosa podataka i drugo.

Osim pomenutih mrežnih operativnih sistema, poznati su IBM-ov mrežni operativni sistem Novell NetWare za komplekse međusobno povezanih LAN mreža, telekomunikacioni monitor CICS (Customer Identification and Control System) za WAN mreže i NetView za mainframe i midrange bazirane mreže.

Sistemi za upravljanje bazama podataka (DBMS – Data Base Management System). – DBMS predstavljaju veoma važan sistemski softverski paket koji omogućava razvoj, korišćenje i upravljanje bazama podataka. DBMS pomaže organizaciji da efikasno kreira i koristi svoju integrisanu kolekciju slogova i datoteka, poznatu kao baza podataka, ali i omogućava različitim korisničkim aplikacijama da pristupaju istoj bazi podataka. Umjesto da koristi korisničku aplikaciju za ekstrakciju podataka, korisnik može sam postavljati jednostavne upite pomoću upitnog jezika koji podržava DBMS. Većina DBMS paketa ima upitne jezike, tj. jezike četvrte generacije (4GL – fourth generation language), kao i druge mogućnosti za razvoj aplikacija.

Poznati DBMS paketi su: Oracle 9i, IBM-ov DB2, kao i MS Access, Lotus Approach i Corel Paradox za mikrokompjutere.

Ostali programi za podršku sistemu. – Uslužni programi samostalno ili kao dio operativnog sistema omogućavaju različite upravljačke i fajl-konverzione funkcije, kao što su backup i recovery podataka, zaštita od virusa, kompresija, kopiranje, sortiranje, spajanje i defragmentaciju fajlova i drugo. Programi za praćenje i zaštitu performansi prate performanse i stepen iskorišćenosti računarskog sistema, tj. obezbjeđuju optimalnu efikasnost sistema i prate i zapisuju neautorizovane pristupe računarskom sistemu, i o

tome porukom obavještavaju korisnika. Aplikativni server programi obezbjeđuju međusloj između operativnog sistema i korisničkih aplikativnih programa. Posebno se koriste kod web baziranih e-business i e-commerce aplikacija, jer se one pomoću ovog međusloja izvršavaju mnogo brže i efikasnije. Poznati aplikativni server programi jesu Web Logic firme BEA i IBM-ov WebSphere, koji rade sa Windows i Unix operativnim sistemima.

Programi za razvoj sistema omogućavaju izradu programa i njihovu pripremu za kompjutersku obradu, tj. izvršavanje. Glavni alati za razvoj softvera jesu prevodioci programskih jezika, programski editori i CASE alati.

Programski jezici omogućavaju programeru da skupom unaprijed utvrđenih instrukcija napiše kompjuterski program. Postoji više različitih programskih jezika, od kojih svaki ima svoj jedinstveni vokabular (skup instrukcija-naredbi), gramatiku i upotrebu. Posmatračemo ih kroz nekoliko generacija.

Mašinski jezici ili **jezici prve generacije** jesu programski jezici najnižeg nivoa čije programske instrukcije predstavljaju binarni kodovi zavisni od mašine na kojoj se izvršavaju. Programiranje na mašinskom jeziku je veoma kompleksno jer programer mora imati detaljna znanja o internim operacijama CPU-a koji koristi. Za izvršavanje jednostavnih zadataka moraju se pisati dugačke serije detaljnih instrukcija. Programiranje u mašinskom jeziku zahtijeva specifikovanje memorijskih adresa za svaku instrukciju ili podatak. Ovo ga čini jako teškim i pogodnim za greške. Program na mašinskom jeziku za sabiranje dva broja (za jednu specifičnu mašinu) prikazan je na slici 1.62.

Asemblerski jezici ili **jezici druge generacije** razvijeni su da bi smanjili teškoće pri pisanju mašinskih programa. Programi pisani asemblerskim jezikom moraju se prevoditi na mašinski kod. Ovo omogućavaju programi prevodioci poznati pod nazivom asembleri. Ovi jezici su poznati kao simbolički jezici jer se kodovi operacija i memorijskih lokacija predstavljaju simbolima. Na primjer izračunavanje $X=Y+Z$ u asemblerskom jeziku može biti realizovano kao na slici 1.62.

Viši programski jezici ili **jezici treće generacije** koriste instrukcije, tj. naredbe bazirane na engleskom jeziku i aritmetičkim izrazima. Kada se prevede na mašinski jezik pomoću programa prevodioca, svaka naredba

višeg programskog jezika predstavlja makroinstrukciju, koja generiše više mašinskih instrukcija. Programi prevodioci za više programske jezike jesu kompajleri i interpreteri. Sintaksa (skup rezervisanih riječi i znakova i gramatičkih pravila) kao i semantika (značenja) naredbi viših programskih jezika ne zavise od internog koda pojedinačnog kompjutera. Na primjer, izračunavanje $X = Y + Z$ programirano u višem programskom jeziku BASIC izgleda kao na slici 1.62.

Četiri nivoa programskih jezika	
Mašinski jezik Koristi binarno kodirane instrukcije 1010 11001 1011 11010 1100 11011	Viši programski jezici Koriste kratke naredbe ili aritmetičku notaciju BASIC : $X=Y+Z$
Asemblerski jezik Koristi simbolički kodirane instrukcije LOD Y ADD Z STR X	4GL Koriste prirodne i neproceduralne naredbe SUM THE FOLLOWING NUMBERS

Slika 1.62. Četiri nivoa programskih jezika

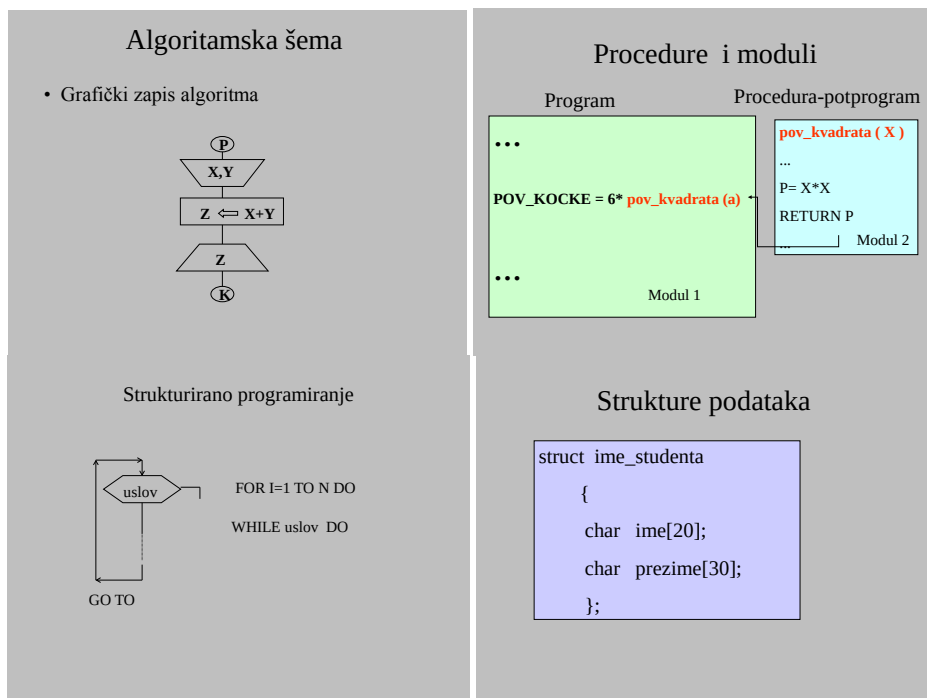
Viši programski jezici su lakši za učenje i programiranje od asemblerskih, jer imaju manje rigidna pravila, oblik i sintaksu. Međutim, programi na višem programskom jeziku obično su manje efikasni od asemblerskih i zahtijevaju više kompjuterskog vremena zbog prevođenja u mašinski kod. Pošto je većina viših programskih jezika mašinski nezavisna, ako instaliramo novi kompjuter, programi se ne moraju mijenjati, a programeri ne moraju učiti novi programski jezik za svaki novi tip kompjutera. Tabela 1.4. prikazuje neke od glavnih viših programskih jezika koji se danas još uvijek koriste.

Tabela 1.4. Viši programski jezici

Viši programski jezici
Ada: Dobio je naziv po imenu Auguste Ade Byron, koja se smatra prvim svjetskim programerom. Razvijen je od strane američke vlade kao standard viših programskih jezika koji treba da zamijeni COBOL i FORTRAN.
BASIC (Beginner's all-Purpose Symbolic Instruction Code): Jednostavni proceduralni jezik za programiranje od strane korisnika.
C: Strukturirani jezik srednjeg nivoa razvijen kao dio UNIX operativnog sistema. Sličan je mašinski zavisnom asemblerskom jeziku po efikasnosti, ali je mašinski nezavisan i ima sve mogućnosti viših programskih jezika.
COBOL (COmmon Business Oriented Language): Programski jezik čija se sintaksa zasniva na engleskom jeziku. Ima veoma široku primjenu u programiranju poslovnih aplikacija
FORTRAN (FORmula TRANslation): Viši programski jezik dizajniran za naučne i inženjerske primjene.
PASCAL: Dobio je naziv po Blaiseu Pascalu. Razvijen je specijalno da uključi koncepte strukturnog programiranja i struktura podataka.

Za više programske jezike vezani su termini algoritamski, proceduralni, strukturirani i modularni jezici. Algoritamski – znači da su to jezici zasnovani na algoritamskom programiranju. Algoritam predstavlja skup pravila (algoritamskih koraka ulaza, obrade, uslova i izlaza) formulisanih u cilju rješavanja nekog problema. Grafički zapis algoritma jeste algoritamska šema (slika 1.63. – gore lijevo). Većina viših programskih jezika podržava algoritamsko programiranje. Proceduralni jezici omogućavaju pisanje programa u vidu skupa procedura (potprograma). Djelovi programa koji se mogu nezavisno izvršiti čine proceduru ili potprogram. Modularno programiranje podrazumijeva mogućnost pisanja i izvršavanja djelova programa u različitim modulima i koristi se uglavnom kod proceduralnih jezika (slika 1.63. – gore desno). Za razliku od jezika prve i druge generacije, koji su sekvencijalni, većina viših programskih jezika je proceduralna. Strukturirano programiranje podrazumijeva upotrebu specijalnih instrukcija za kontrolisanje strukture programa kao što su *if-then-else* naredbe, *for*, *while* i *do* petlje za cikluse i drugo, kako bi se izbjegla *go-to* naredba karakteristična za sekvencijalne programe prve i druge generacije (slika 1.63. – dolje lijevo). Neki viši programski jezici

osim osnovnih tipova podataka koriste korisnički definisane tipove, tzv. strukture podataka (slika 1.63. – dolje desno).



Slika 1.63. Osobine viših programskih jezika: algoritamske šeme, procedure i moduli, strukturirano programiranje, podržavanje strukture podataka

Jezici četvrte generacije (4GL jezici) obuhvataju različite programske jezike koji su neproceduralni, tj. koji su konverzacioni, što znači da korisnik i programer specifikuje samo šta želi da uradi, a ne i kojom procedurom (nizom instrukcija) to treba uraditi. Istraživanja u oblasti vještačke inteligencije dovode do pojave prirodnih jezika sličnih engleskom ili nekom drugom jeziku. Na primjer, kod INTELLECT programskog jezika možemo koristiti naredbu oblika “*What are the average exam scores in MIS 200?*”.

Lakoća upotrebe 4GL jezika, sa druge strane, dovodi do smanjenja njihove efikasnosti i fleksibilnosti. Unaprijed definisani formati i procedure ovih jezika često predstavljaju prepreku preko koje programerima nije lako preći. Isto tako, mašinski kod generisan prevođenjem ovih jezika često je manje efikasan (u terminima brzine izvršavanja i količine zahtijevanog memorijskog prostora) nego programi napisani višim programskim

jezicima. Zato aplikacije napisane u 4GL jezicima ne daju dobro vrijeme odziva prilikom procesiranja velike količine real-time transakcija i korisničkih zahtjeva. Programski jezici 4GL dobar su izbor kod poslovnih aplikacija koje ne procesiraju veliku količinu transakcija.

Ovaj problem se često rješava kombinovanjem 4GL jezika sa drugim programskim jezicima, čime se postiže fleksibilnost i efikasnost, a programeri se oslobađaju napornog posla pisanja procedura za neke operacije. Ovo je najčešće slučaj kod aplikacija za baze podataka pisane 4GL jezikom koji je sastavni dio DBMS-a, i kod kojeg se upitni jezik (npr. SQL), kombinuje sa strukturnim naredbama nekog višeg programskog jezika. Tako imamo 4GL/Oracle, dBase, SQLServer i druge.

Objektno orijentisani jezici, kao što su Visual Basic, C++ i Java, danas postaju glavna platforma za razvoj softvera. Ukratko, za razliku od većine programskih jezika koji odvajaju podatke od procedura i akcija koje se na njima izvršavaju, objektni jezici ih objedinjuju u okviru objekata. Dakle, objekat se sastoji od podataka i operacija koje mogu biti izvršene nad podacima. Na primjer, jedan objekat može biti skup podataka o štednom računu klijenta banke zajedno sa operacijom za izračunavanje interesa. Isto tako, objekat može biti video displej prozor plus displej akcija koja ga otvara.

Kod proceduralnih jezika program se sastoji od procedura koje izvršavaju različite operacije nad podacima. Kod objektnih jezika objekti aktiviraju druge objekte da izvršavaju operacije nad sobom. Na primjer, da bi se otvorio prozor na kompjuterskom video displeju, početni meni objekat šalje poruku objektu prozora da se otvori i prozor se pojavljuje na ekranu. Ovo je moguće jer objekat prozor sadrži programski kod za sopstveno otvaranje.

Objektni jezici predstavljaju jednostavnu i efikasnu platformu za programiranje grafičkih interfejsa zahtijevanih od većine aplikacija. Mogućnost višestrukog korišćenja jednom definisanog objekta jeste glavna prednost objektnog programiranja. Tako na primjer, programer može konstruisati korisnički interfejs za novi program, sastavljajući ga od već postojećih objekata kao što su prozori, boksovi, komandna dugmad, ikone i drugo. Većina objektno orijentisanih paketa za programiranje obezbjeđuje GUI, tj. *point and click* i *drag and drop* vizuelno sastavljanje objekata ili tzv. vizuelno programsko okruženje (Visual Basic, Visual C++ i drugi).

HTML, XML i Java jesu programski jezici za izradu multimedijalnih veb-strana, veb-sajtova i veb baziranih aplikacija.

HTML (Hypertext Markup Language) jeste jezik za kreiranje hypertext i hypermedia dokumenata. On određuje strukturu i format ovih dokumenata putem markera (slika 1.64.). HTML insertuje kontrolne kodove u dokumenat kod tačaka koje su specifikovane kao hiperlinkovi. Kao što smo ranije rekli, većina programskih paketa opšte namjene automatski konvertuje dokumenta u HTML format. Takođe postoje specijalni programi za veb-publikovanje kao što su Microsoft FrontPage i Lotus FastSite, koji omogućavaju dizajn i kreiranje veb-strana bez formalnog HTML programiranja.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<HTML>
  <HEAD>
    <TITLE>A study of population dynamics</TITLE>
    <STYLE type="text/css">
      BODY { background: white; color: black}
      A:link { color: red }
      A:visited { color: maroon }
      A:active { color: fuchsia }
    </STYLE>
  </HEAD>
  <BODY>
    ... document body...
  </BODY>
</HTML>
```

Slika 1.64. Primjer HTML dokumenta

XML (eXtensible Markup Language) nije jezik za formatiranje veb-strana, već on opisuje njihov sadržaj primjenom identifikacionih tagova ili kontekstualnih labela za podatke na veb-strani. Na primjer, veb-strana putničke agencije sa nazivima avionskih linija i vremenima letova može

koristiti XML tagove *ime linije* i *vrijeme leta* kako bi se označilo svako vrijeme leta za neku liniju na toj strani. Ili, podaci o proizvodima na zalihama koji se nalaze na jednoj veb-strani mogu biti označeni tagovima *vrsta*, *cijena* i *veličina*. Slika 1.65. prikazuje veb-stranu sa sadnicama koje posjeduje jedna plantaža. Na strani se pojavljuju uobičajeni i botanički naziv sadnice, zona u kojoj uspijeva sadnica, količina svjetla koja joj je potrebna i šifre dostupnih proizvoda. Ovi sadržaji označeni su crvenim tagovima sa lijeve i desne strane. Ovakvom klasifikacijom podataka na veb-strani, XML čini veb-sajtove lakšim za pretraživanje, sortiranje i analizu. Na primjer, softver za pretraživanje, koji je zasnovan na XML-u, može veoma lako pronaći bilo koji proizvod koji mu je specifikovan, zahvaljujući identifikacionim tagovima kojim je taj proizvod označen. Takođe, veb-sajtovi koji koriste XML omogućavaju jednostavno utvrđivanje koje su veb-strane klijenti koristili i koje su proizvode razmatrali. Dakle, XML, podržavajući i unapređujući elektronsku razmjenu poslovnih podataka između kompanija i njihovih klijenata, dobavljača i drugih poslovnih partnera, omogućava jednostavniji i efektivniji e-commerce.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
- <!--
Edited by XMLSpy®
-->
- <CATALOG>
- <PLANT>
  <COMMON>Bloodroot</COMMON>
  <BOTANICAL>Sanguinaria canadensis</BOTANICAL>
  <ZONE>4</ZONE>
  <LIGHT>Mostly Shady</LIGHT>
  <PRICE>$2.44</PRICE>
  <AVAILABILITY>031599</AVAILABILITY>
</PLANT>
- <PLANT>
  <COMMON>Columbine</COMMON>
  <BOTANICAL>Aquilegia canadensis</BOTANICAL>
  <ZONE>3</ZONE>
  <LIGHT>Mostly Shady</LIGHT>
  <PRICE>$9.37</PRICE>
  <AVAILABILITY>030699</AVAILABILITY>
</PLANT>
```

Slika 1.65. Primjer XML dokumenta

Java jeste objektno orijentisani programski jezik koji je revolucionizovao programiranje aplikacija za web i organizacione intranet i ekstranet mreže. Java je sličan C++ jeziku, ali je mnogo jednostavniji, sigurniji i mašinski nezavisniji od njega. Java programski jezik specijalno je dizajniran za real-time, interaktivne, veb bazirane mrežne aplikacije. Java aplikacija se sastoji od malih aplikativnih programa, tzv. apleta koji mogu biti izvršeni na bilo kom kompjuteru ili operativnom sistemu u mreži.

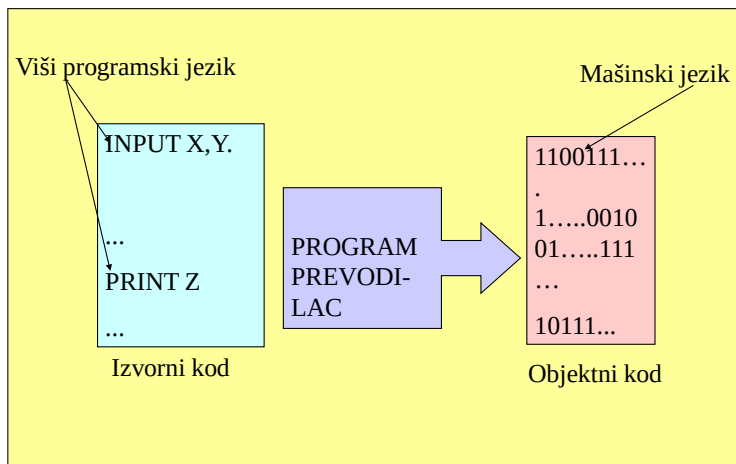
Jednostavnost kreiranja Java apleta i njihovog distribuiranja od servera do klijentskih PC i drugih mrežnih kompjutera jeste glavni razlog popularnosti ovog programskog jezika. Apleti su mali programi specijalne namjene ili mali moduli velikih aplikativnih programa. Apleti su rezidentni na veb-sajtovima na serveru sve dok ne budu zatraženi od strane klijenta, kada se jednostavno distribuiraju preko interneta ili intranet i ekstranet mreže. Apleti su potpuno nezavisni od platforme. Mogu se ravnopravno izvršavati pod Windows, UNIX ili nekim drugim sistemom bez ikakvih modifikacija. Java takođe unapređuje brzinu izvršavanja, što je njen glavni nedostatak.

Programski alati jesu alati koji pomažu programerima da razvijaju kompjuterske programe. Tu spadaju prevodioci programskih jezika, programski editori i CASE (Computer Aided Software Engineering) alati.

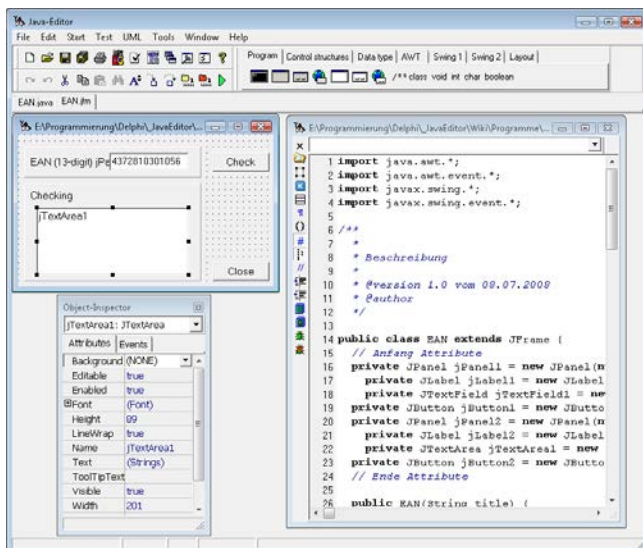
Prevodioci programskih jezika. – Kompjuterski program se sastoji od niza instrukcija napisanih programskim jezicima (izvorni kod), koji moraju biti prevedeni pomoću programa prevodioca u mašinski jezik (izvršni ili objektni kod), kako bi mogli biti procesirani, tj. izvršeni od strane CPU (slika 1.66.). Programi prevodioci su asembleri, kompajleri i interpreteri. Asembleri prevode programe napisane u asemblerskom jeziku na mašinski jezik. Kompajleri prevode naredbe viših programskih jezika u mašinski kod. Interpreter je specijalni tip kompajlera koji prevodi i izvršava jednu po jednu instrukciju programa, za razliku od kompajlera, koji prvo produkuje cjelokupan mašinski kod, pa ga tek onda izvršava u cjelosti. Java je primjer jezika koji se prevodi interpreterom. Programske instrukcije u okviru Java apleta interpretiraju se i izvršavaju “u letu” kada počne izvršavanje apleta na klijentskom PC računaru.

Programski editori omogućavaju editovanje i debugovanje (identifikovanje grešaka) izvornog programskog koda. Oni obično imaju GUI (omogućavaju vizuelno programiranje) i pomažu programerima da efikasno i jednostavno

identifikuju, minimiziraju i isprave greške prilikom programiranja. Slika 1.67. prikazuje editor za programski jezik Java.

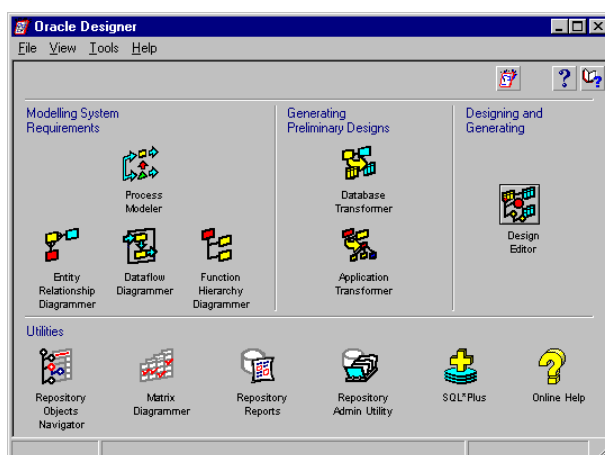


Slika 1.66. Prevođenje izvornog koda u objektni



Slika 1.67. Java editor

CASE (Computer Aided Software Engineering) jesu alati za automatsko generisanje softvera. Oni obično sadrže dijagram pakete pomoću koji se prave grafičke specifikacije softvera, generatore koda, koji na osnovu ovih specifikacija generišu programski kod, biblioteke i prototipove. Ovi alati povećavaju efikasnost i produktivnost programera, kao i kvalitet dobijenih programa. Sve ovo snižava cijenu softvera, koji je inače najskuplja komponenta kompjuterskih sistema. Najviše se primjenjuju kod baza podataka za generisanje baze i aplikacija. Na slici 1.68. predstavljen je CASE alat *oracle designer*.



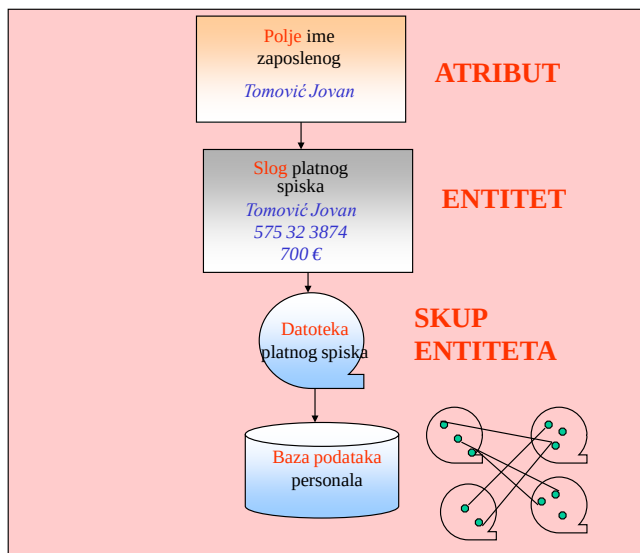
Slika 1.68. CASE alat oracle designer

1.6. ORGANIZACIJA PODATAKA

Podaci predstavljaju vitalni organizacioni resurs kojim se mora upravljati kao bilo kojim drugim poslovnim sredstvom. Danas preduzeća ne mogu opstati niti uspješno poslovati bez kvalitetnih podataka o sopstvenom poslovanju i eksternom okruženju. Upravljanje podacima podrazumijeva poznavanje IS tehnologija, kao što su DBMS, data warehousing i druge, kao i metodologija upravljanja i organizovanja podataka koji će zadovoljiti informacione zahtjeve preduzeća.

1.6.1. OSNOVNI KONCEPTI PODATKA

Podaci su u informacionom sistemu logički organizovani u nekoliko hijerarhijskih nivoa u vidu karaktera, polja, slogova, fajlova i baza podataka (slika 1.69.).



Slika 1.69. Hijerarhijska organizacija podataka

Karakter je najelementarnija jedinica podatka koja se sastoji od jednog alfabetskog, numeričkog ili nekog drugog znaka (bit i bajt su još elementarnija jedinica podatka, ali se one odnose na fizičku, a ne na logičku organizaciju podataka). Sa korisničkog stanovišta (tj. sa logičkog stanovišta), karakter je najosnovniji dio podatka koji može biti razmatran i kojim se može upravljati.

Polje je sljedeći viši nivo organizovanja podatka. Polje se sastoji od niza karaktera. Na primjer, niz karaktera koji predstavlja ime neke ličnosti čini polje IME, a niz cifara i specijalnih znakova razdvajanja (kao što su tačka i zarez) koji predstavljaju iznos prodaje – čine polje IZNOS PRODAJE. Polje obično predstavlja neki atribut (osobinu ili svojstvo) nekog entiteta (objekta, osobe, mjesta ili događaja). Na primjer zarada zaposlenog je atribut entiteta ‘zaposleni’.

Slog. – Logički povezana polja grupišu se u slogove (zapise ili *recorde*). Dakle, slog predstavlja skup atributa koji opisuju neki entitet. Primjer za to je slog platnog spiska koji može sadržati polja koja predstavljaju attribute ime, broj socijalnog osiguranja i iznos plate. Slog može biti fiksne dužine, ako sadrži fiksiran broj polja fiksirane dužine, ili varijabilne dužine, u slučaju da sadrži varijabilan broj polja promjenljive dužine.

Fajl ili datoteka predstavlja grupu međusobno povezanih slogova. U teoriji baza podataka ovakva grupa povezanih slogova još se naziva i **tabela** ili **skup entiteta**. Tako fajl, tj. tabela ‘zaposleni’ sadrži slogove svih zaposlenih u firmi. Fajlovi se najčešće klasifikuju zavisno od primjene, kao na primjer: fajl platnog spiska, fajl zaliha itd. Takođe se mogu klasifikovati zavisno od tipa podataka koje sadrže, kao što su dokument fajl, grafički fajl i dr.

Baza podataka jeste integrisana kolekcija logički povezanih slogova. Baza podataka objedinjuje slogove raspoređene u različitim fajlovima u jedan zajednički logički prostor koji obezbjeđuje podatke za više različitih aplikacija. Baza podataka ne zavisi od aplikacija koje je koriste kao ni od tipa memorijskih uređaja koji se koriste za smještaj podataka. Dakle, baza podataka sadrži skupove entiteta (tabele) kao i odnose između entiteta. Na primjer, baza podataka personala može sadržati datoteke, tj. skupove entiteta: Zaposleni, Izdržavana lica, Radno iskustvo, Platni spisak i druge.

1.6.2. SISTEMI ZA UPRAVLJANJE BAZAMA PODATAKA

Sistem za upravljanje bazom podataka ili DBMS (Database Management System) jeste softverski paket koji predstavlja interfejs između korisnika i baze podataka. On pomaže korisnicima da jednostavno i lako pristupaju slogovima u bazi podataka. Šire, osnovne primjene DBMS programa jesu razvoj, pretraživanje i održavanje baze podataka i razvoj korisničkih aplikativnih programa, kako bi se obezbijedile ažurne informacije potrebne korisnicima i njihovim organizacijama. Tabela 1.4. prikazuje osnovne funkcije i komponente DBMS-a koje ih podržavaju.

Razvoj baze podataka. – DBMS paketi, kao što su Microsoft Access ili Lotus Approach, omogućavaju krajnjim korisnicima jednostavan razvoj baze podataka koja im je potrebna. Velike organizacije sa klijent/server ili mainframe baziranim sistemom problem razvoja baze podataka povjeravaju projektantima i administratorima baze podataka (DBA – database

administrators) i drugim stručnjacima iz ove oblasti. Na taj način se obezbjeđuje i poboljšava integritet i sigurnost podataka u organizacijskim bazama. Projektanti i administratori baze podataka koriste DDL (data definition language) koji se nalazi u okviru DBMS-a (kao što je na primjer *oracle* ili IBM DB2) za specifikaciju i generisanje podataka, odnosa i strukture baze, kao i za njihovu modifikaciju, ukoliko je to potrebno. Te informacije, tj. specifikacije i definicije podataka čuvaju se u okviru *rječnika podataka* kojim upravlja DBA.

Tabela 1.4.: Funkcije i komponente DBMS-a

Funkcija	DBMS komponenta
Definisanje baze podataka	Jezici i grafički alati za definisanje entiteta, veza, integritetskih ograničenja i prava pristupa
Neproceduralni pristup	Jezici i grafički alati za pristup podacima bez kompletnog kodiranja
Razvoj aplikacija	Grafički alati za razvoj menija, formi za unos podataka i izvještaja, 4GL jezici
Proceduralni jezički interfejs	Jezici koji kombinuju neproceduralni pristup podacima sa svim mogućnostima višeg programskog jezika
Procesiranje transakcija	Kontrolni mehanizmi koji sprečavaju konflikte prilikom konkurentnog dostupa i oporavljaju izgubljene podatke u slučaju kraha sistema
Podešavanje baze podataka	Alati za praćenje i poboljšanje performansi baze podataka.

Rječnik podataka jeste alat za administraciju baze podataka. Riječ je o kompjuterskom katalogu (direktorijumu) koji sadrži podatke o podacima, tj. **metapodatke**. Rječnik podataka uključuje softversku komponentu koja upravlja bazom metapodataka, tj. podacima o strukturi i drugim karakteristikama baze. Na primjer, rječnik podataka može sadržati nazive i opise svih tipova slogova, kao i njihove međusobne veze, ograničenja pristupa krajnjih korisnika i aplikativnih programa i drugo. Administrator baze podataka može pretraživati rječnik podataka i provjeravati status

metapodataka sa bilo kojeg aspekta. On takođe može po potrebi mijenjati definicije podataka.

Pretraživanje baze podataka. – Mogućnosti pretraživanja (postavljanja upita) baze jesu glavna prednost DBMS-a. Pretraživanje baze podataka korisnici obavljaju pomoću upitnih jezika ili generatora izvještaja, koje DBMS sadrži u sebi. Neposredne odgovore dobijaju se ili u vidu ekranskih prikaza ili u vidu štampanih izvještaja. Upitni jezici ne zahtijevaju kompleksno programiranje, već se definiše kratki i jednostavni ad hoc zahtjev. Generator izvještaja omogućava jednostavno formatiranje informacija, tj. izvještaja i njihovo generisanje, odnosno štampu.

SQL (Structured Query Language) jeste upitni jezik kojeg posjeduje veliki broj DBMS-ova. Osnovni oblik SQL upita jeste:

SELECT... FROM... WHERE

Poslije SELECT klauzule navodi se lista polja koja korisnik želi da dobije. Poslije FROM klauzule navodi se lista fajlova, tj. tabela iz kojih se podaci pretražuju. U WHERE klauzuli se specifikuje uslov koji pretraživanje ograničava na one slogove koji ga zadovoljavaju. Mnogi DBMS paketi omogućavaju GUI, tj. *point and click* metod za definisanje upita, koji je lakši za upotrebu. Ovako grafički definisani upiti prevode se zatim u odgovarajuću SQL komandu. Neki paketi podržavaju upitne naredbe upotrebom prirodnog jezika, sličnog koverzacijskom engleskom jeziku. Tabela 1.5. prikazuje komparaciju upita prirodnim jezikom i SQL upita za Microsoft Access.

Održavanje baze podataka podrazumijeva kontinuirane promjene podataka koje odražavaju nove poslovne transakcije ili neke druge poslovne događaje, kao i druge vrste promjena koje osiguravaju tačnost podataka u bazi. Ovaj proces održavanja baze podataka obavljaju programi za procesiranje transakcija i korisnički aplikativni programi.

Razvoj aplikacija za baze podataka DBMS omogućava na taj način što **krajnjim korisnicima, analitičarima sistema i programerima** obezbjeđuje interne 4GL jezike i ugrađene alate za razvoj softvera. Na primjer, pomoću DBMS alata mogu se razviti ekrani za unos podataka, izvještaji ili veb-strane za neku poslovnu aplikaciju.

Tabela 1.5. Komparacija upita prirodnim i SQL jezikom

Prirodni jezik WHAT CUSTOMERS DIDN'T HAVE ANY ORDERS LAST MONTH?
SQL upit (MS ACCESS) SELECT [Customers].[Company Name], [Customers].[Contact Name] FROM [Customers] WHERE not Exists {SELECT [Ship Name] FROM [Orders] where Month {[Order Date]} = 12 and Year {[Order Date]} =2004 and [Customer].[Customer ID]= [Orders].[Customer ID]}

Posao aplikacionih programera je znatno olakšan jer se ne moraju razvijati detaljne procedure za upravljanje podacima. Umjesto toga oni jednostavno koriste naredbe DML (Data Manipulation Language) jezika, koje pozivaju DBMS da izvrši potrebne aktivnosti u vezi sa upravljanjem podacima (unos novih podataka, brisanje, izmjene, pretraživanje).

1.6.3. TIPOVI BAZA PODATAKA

Kontinuirani razvoj informacionih tehnologija i njihovih poslovnih primjena rezultirao je evolucijom različitih tipova baza podataka.

Operacione (transakcione, produkcione) baze podataka memorišu detaljne i aktuelne podatke potrebne za podršku poslovnih procesa i operacija u e-biznis preduzeću. Primjeri za to su baza kupaca, baza ljudskih resursa – personala, baza zaliha i mnoge druge baze generisane poslovnim operacijama.

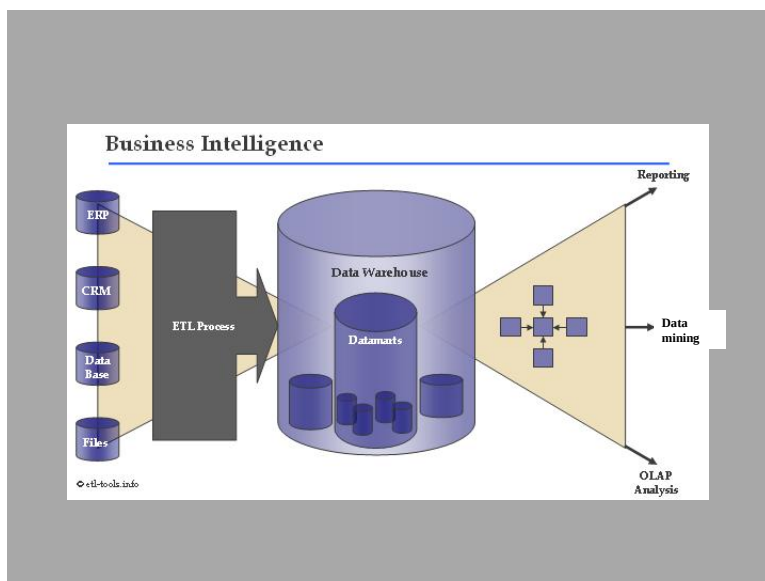
Distribuirane baze podataka. – Mnoge organizacije repliciraju i distribuiraju kopije ili djelove baze podataka na različite mrežne servere. Distribuirane baze mogu se, dakle, nalaziti na različitim serverima web, intranet, ekstranet ili neke druge kompanijske mreže. Distribuirane baze mogu biti kopije operacionih ili analitičkih, hipermedijalnih ili diskusionih ili nekih drugih tipova baza podataka. Replikacija i distribucija baza

podataka umanjuje performanse i sigurnost baze podataka. Garancija da će svi podaci u organizacijskim distribuiranim bazama podataka biti konzistentno i istovremeno ažurirani jeste jedan od glavnih zadataka distribuiranih DBMS-a.

Eksterne baze podataka. – Pristup informacijama iz eksternih baza podataka moguć je putem velikog broja komercijalnih online servisa, kao i sa plaćanjem ili bez plaćanja iz mnogih izvora na internetu. Veb-sajtovi obezbjeđuju putem hiperlinkovanih strana pretraživanje multimedijalnih dokumenata u hipermedijalnim bazama podataka. Eksterni podaci su najčešće dati u vidu statističkih baza podataka, nastalih kao rezultat ekonomskih i demografskih istraživanja i aktivnosti. Takođe su popularne bibliografske ili full text baze, iz kojih se mogu vidjeti ili downloadovati apstrakti ili kompletne kopije raznih časopisa, istraživačkih radova i drugih publikovanih materijala.

Data Warehouse i Data Mining. – Data warehouse baza podataka memoriše podatke ekstrahovane iz različitih operacionih, eksternih ili nekih drugih baza podataka u okviru organizacije. Ona je centralni izvor prečišćenih i transformisanih aktuelnih i istorijskih podataka koje koriste menadžeri i drugi poslovni stručnjaci za data mining, OLAP (online analytical processing) i za druge vrste poslovnih analiza, istraživanja tržišta, kao i za podršku odlučivanju. Data warehouse može biti podijeljen na **data martove**, koji sadrže podskupove podataka warehouse-a fokusiranih na neki specifični aspekt kompanije, kao što je npr. odjel ili poslovni proces.

Proces prečišćavanja i transformacije podataka (ETL – extraction, transformation and loading), podrazumijeva konsolidaciju podataka iz nekoliko različitih izvora, filtriranje i izbacivanje nepotrebnih podataka, korigovanje nekorektnih podataka, konvertovanje podataka u nove višedimenzionalne strukture podataka i agregiranje. Tako se prečišćeni podaci smještaju u data warehouse odakle se mogu prenositi u odgovarajuće data martove ili analitičke baze u formatu pogodnom za različite tipove analiza. Prenosjenje podataka iz izvornih baza u data warehouse obavlja se po potrebi svakih nekoliko sati, dnevno, sedmično ili mjesečno. Postoji veliki broj analitičkih softverskih alata za internet i intranet veb-sisteme i druge, koji omogućavaju upite, izvještavanje, data mining, modeliranje i analize iz data warehousea. Ovi sistemi za punjenje podataka u DW, tj. sama DW baza zajedno sa alatima za OLAP i data mining analizu, čine *business intelligence* (BI) sisteme (slika 1.70.).



Slika 1.70. BI sistem

Data mining jeste jedna od glavnih primjena data warehouse baza. Data mining procesom podaci se u data warehouseu analiziraju radi otkrivanja skrivenih obrazaca ponašanja i trendova u velikoj količini podataka nastalih u poslovnim aktivnostima organizacije u tekućem periodu (aktuelni podaci) ali i prošlim periodima (istorijski podaci). Ovo pomaže menadžerima da donose odluke o strateškim promjenama u poslovnim aktivnostima radi što boljeg pozicioniranja na tržištu.

Data mining procesom automatski se mogu otkriti nove korelacije, modeli i trendovi u velikim količinama poslovnih podataka (najčešće nekoliko TERABAJTA podataka). Data mining softver koristi napredne algoritme za prepoznavanje modela, koji su zasnovani na različitim statističkim tehnikama (stabla odlučivanja) ili mašinskom učenju (neuronske mreže, genetski algoritmi itd.). Data mining, OLAP i druge tehnologije za analizu podataka iz data warehousea danas predstavljaju vitalnu podršku za poslovno odlučivanje.

Hipermedijalne baze podataka na webu. – Rapidni napredak **web-**tehnologija značajno je povećao primjenu baza hipermedijalnih dokumenata, koje se nazivaju hipermedijalne baze podataka.

Hipermedijalna baza podataka je zapravo veb-sajt koji se sastoji od hyperlinkovanih multimedijalnih strana. Dakle, sa tačke gledišta teorije baza podataka, hipermedijalna baza podataka nije skup međusobno povezanih slogova, već skup međusobno hyperlinkovanih multimedijalnih strana.

Hipermedijalne baze podataka realizuju se u troslojnoj klijent/server mreži (intranet mreži) – klijent sa web browserom, veb-server i server hipermedijalne baze podataka. Klijentov PC, koristeći web browser ostvaruje konekciju sa veb-serverom. Veb-server izvršava softver za pristup i transfer zahtijevanih veb-strana. Svaka veb-strana opisana je HTML kodom ili XML tagovima, a sastoji se od skupa grafičkih, video i audio fajlova. Veb-server aktivira DBMS koji dalje upravlja pronalaženjem i transferom hipermedijalnih fajlova do web browsera.

1.6.4. PROBLEMI I IZAZOVI UPRAVLJANJA PODACIMA

Menadžeri moraju gledati na podatke kao na veoma važan resurs koji zahtijeva odgovarajuće upravljanje u cilju uspjeha i opstanka njihove organizacije. Međutim, ovo je mnogo lakše reći nego ostvariti. Ovo najčešće podrazumijeva niz kompleksnih aktivnosti, kao što su definisanje i primjena polisa za vlasništvo i pristup podacima, definisanje strateških i tehničkih planova baze podataka, setovanje i primjena operativnih procedura, upravljanje dizajniranjem i administracijom baze podataka, ocjena i izbor softvera za bazu podataka i drugo. Međutim, dobro organizovano upravljanje podacima obezbjeđuje menadžmentu niz važnih prednosti.

Nekada su organizacije za procesiranje informacija umjesto baza podataka koristile fajl sisteme. Kod ovih sistema, svako funkcionalno područje imalo je svoj skup nezavisnih fajlova. Svaki fajl je imao svoj skup programa koji njime upravljaju. Na taj način organizacija je čuvala iste informacije u više fajlova, a za deset godina poslovanja imala je na stotine fajlova i programa. Glavni problemi ovakve organizacije podataka jesu redundanca podataka (dupliciranje podataka u više različitih fajlova), konfuzija (isti podaci mogu imati sasvim različita značenja u različitim aplikacijama), zavisnost programa od podataka (promjena formata podatka u nekom fajlu zahtijeva izmjenu svih programa koji pristupaju tom fajlu), nefleksibilnost (izvještava se samo na osnovu zahtijevanog programiranja, nema mogućnosti direktnih upita i ad hoc izvještaja), nesigurnost podataka (nema mogućnosti

definisanja prava pristupa) i nemogućnost dijeljenja podataka između više aplikacija.

Pomoću DBMS-a reducira se dupliciranje podataka, podaci se integrišu i omogućava se pristup od strane više različitih korisničkih aplikacija. Aplikacije su nezavisne od formata podataka i tipa memorijskih uređaja. DBMS dozvoljava menadžerima da koriste direktne ad hoc upite bazi i automatski generišu različite izvještaje iz baza podataka, data warehousea ili data martova bez kompleksnog programiranja. Razvoj softvera je pojednostavljen jer programi ne zavise od logičkog formata podataka ili načina njihovog fizičkog smještaja na sekundarnim memorijskim uređajima. Na kraju, integritet i sigurnost podataka su povećani, jer se pristup i izmjene podataka kontrolišu od strane DBMS softvera i rječnika podataka.

Problemi u upravljanju podacima proističu iz tehnološke kompleksnosti i velike količine poslovnih podataka kojima se mora upravljati. Razvoj velikih baza podataka sa kompleksnim tipovima podataka i instalacija data warehousea može biti veoma teška i skupa. Javlja se potreba za novim i većim hardverskim resursima jer su memorijski zahtjevi za organizacione podatke, kontrolne i analitičke podatke, kao i za softver za upravljanje bazom i data warehouseom znatno veći. Kompleksnost podataka i softvera može dovesti do produženja vremena procesiranja i sporog odziva. Na kraju, ukoliko organizacija nema centralizovanu bazu podataka, njena podložnost greškama, upadima i krahovima se povećava. Osim toga, kod distribuiranih baza podataka može doći i do nekonzistentnosti podataka. Prema tome, podrška sigurnosti i integritetu podataka u okviru baza podataka i data warehousea jeste glavni cilj upravljanja podacima u e-biznis preduzeću.

1.6.5. TEHNIČKE OSNOVE UPRAVLJANJA PODACIMA

Može se samo zamisliti koliko bi bilo teško dobiti bilo koju informaciju iz informacionog sistema u kome su podaci sačuvani neorganizovano i gdje ne postoji mogućnost sistematskog pristupa podacima. Dakle, u okviru informacionog sistema podaci moraju biti organizovani i strukturirani u nekom logičkom formatu koji će omogućiti jednostavan pristup, efikasno procesiranje, brzo pretraživanje i efikasno upravljanje njima. Strukture podataka i metode pristupa tim podacima čine suštinu tehničkih koncepata organizacije podataka.

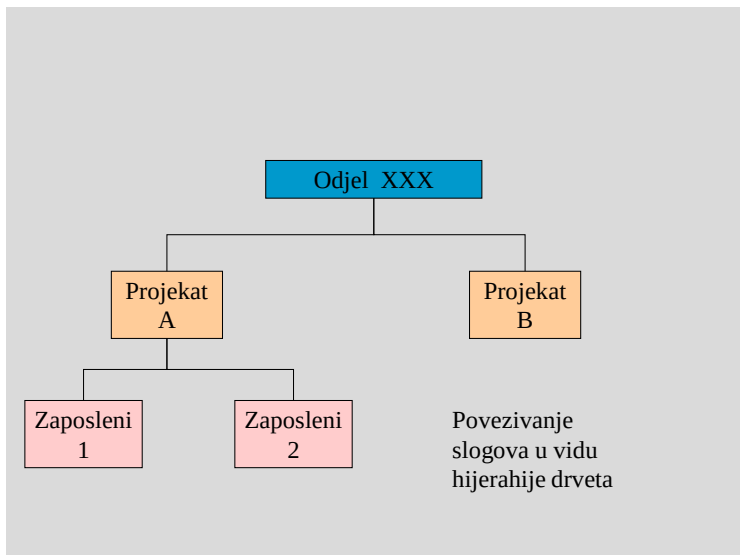
1.6.5.1. STRUKTURE PODATAKA

Veze između slogova u bazi podataka bazirane su na nekoj logičkoj strukturi podataka ili **modelu podataka**. DBMS paketi dizajnirani su tako da koriste određenu strukturu podataka, obezbeđujući korisnicima brz i jednostavan pristup informacijama u bazi podataka. Pet osnovnih modela podataka jesu: hijerarhijski, mrežni, relacioni, objektno orijentisani i višedimenzionalni model.

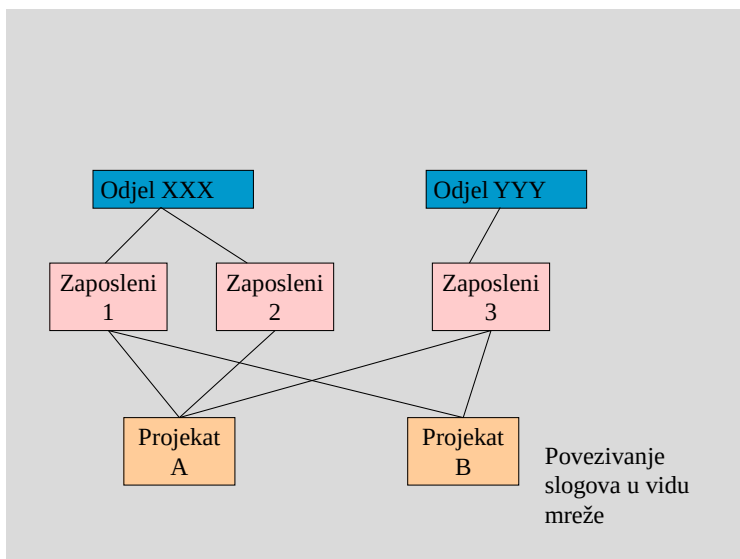
Hijerarhijska struktura. – Raniji mainfram bazirani DBMS paketi koristili su hijerarhijsku strukturu, kod koje se veze između slogova prikazuju u vidu hijerarhijske strukture slične drvetu. Kod klasičnog hijerarhijskog modela svi slogovi su zavisni i aranžirani su u neku strukturu sa više nivoa, koja se sastoji od jednog korijenskog sloga i više podređenih nivoa. Sve veze između slogova imaju odnos *jedan prema mnogo*, pošto je svaki slog povezan sa samo jednim iznad sebe. Slog na najvišem nivou hijerarhije naziva se korijen. Na primjer, na slici 1.71. korijen je slog *Odjel XXX*. Podacima se pristupa tako što se polazi od korijena i progresivno se kreće po granama drveta prema dolje, sve dok se ne locira željeni slog.

Mrežna struktura može predstavljati i kompleksnije logičke veze, i ona se još uvijek koristi kod nekih mainframe DBMS paketa. Ona omogućava veze tipa *mnogo prema mnogo*. Kod mrežnog modela slogu se pristupa tako što se slijedi jedan od više mogućih puteva, jer jedan slog može biti povezan sa više drugih slogova. Na primjer, na slici 1.72. slog odjela povezan je sa više slogova zaposlenih, jedan zaposleni može biti povezan sa više projekata, kao što se i jedan slog projekat može povezati sa više slogova zaposlenih. Tako uvijek možemo pronaći sve zaposlene u nekom odjelu, ili sve projekte na kojima učestvuje neki zaposleni, kao i sve zaposlene koji rade na nekom projektu.

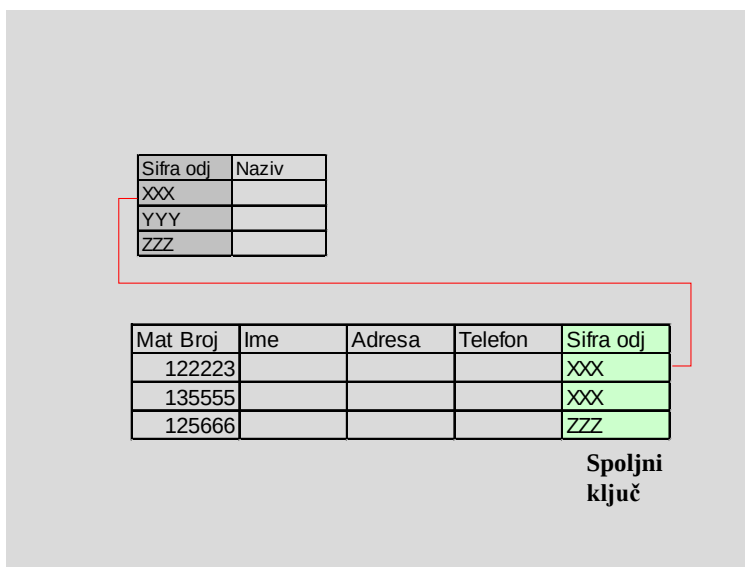
Relaciona struktura jeste najpopularniji model jer ga koristi većina DBMS paketa, kako za mikrokompjutere tako i za midrange i mainframe sisteme. Kod relacionog modela podaci se čuvaju u vidu jednostavnih tabela. Slika 1.73. prikazuje relacioni model baze podataka sa dvije tabele koje pokazuju odnos između odjela i zaposlenih u njima. U toj bazi podataka mogu postojati i druge tabele koje prikazuju odnose između projekata, odjeljenja, proizvodnih linija i drugo.



Slika1.71. Hijerarhijska struktura



Slika1.72. Mrežna struktura

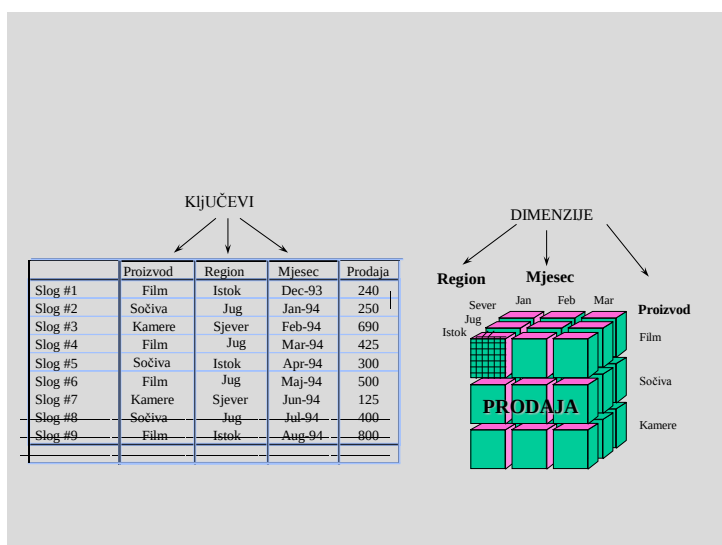


Slika 1.73. Relaciona struktura

DBMS paketi bazirani na relacionom modelu mogu povezivati slogove iz različitih tabela u odnosima *jedan prema jedan*, *jedan prema mnogo* i *mnogo prema mnogo*. Za odnose *jedan prema jedan* i *jedan prema mnogo* tabele se povezuju tako što se tabeli prema kojoj je odnos mnogo, doda jedna kolona iz nadređene tabele. Ova kolona iz nadređene tabele jednoznačno identifikuje slogove u okviru nje i naziva se **primarni ključ** (primarni ključ se može sastojati i iz više kolona, tj. atributa). Tako je na slici 1.73. tabeli zaposleni dodat atribut ŠIFRA ODJELA, jer je on primarni ključ njoj nadređene tabele odjel. Na taj je način svaki zaposleni povezan sa odgovarajućim odjelom. Kolona, tj. atribut ŠIFRA ODJELA u tabeli zaposleni jeste **spoljni ključ**. Odnos između slogova *mnogo prema mnogo* predstavlja se posebnom tabelom, čije su kolone spoljni ključevi tabela koje povezuje. Da bi se obezbijedile adekvatne informacije za korisnika, relacioni DBMS prilikom pretraživanja povezuje slogove iz različitih tabela. Tako DBMS može pronaći i prikazati ime zaposlenog i njegovu adresu (iz tabele zaposleni) i naziv odjela u kome radi (iz tabele odjel), koristeći pri tome za povezivanje ŠIFRU ODJELA.

Višedimenzionalna struktura jeste varijacija relacionog modela koji koristi višedimenzionalne strukture podataka za organizovanje podataka i predstavljanje odnosa među njima. Višedimenzionalna struktura se može

vizuelno posmatrati kao kocka sastavljena od manjih kocaka. Svaka strana kocke je jedna dimenzija podatka (činjenice) koji kocka predstavlja. Slika 1.74. prikazuje kocku koja predstavlja prodaju preko dimenzija proizvod, region i mjesec. Svaka mala kocka u okviru ove višedimenzionalne strukture sadrži agregirani iznos prodaje za neki proizvod u nekom regionu za jedan mjesec. Ona se dalje sastoji od još manjih kocaka koje prikazuju iznos prodaje na detaljnijem nivou, na primjer, na nivou dana, vrste proizvoda ili grada. Na slici je prikazana i odgovarajuća relaciona tabela kojom se realizuje ova kocka.



Slika 1.74. Višedimenzionalna struktura

Osnovna prednost višedimenzionalnih struktura jeste način vizuelizacije podataka, koji je kompaktan i jednostavan za razumijevanje. Na taj način, višedimenzionalne baze podataka postaju najpopularnije za podršku odlučivanju, tj. za brze odgovore na kompleksne i ad hoc poslovne upite. Ovakve brze preokrete u procesu pretraživanja podataka omogućavaju OLAP (Online Analytical Processing) alati. Ovi alati pružaju efikasne analitičke tehnike, kao što su *drill-up* i *drill-down* za agregiranje i spuštanje do nivoa detalja podataka (zumiranje malih kocaka), kao i za rotaciju kocke *pivot* operacijom. Jedan od poznatih OLAP alata jeste *oracle discoverer*. Slika 1.75. prikazuje jedan model (OLAP kocku) za analizu prodaje u ovom alatu.

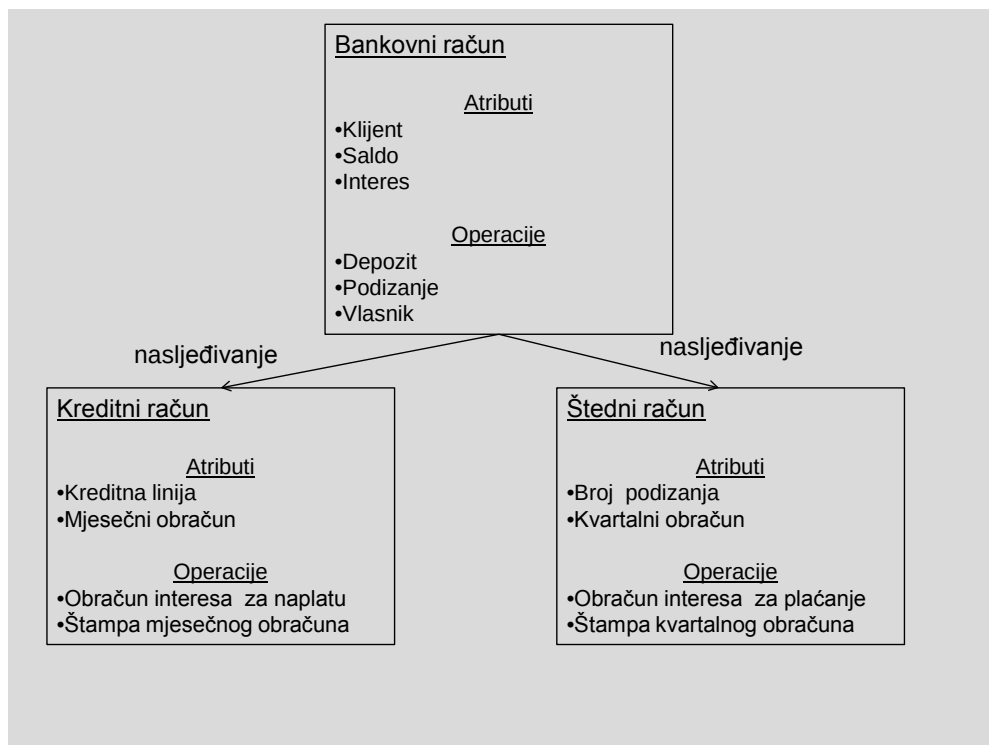
	Jan		Feb
	Prihod SUM	Trosak SUM	Profit SUM
Pr mj - Podgorica	122.000000	109.000000	13.000000
Hemikalija A			
Mast A			
Sampion Kosili	60.000000	52.000000	8.000000
Sapun Cleopatra	62.000000	57.000000	5.000000
Pr mj - Stari Bar	45.000000	40.000000	5.000000
Sampion Kosili	45.000000	40.000000	5.000000
Sum:	167.000000	149.000000	18.000000
Average:	83.500000	74.500000	9.000000

Slika 1.75. Oracle discoverer – OLAP analiza prodaje

Kada smo govorili o data warehouse tipu baza podataka, rekli smo da je njihova struktura višedimenzionalna. Podaci se iz operativnih baza podataka prilikom prenošenja u data warehouse moraju transformisati, tj. agregirati kako bi se prilagodili višedimenzionalnoj strukturi podataka.

Objektno orijentisani modeli podataka predstavljaju ključnu tehnologiju za novu generaciju multimedijalnih veb baziranih aplikacija. Kao što prikazuje slika 1.76., objekat se sastoji od vrijednosti podataka koji predstavljaju attribute nekog entiteta i operacija koje se mogu izvršiti nad tim podacima. Ova osobina enkapsulacije podataka omogućava bolje upravljanje kompleksnim tipovima podataka kao što su grafika, glas, tekst, video zapisi i drugo. Objektno orijentisani modeli takođe podržavaju princip nasljeđivanja kod koga objekti na nižem nivou nasljeđuju neke ili sve osobine roditeljskih objekata. Tako na slici 1.76., kreditni i depozitni računi nasljeđuju zajedničke attribute i operacije od roditeljskog objekta bankovni računi. Ove mogućnosti čine objektno orijentisane DBMS popularnim, posebno kod multimedijalnih veb baziranih aplikacija na internetu ili intranet i ektranet mrežama, kao i za upravljanje hipermedijalnim bazama podataka. Industrijske prognoze govore da će objektno orijentisani DBMS postati ključna platforma za upravljanje hiperlinkovanim multimedijalnim

veb-stranama korporacijskih veb-sajtova. Ovo je moguće zato što objektno orijentisani DBMS lako upravljaju i pristupaju objektima kao što su dokumenta, fotografije, video i audio zapisi i drugo. Oni mogu raditi mnogo bolje nego relacioni modeli sa kompleksnim tipovima podataka. Stoga danas najviše korišćeni relacioni DBMS (RDBMS) ugrađuju objektivne ekstenzije svojih modela.



Slika 1.76. Objekat *bankovni račun* u objektnom modelu baze podataka banke

Evaluacija modela podataka. – Hijerarhijski model podataka je prirodan model za baze podataka koje se koriste za strukturirane rutinske tipove procesiranja transakcija, kakve imamo kod većine poslovnih operacija. Podaci nastali iz takvih operacija mogu se jednostavno predstaviti kao grupe slogova sa hijerarhijskim vezama. Međutim, postoji mnogo slučajeva u kojima slogovi nemaju hijerarhijske odnose. Na primjer, zaposleni iz jednog

ili više odjela mogu raditi na jednom ili više projekata. Mrežna struktura je povoljna za predstavljanje odnosa *mnogo prema mnogo*. Ona je fleksibilnija od hijerarhijske i podržava različite tipove poslovnih operacija. Međutim, isto kao i kod hijerarhijske strukture, odnosi moraju biti specifikovani unaprijed, pa model ne podržava ad hoc zahtjeve za informacijama.

Relacioni model, sa druge strane, omogućava jednostavno dobijanje informacija kao odgovora na ad hoc zahtjeve. Ovo je moguće stoga što se sve veze između slogova ne moraju specifikovati unaprijed, već ih DBMS kreira u vidu novih tabela, koristeći djelove tabela koje ulaze u te veze. Relacione baze podataka su lakše za projektante i programere jer ne rade na nivou slogova, već rade sa relacijama, tj. tabelama. Međutim, glavni nedostatak relacionog modela u odnosu na ova dva prethodna jeste to što on ne može procesirati tako veliki broj transakcija brzo i efikasno kao hijerarhijski i mrežni modeli, ili recimo kao objektno orijentisani modeli. Ovaj nedostatak se kod naprednih RDBMS prevazilazi uvođenjem objektno orijentisanih ekstenzija. DBMS bazirani na objektno orijentisanim i višedimenzionalnim modelima koji omogućavaju OLAP i veb bazirane aplikacije trenutno predstavljaju trendove u ovoj oblasti.

1.6.5.2. METODE PRISTUPA PODACIMA

Efikasan pristup podacima je veoma bitan. Prilikom održavanja baze podataka, slogovi ili objekti se kontinuirano dodaju, brišu ili ažuriraju kako bi reflektovali poslovne transakcije. Podacima se stalno pristupa da bi se obezbijedili odgovori na korisničke zahtjeve.

Da bi ovo bilo moguće, slogovi obično imaju jedno ili više identifikacionih polja, tj. **primarnih ključeva** koji omogućavaju identifikovanje i lociranje sloga. Na primjer, broj indeksa je polje koje jednoznačno identifikuje slog 'student'.

Relacioni DBMS koriste primarne ključeve za povezivanje slogova u različitim datotekama (tabelama). Svaka tabela (fajl) u relacionom modelu mora imati primarni ključ. Ovo polje ili grupa polja jednoznačno identifikuje slogove u okviru fajla, i može se naći u drugim povezanim fajlovima. Na primjer, na slici 1.73. šifra odjela jeste primarni ključ tabele 'odjel', ali i polje tabele 'zaposleni', tj. predstavlja njen spoljni ključ. Na taj

način RDBMS može obezbijediti informacije iz više povezanih tabela koristeći primarne i spoljne ključeve.

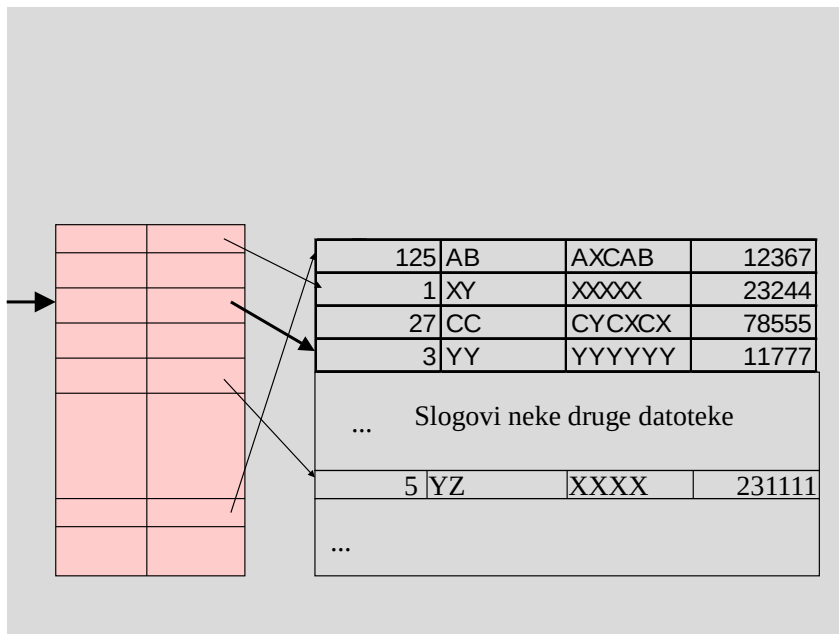
Kod hijerarhijskog i mrežnog modela povezivanje slogova obavlja se pomoću pointer polja. Ovo polje ukazuje na lokaciju povezanog sloga u okviru istog ili nekog drugog fajla. Na taj način DBMS zasnovani na ovim modelima mogu pretraživati informacije iz više različitih fajlova.

Sekvencijalni pristup. – Jedan od osnovnih i najprostijih načina pristupa podacima jeste sekvencijalni pristup. Ovaj metod koristi sekvencijalnu organizaciju kod koje su slogovi fizički smješteni u rastućem ili opadajućem poretku ključnog polja. Na primjer, u fajlu za platni spisak slogovi mogu biti numerički uređeni po broju socijalnog osiguranja zaposlenog. Sekvencijalni pristup je brz i efikasan kada je riječ o velikoj količini podataka koji se procesiraju periodično. Ovakav pristup zahtijeva da svaka nova transakcija bude sačuvana u odgovarajućoj sekvenci. Takođe, većina fajlova mora biti pretražena da bi se locirao, sačuvao ili modifikovao čak i mali broj slogova. Dakle, ovaj metod je prespor za aplikacije koje zahtijevaju trenutno ažuriranje ili odgovor na zahtjev.

Direktni pristup. – Kod ovog pristupa slogovi nisu aranžirani u nekom poretku na memorijskom medijumu, već su proizvoljno raspoređeni. Stoga, DBMS mora voditi trag o memorijskoj lokaciji za svaki slog. Unos novih podataka ne zahtijeva sortiranje, pa se jednostavno i brzo realizuje. Postoji više načina pristupa podacima kod direktnog metoda. Tri najviše korišćena jesu transformacija ključa, indeksni i indeksno-sekvencijalni pristup.

Pri upotrebi metoda **transformacije ključa** vrši se aritmetičko izračunavanje nad poljem ključa jednog sloga, a dobijeni broj predstavlja adresu za memorisanje i pristup tom slogu. Proces se naziva transformacija ključa jer se na ključnom polju primjenjuje neka aritmetička operacija da bi se ono transformisalo u memorijsku adresu lokacije sloga.

Drugi metod direktnog pristupa za memorisanje i lociranje slogova koristi indeks ključeva i odgovarajućih memorijskih adresa. Novi slog se unosi na sljedeću slobodnu lokaciju, a njegov ključ i adresa se unose u indeks. Indeks je uređen po vrijednostima ključa. Slogovima se kod **indeksnog metoda** pristupa tako što se prvo pristupi indeksu da se pronađe adresa sloga, pa onda se, na osnovu te adrese, pristupa samom slogu (slika 1.77.).



Slika 1.77. Indeksni pristup podacima u glavnoj datoteci

Kod **indeksno-sekvencijalnog metoda** ISAM (Indexed Sequential Access Method) slogovi se memorišu u sekvencijalnom poretku po ključu. Dodatno, svaka tabela ima i indeks koji odgovara jednom ili više ključnih polja slogova sa njihovim memorijskim lokacijama. Slogu se pristupa direktno tako što se preko korišćenja ključnih polja pretražuje indeks i locira se adresa sloga, baš kao što mi lociramo neki pojam u knjizi pretražujući indeks pojmova. Ukoliko se procesira mali broj slogova brzo, koristi se direktni pristup putem indeksa. Međutim, u slučaju periodičnog procesiranja velikog broja slogova, koristi se sekvencijalni pristup. Na primjer, mjesečni izvještaji za kupce mogu se generisati sekvencijalnim pristupom.

1.6.6. RAZVOJ BAZA PODATAKA

Razvoj malih ličnih baza podataka, uz korišćenje DBMS-a za mikrokompjutere, relativno je jednostavan. Međutim, razvoj velikih baza podataka sa kompleksnim tipovima podataka predstavlja ozbiljan zadatak. U mnogim organizacijama razvoj i upravljanje velikim bazama podataka spada

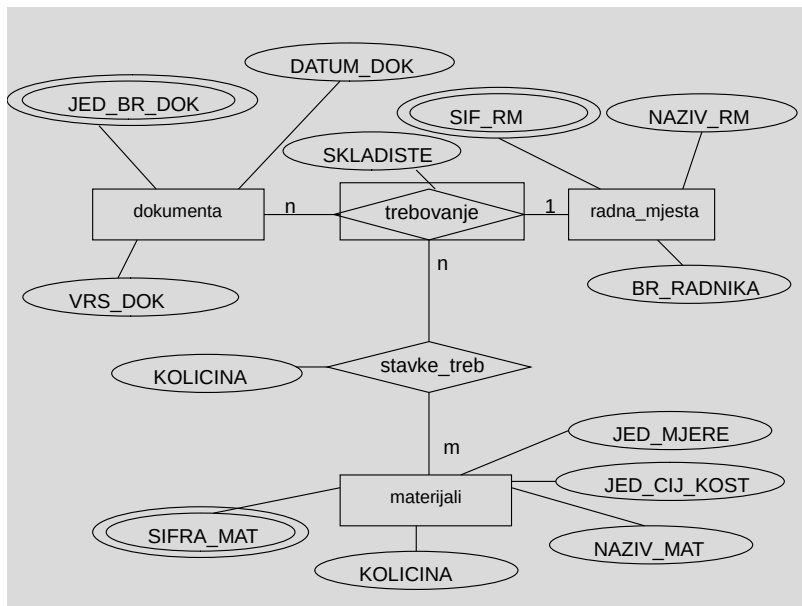
u odgovornost administratora baze podataka i analitičara dizajna baze. Oni rade sa korisnicima i analitičarima sistema na modeliranju poslovnih procesa i podataka koje ti procesi zahtijevaju. Nakon toga oni definišu odgovarajuće podatke u bazi i strukturu, tj. odnose među tim podacima.

Razvoj baze podataka započinje procesom planiranja podataka. Administratori i dizajneri baze podataka zajedno sa menadžerima razvijaju model poslovnih procesa. Oni zatim definišu informacione zahtjeve korisnika u okviru poslovnih procesa. Potom korisnici identifikuju ključne podatke koji su potrebni za obavljanje specifičnih poslovnih aktivnosti. Na osnovu ovoga razvijaju se **Entity Relationship Diagrams** (E-R dijagrami), koji modeliraju odnose između entiteta uključenih u neki poslovni proces. Entiteti se predstavljaju u pravougaonicima, njihovi atributi u elipsama, a veze između entiteta – rombovima (u duplim elipsama predstavljaju se atributi primarnog ključa). Slika 1.78. prikazuje odnose između entiteta u procesu proizvodnje. Korisnici i dizajneri baze podataka ovim E-R dijagramom specificirali su podatke potrebne za automatizaciju procesa trebovanja materijala za proizvodnju u okviru svog ERP sistema.

Korisnički pogled jeste jedan od ključnih elemenata modeliranja podataka i odnosa između njih. Na primjer, samo korisnik može odgovoriti na pitanja kao što su:

- može li jedno trebovanje da se odnosi na jedno ili više radnih mjesta (na slici 1.78. oznake n i 1 označavaju specifikaciju pravila da se trebovanje može odnositi isključivo na jedno radno mjesto);
- može li jedan dobavljač dostavljati više od jednog tipa proizvoda;
- može li jedan zaposleni primati platu u više rata;
- može li jedan zaposleni biti član više projektnih grupa itd.

Odgovori na ovakva pitanja omogućavaju identifikovanje odnosa između podataka reprezentovanih u vidu modela podataka koji podržava jedan poslovni proces. Modeli podataka predstavljaju logički dizajn baze podataka. Najzastupljeniji relacioni model predstavlja bazu podataka u vidu skupa relacija ili tabela. Relacije ili tabele se kreiraju na osnovu E-R dijagrama (slika 1.79. prikazuje logički relacioni model razvijen iz E-R dijagrama sa slike 1.78.), na osnovu kojeg se definiše fizički dizajn baze podataka i razvijaju se aplikativni programi koji podržavaju poslovne procese u organizaciji.



Slika 1.78. E-R dijagram dijela baze podataka proizvodnje

dokumenta (JED_BR_DOK, VRS_DOK, DATUM_DOK)
 radna_mj (SIF_RM, NAZIV_RM, BR_RADNIKA)
 trebovanje (JED_BR_DOK, SIF_RM, SKLADISTE)
 materijali (SIFRA_MAT, NAZIV_MAT, KOLICINA, JED_MJERE, JED_CIJ_KOST)
 stavke_treb (JED_BR_DOK, SIF_RM, SIFRA_MAT, KOLIČINA)

Slika 1.79. Relacioni model dijela baze podataka proizvodnje

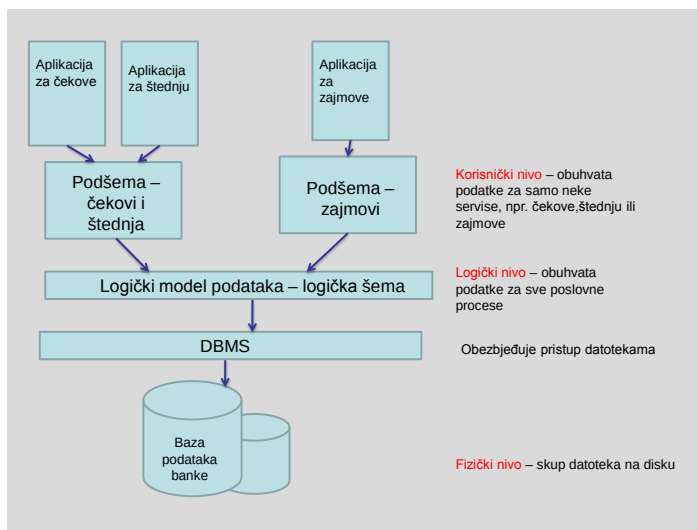
Dakle, koraci, odnosno faze u razvoju baze podataka jesu:

- modeliranje poslovnih procesa, tj. njihova dekompozicija do elementarnih, atomskih procesa;
- identifikovanje podataka za sve atomske poslovne procese;

- razvoj konceptualnog modela podataka, npr. E-R dijagrama;
- transformisanje konceptualnog modela u logički, npr. relacioni model;
- konfiguracija i generisanje fizičkog modela na osnovu logičkog (ova faza podrazumijeva generisanje indeksa i datoteka);
- kreiranje programskih aplikacija za rad sa bazom podataka.

Današnji DBMS-ovi imaju CASE alate koji omogućavaju automatsku podršku za svih 6 faza razvoja baze. To su alati sa grafičkim korisničkim okruženjem, koje omogućava jednostavno kreiranje modela procesa i konceptualnih modela podataka (E-R dijagrama). Kreirane specifikacije čuvaju se u repozitorijumu podataka. Na osnovu ovakvih logičkih specifikacija CASE alati automatski generišu iz E-R dijagrama logički i fizički model podataka. Iz modela poslovnih procesa automatski se takođe mogu generisati i aplikacije za bazu. Na slici 1.68. prikazan je *oracle designer*, CASE alat u okviru ORACLE DBMS-a.

Model podataka naziva se još i šema podataka, i predstavlja cjelokupan logički pogled na odnose među podacima u bazi podataka. Podšema baze jeste korisnički logički pogled na dio podataka i odnosa među njima, koji koristi jedna specifična korisnička aplikacija. Fizički dizajn baze predstavlja fizički pogled na podatke, a on opisuje kako će podaci biti fizički smješteni na memorijskim uređajima kompjuterskog sistema. Tako slika 1.80. prikazuje ove različite poglede na primjeru baze podataka banke. U ovom primjeru čekovi, štednja i krediti jesu poslovni procesi čiji su modeli podataka dio cjelokupnog modela podataka banke i predstavljaju korisnički logički pogled na podatke ili podšeme. Cjelokupni model podataka podržava sve poslovne procese banke i predstavlja logički pogled na bazu, tj. šemu baze. Osim šeme i podšeme baze, tu je i fizički pogled na bazu. Interfejs između logičkog i fizičkog nivoa obezbjeđuje DBMS.



Slika 1.80. Fizički, logički i korisnički pogled na podatke

1.7. PITANJA I ZADACI ZA VJEŽBU

1. Zašto su IS važni?
2. Koja su znanja o IS bitna za menadžment i poslovanje?
3. Objasnite osnovne koncepte sistema, kao i osobine otvorenosti i adaptivnosti.
4. Definišite IS i objasnite njegovu ulogu u poslovnom sistemu.
5. Koje su osnovne komponente (resursi) i aktivnosti IS?
6. Univerzitetski informacijski sistem studentima koji ga koriste pruža mogućnosti obavljanja velikog broja različitih zadataka. Na primjer, student može na računaru (desktop-u ili laptop-u) u nekoj od kompjuterskih sala da otkuca svoj seminarski rad. Pri tome koristi neki tekst procesor (na primjer MS WORD), koji je memorisan na hard disku tog mikrokompjutera. Kada je seminarski rad otkucan i na odgovarajući način formatiran, student ga može memorisati na hard disku dotičnog mikrokompjutera, zatim na floppy disku i putem e-maila preko interneta poslati kopiju fajla predavaču. Fajl zatim može odštampati na printeru koji je povezan sa njegovim računarom. U slučaju da je student za ime fajla koristio naziv pod kojim već postoji neki dokument na njegovom kompjuteru, program će prikazati upozorenje i čekati dok ne primi odgovarajuću komandu.
 - Identifikujte komponente ovog IS, tj. ljude, hardver, softver, mrežu, podatke i informacije koje su navedene u primjeru.
 - Identifikujte aktivnosti ovog IS, tj. input, procesiranje, output, memorisanje i kontrolne aktivnosti koje su navedene u primjeru.
7. Preduzeće za prodaju kancelarijskih proizvoda prima više od 10000 narudžbi mjesečno za više od 1000 različitih vrsta kancelarijskih proizvoda koje posjeduje u skladištu. Oko 60 PC računara je povezano, u okviru lokalne mreže ovog preduzeća, sa nekoliko servera. Narudžbe koje se primaju telefonom ili poštom operateri unose na umreženim kompjuterima. Narudžbe takođe mogu biti unesene direktno od strane kupca preko e-commerce veb-sajta razvijenog od strane ovog preduzeća. Unos narudžbi se obavlja pomoću specijalno formatizovanih ekrana koji pomažu korisniku da ispoštuje procedure unosa podataka. Uneseni podaci se memorišu na magnetnim diskovima nekog od servera.

Kada je narudžba unesena, server provjerava da li na zalihama ima naručenog proizvoda, ažurira bazu kupaca i proizvoda koja je memorisana na njegovim magnetnim diskovima, a zatim šalje spisak naručenih proizvoda na štampač u skladištu. Na osnovu ovog spiska personal skladišta će isporučiti narudžbu kupcu. Direktor i menadžment preduzeća imaju umrežene PC računare u svojim kancelarijama. Koristeći jednostavne upitne komande bazi podataka, oni dobijaju odgovore i izvještaje o narudžbama, kupcima, robi, trendovima u zahtjevima za pojedinim proizvodima i drugo.

- Identifikujte ljude, opremu, programe, podatke i informacije u ovom IS.
 - Identifikujte aktivnosti koje se obavljaju u okviru ovog IS.
8. Pored svake stavke upišite koja vrsta IS (TPS, MIS, DSS, ES, EIS) podržava odgovarajuću situaciju:
- potrebno je generisati pregled svih nabavki materijala *šećer u kristalu* u posljednjih mjesec dana;
 - odjel za odnose sa korisnicima usluga želi da zna šta je uticalo na to da je jedan broj korisnika u posljednja 3 mjeseca otkazao njihove usluge;
 - isti taj odjel želi da ima pregled svih korisnika po geografskim lokacijama, imovinskom stanju i polu;
 - uprava računovodstva traži izvještaj o promjenama na kontu Kupci koje su napravljene na dan 12. 9. 2004. godine;
 - menadžment projektuje poslovanje preduzeća za narednih 5 godina i potrebni su mu bilansi stanja i uspjeha iz prethodnih 5 godina poslovanja;
 - menadžment želi da u trenutku nastanka sazna informaciju o procentu smanjenja prihoda u odnosu na prethodni mese, kao i o eventualnoj pojavi deficitnog bilansa;
 - firma se nalazi pred odlukom o prijemu novog menadžera marketinga i ima sve potrebne karakteristike prijavljenih kandidata, kao i kriterijume za ocjenjivanje tih karakteristika, samo je potrebno sastaviti rang-listu alternativa.
9. Jedno preduzeće koje ima 20 zaposlenih treba da formira svoju LAN mrežu. Svakom zaposlenom treba omogućiti da pristupa centralnoj bazi

podataka (da unosi, ažurira i pretražuje). Svaki zaposleni treba da ima mogućnost izrade i štampanja dokumenata, njihovog čuvanja na centralnom mjestu, ali i mogućnost posebnog čuvanja kopija dokumenata koje je on izradio. Preduzeće obavlja u prosjeku 200 poslovnih transakcija dnevno. Koliko i koje tipove računara bi preduzeće trebalo da nabavi?

10. Kako se mijenja situacija ako

- zaposleni ne moraju praviti sopstvene rezervne kopije dokumenata;
 - preduzeće želi da ima data warehouse, data mining i OLAP alate;
 - preduzeće želi da ima intranet mrežu i elektronsku trgovinu;
 - je riječ o korporaciji koja ima 20000 zaposlenih i odjele locirane širom svijeta, a obrađuje nekoliko miliona poslovnih transakcija dnevno;
 - preduzeće ima 10 mobilnih zaposlenih koji treba u svakom trenutku da imaju mogućnost kontakta sa ostalim zaposlenima, kao i mogućnost pristupa kompanijskim podacima i informacijama;
 - preduzeće koristi data mining koji zahtijeva oko jedan trilion operacija u sekundi.
11. Program je niz instrukcija koje se zadaju računaru. Definišite jedan jednostavni program i objasnite kako se on izvršava na računaru, tj. kakva je uloga ulaznih uređaja, CPU, primarne memorije, sekundarne memorije i izlaznih uređaja u tom procesu.
12. Računarski sistem obrađuje instrukciju za 10 nanosekundi. Kolika je njegova brzina izražena u MIPS?
13. Jedna kompanija ima jedan centralni server i 20 radnih stanica u centralnoj zgradi. Poslovnice koje se nalaze na dvije različite lokacije sa centralnim serverom su povezane standardnom iznajmljenom telefonskom linijom. Svaka poslovnica ima svoj server i po 7 radnih stanica. Na centralnim serverima je baza podataka i mrežni operativni sistem. Na svakoj radnoj stanici se nalazi instaliran softver za rad sa bazom podataka, koji se na njima i izvršava. Na radnim stanicama zaposleni izrađuju standardna dokumenta i skladište ih u bazi dokumenata, koja se nalazi takođe na centralnom serveru. Jedna radna stanica u centralnoj zgradi je modemska (dial-up) povezana na internet.

- Koliko LAN-ova prepoznajete u ovom primjeru?
 - Koji je tip mreže u pitanju?
 - Šta bi trebalo promijeniti u ovoj mreži pa da ona bude troslojna klijent/server mreža?
 - Da li je ovo primjer jedne intranet mreže?
 - Šta bi trebalo promijeniti pa da mreža bude intranet?
14. Kompanija ima dva servera i 20 radnih stanica u centralnoj zgradi. Na jednom serveru se nalazi baza podataka, dok je drugi veb-server. Radne stanice putem browsera pristupaju aplikacijama i multimedijalnim informacijama koje su smještene na veb-serveru. Aplikacije se nalaze i izvršavaju na veb-serveru. Kompanija ima 10 poslovnica koje se nalaze na različitim lokacijama. One veb-aplikacijama i podacima pristupaju putem interneta. Takođe poslovni partneri mogu pristupiti aplikacijama i podacima putem interneta. Bilo koji član radnog tima može sa bilo koje lokacije komunicirati sa svakim zaposlenim i pristupiti potrebnim informacijama i podacima putem interneta.
- Da li je kompanijska mreža intranet?
 - Intranet se sastoji od 11 _____.
 - Da li u ovom primjeru prepoznajete neki ekstranet?
 - Da li je riječ o VPN tipu mreže i zašto?
 - Šta mora ovaj intranet da ima u pogledu sigurnosti?
15. Kompanija za grafički dizajn ima 15 zaposlenih koji rade dizajn stranica i ilustracije za izdavače časopisa i knjiga koji se nalaze na različitim lokacijama. Svakodnevno se šalju fajlovi sa ilustracijama i dizajnom klijentima na pregledanje. Prosječna veličina fajla koji se šalje jeste 4MB, a šalje se u prosjeku 25 fajlova dnevno. Kompanija koristi iznajmljenu telefonsku liniju i softver koji omogućava da 20 korisnika koriste internet. Cilj menadžmenta je da izabere najpovoljniji mrežni servis. Ako su dostupni sljedeći servisi sa sljedećim karakteristikama:

Servis	Brzina prenosa	Troškovi
Dial-up servis sa 56KBPS analognim modemima za svakog	56KBPS	40 eura mjesečno za internet usluge + 35 eura mjesečno troškova telefona

zaposlenog		
ISDN linija	128 KBPS	100 eura mjesečno + 300 eura cijene instalacije
SDSL	512KBPS	100 eura za DSL modem + 175 eura mjesečni trošak
T1 linija	1.5 MBPS	1200 eura mjesečno

- Koja je prosječna količina vremena koja se dnevno troši na transmisiju fajlova za svaki od datih servisa?
 - Koji je od datih servisa najpovoljniji za ovu kompaniju? Zašto?
 - Ukoliko se kompanija proširi na 60 zaposlenih i u prosjeku 100 transmisija fajlova dnevno, koji servis bi u tom slučaju bio najpovoljniji?
16. Na strani veb-sajta jedne muzičke grupe nalazi se logo muzičke izdavačke kuće, spisak od 12 albuma, fotografije sva četiri člana grupe, video snimak sa najposjećenijeg koncerta i naziv njihove najpopularnije pjesme. Klikom na logo izdavačke kuće prelazimo na njen veb-sajt. Klikom na neki od albuma prelazimo na veb-stranu koja prikazuje pjesme sa tog albuma i ostale informacije o njemu. Klikom na fotografiju nekog člana grupe u browser prozoru pojavljuje se njegova biografija, a klikom na naziv pjesme može se odslušati ova muzička numera. Da li je ovaj dokument hypertext ili hypermedia? Da li na ovoj strani ima hiperlinkova i koliko? Ovaj dokument je _____ formata.
17. Kompanija želi da nabavi softver za obavljanje sljedećih zadataka:
- surfovanje webom, intranetom i ekstranetom;
 - slanje poruka između radnih stanica;
 - pomoć zaposlenima prilikom timskog rada;
 - pomoć prodajnim agentima da bilježe vrijeme sastanaka i poziva u vezi sa prodajom;
 - kucanje dopisa i izvještaja;
 - analiza prodaje (u tabelarnom formatu) i grafičko prikazivanje;
 - izrada različitih grafičkih prezentacija i njihovo publikovanje na intranetu.

Koji su softverski paketi potrebni ovoj kompaniji?

18. Koji od podsistema računovodstvenog informacionog sistema podržava svaku od sljedećih aktivnosti:
- unos podataka o narudžbi kupca i generisanje otpremnice;
 - evidentiranje promjena na zalihama prema otpremnici;
 - knjiženje promjene na zalihama prema otpremnici u glavnoj knjizi;
 - generisanje fakture za kupca;
 - knjiženje podataka o prodaji u glavnoj knjizi;
 - isplata poreza na promet preko žiro-računa;
 - naplata potraživanja od kupca prema izvodu banke;
 - unos podataka o nabavci (podaci sa ulazne fakture dobavljača);
 - generisanje prijemnice za zalihe;
 - knjiženje podataka o nabavci u glavnoj knjizi;
 - evidentiranje promjena na zalihama prema prijemnici;
 - isplata troškova prevoza prilikom nabavke čekom;
 - isplata dugovanja dobavljaču prema virmanskom nalogu;
 - generisanje periodičnih bilansa stanja i uspjeha i
 - zaključna knjiženja.
19. Intranet jedne kompanije treba da ima 12 radnih stanica, veb-server i server baze podataka. Svaka radna stanica pomoću browsera treba da koristi interaktivne, real-time veb-aplikacije koje se izvršavaju na veb-serveru. Veb-server je povezan sa dva štampača, koje treba da koriste sve radne stanice zajedno. Da li je dovoljno da organizacija nabavi operativni sistem sa mogućnošću multitasking-a? Zašto? A multiprogramiranja? Zašto? Da li je organizaciji potreban OS sa mogućnošću time sharinga? Zašto? Kako se još zovu OS koji imaju mogućnost time sharinga? U čemu je razlika između multiprogramiranja i multiprocesiranja?
20. Dio poslovnog informacionog sistema jedne firme treba da real-time procesira oko 5000 transakcija u sekundi. Imate ponude za realizaciju sistema pomoću veb-orijentisanog jezika Java, objektnih jezika Visual C++ i Visual Basic, kao i 4GL jezika. Koju od ponuda ćete izabrati? Zašto?
21. Vaša kompanija je u posljednje dvije godine porasla sa 40 na 200 zaposlenih. Potrebno je da nabavite programski sistem za ljudske resurse kako biste automatizovali funkcije u ovom domenu. Imate na

raspolaganju dvije opcije. Jedna je da nabavite klijent/server programski paket koji će se izvršavati na kompanijskom midrange kompjuteru. Druga opcija je da koristite usluge aplikativnog servis provajdera koji omogućava izvršavanje softvera za ljudske resurse preko weba. Departman za ljudske resurse već ima PC računare sa web browserima i pristupom internetu. Osoblje informacionog sistema posjeduje dvojicu specijalista.

Najbolja ponuda programskog paketa za ljudske resurse ima cijenu od 9500 eura. Jedan IS specijalista sa godišnjom platom od 65000 eura treba da potroši po 4 sata svake od 40 radnih nedjelja održavajući ovaj program i prilagođavajući nove verzije (upgrade). Nakon godinu dana otkada je softver nabavljen, upgrade se plaća isporučiocu paketa u iznosu od 1000 eura godišnje

Aplikativni servis provajder zahtijeva 1500 eura za inicijalni set-up sistema i 5 eura mjesečno po svakom zaposlenom u firmi. Nema potreba za novim nabavkama hardvera, niti su predviđeni troškovi održavanja softvera jer to obavlja sam provajder.

- Koliki su troškovi svake od ovih opcija u prvoj godini?
- Koja je opcija manje skupa u periodu od tri naredne godine?
- Koju opciju biste vi izabrali? Zašto? Koje sve činjenice koristite pri donošenju odluke? Koji su rizici svake od ovih opcija?

22. Zaokružite u tabeli jedan karakter, polje, slog, fajl, atribut, entitet, skup entiteta.

Broj indeksa	Ime studenta	Adresa	Telefon
22/04	Marković Jovan	Podgorica	2222
97/04	Adžić Ana	Bijelo Polje	3333
111/04	Burić Igor	Bar	5555

23. Fakultetska studentska služba čuva podatke o studentima i ispitima u odvojenim fajlovima na svom PC računaru. Odjel za plate zaposlenih na fakultetu takođe ima svoj PC i svoju nezavisnu kolekciju fajlova. Uprava je odlučila da profesorima u sklopu plate isplati povišicu zavisno od broja studenata koje su ispitali u toku mjeseca. Može li se u odjelu za plate izračunati iznos povišice za profesore? Zašto? Da li ovdje ima

redundance podataka? Može li aplikacija iz odjela za plate pristupiti fajlu o ispitima? Može li studentska služba odmah obezbijediti izvještaj o profesorima i broju obavljenih ispitivanja? Zašto? U slučaju da se format broja indeksa promijenio, tako da sada više nije tekstualni podatak maksimalne dužine 6 već dužine 8, može li studentska služba proći bez izmjene programa? Šta bi trebalo uraditi pa da sve ovo bude moguće?

24. Tri firme koje čine konzorcijum, imaju svoje operativne baze podataka. Jedna ima *oracle* DBMS, a ostale dvije SQL server DBMS. Firme su povezane ekstranet mrežom, baziranom na internet osiguranim linkovima. Strateški menadžment konzorcijuma želi da analizira sumarne poslovne podatke za sve 3 firme kroz više različitih dimenzija. Prilikom analize želi da ima aktuelne ali i istorijske podatke. Konzorcijum je odlučio da za tu svrhu koristi server firme koja ima *oracle* DBMS, kao i ovaj sistem za upravljanje podacima. Za OLAP analizu koristiće *oracle discoverer* softver iz Oracle-ovog *business intelligence* paketa. Šta konzorcijum mora uraditi sa podacima? Strateški menadžment bi takođe želio da automatski otkriva faktore koji najviše utiču na količinu prodate robe, kao i karakteristike kupaca čiju su lojalnost izgubili, tj. želio bi utvrditi zašto im kupci otkazuju lojalnost. Šta se mora nabaviti da bi ovo bilo moguće?
25. Jedna avio-kompanija sa više širom svijeta dislociranih dispečera letova treba da dizajnira svoju bazu podataka. Svaki udaljeni dispečer ima sopstvenu LAN mrežu sa serverom. Intranet ove kompanije baziran je na osiguranim internet linkovima. Da li je za ovu kompaniju dobra centralizovana baza podataka? Zašto? Kod distribuiranog koncepta kompanija ima na raspolaganju dvije opcije.
- Centralna baza će se podijeliti i svaki dislocirani server imaće svoj dio podataka sa kojima radi. Promjene na lokalnim fajlovima noću će biti usklađene sa centralnom bazom na batch osnovi.
 - Kompletne centralne baze će biti multiplicirane (duplicirane) na svim izmještenim lokacijama. Svaka promjena napravljena u Frankfurtu, automatski će biti reduplicirana u Njujorku ili Hong Kongu. Ova strategija zahtijeva ažuriranje centralne baze na manje od jednog sata.

Koja je od ove dvije opcije bolja za datu avio-kompaniju? Od čega to zavisi?

26. Nacrtajte E-R dijagram dijela baze podataka fakulteta koji sadrži podatke o predmetima i profesorima koji ih predaju. Dopunite dijagram tako da sadrži podatke o studentima i njihovim ocjenama iz pojedinih predmeta. Kako se predstavljaju datoteke kod relacione strukture baze i kako se slogovi povezuju? Od čega to zavisi? Predstavite tabelama datoteke Student, Predmet i Student_Predmet.

2. RELACIONE BAZE PODATAKA

2.1. POJAM BAZE PODATAKA

Skup sačinjen od podataka kojim se modelira stanje nekog sistema i mehanizama za manipulisanje tim podacima – naziva se bazom podataka.

Baza podataka se sastoji iz dvije komponente:

- sistema za upravljanje bazama podataka (DBMS) i
- baze zapisa.

DBMS jeste skup programa koji korisniku omogućavaju da kreira i održava bazu zapisa i da postavlja upite bazi, tj. da pristupa odgovarajućim podacima i da daje odgovore. U tom se smislu DBMS može shvatiti kao interpretator korisničkih zahtjeva. Uz to, on korisniku omogućava da se koncentriše na to šta treba uraditi, a ne i kako treba uraditi.

Baza zapisa predstavlja kolekciju samih podataka permanentno smještenih na sekundarnoj memoriji.

Namjena baze podataka zavisi od informacije koja se u njoj čuva, pa tako npr. imamo bazu podataka fakulteta, banke i sl.

2.2. FIZIČKI, KONCEPTUALNI I KORISNIČKI NIVO APSTRAKCIJE PODATAKA

Apstrakcija podataka jeste mehanizam koji obezbjeđuje korisnicima jedan apstraktan pogled na bazu podataka, pri čemu skriva dosta kompleksne strukture za smještaj podataka u bazi. Činjenica da se korisnici baza podataka znatno razlikuju po znanju iz ove oblasti zahtijeva postojanje nekoliko nivoa apstrakcije podataka.

Fizički nivo, koji predstavlja najniži nivo apstrakcije i koji u stvari definiše način smještanja podataka na sekundarnoj memoriji, opisuje veoma kompleksne strukture za smještaj i pristup podacima. Njime se bave uglavnom **sistemske programeri** i proizvođači DBMS-a.

Konceptualni nivo jeste sljedeći, viši nivo apstrakcije koji opisuje kako će podaci biti smješteni u bazi. On to čini skupom relativno jednostavnih struktura iza kojih se kriju kompleksne strukture fizičkog nivoa, o kome korisnici konceptualnog nivoa ne moraju znati ništa. Konceptualni nivo apstrakcije koriste **projektanti** i **administratori** baze podataka.

Na kraju, treću, veoma važnu klasu korisnika predstavljaju **krajnji, tj. naivni korisnici**, kojih ima najviše, a nivo njihovog znanja je, po pravilu, mali. Zato je poželjno da se takvim korisnicima podaci predstave u obliku struktura koje što vjernije odgovaraju objektima realnog sistema. Sa stanovišta bezbjednosti podataka, potrebno je omogućiti da raznim korisnicima mogu biti dostupne različite strukture sa sličnim ili različitim sadržajem. Drugim riječima, za svaku grupu korisnika potrebno je što više olakšati pristup podacima kojima oni smiju pristupati, dok im ostale treba učiniti nedostupnim. Dakle, izdvaja se treći nivo apstrakcije podataka, tzv. **korisnički ili eksterni nivo**. Kako se skup struktura podataka koji odgovaraju jednoj klasi korisnika naziva pogledom, to se ovaj nivo često zove i nivo pogleda. Pri tom, razne klase krajnjih korisnika zahtijevaju različite poglede.

2.3. OSNOVNI POJMOVI O MODELU PODATAKA

Na konceptualnom nivou, da bi se mogao definisati sadržaj baze podataka, kao i način manipulisanja podacima, potrebna su nam odgovarajuća konceptualna sredstva stavljena na raspolaganje projektantu baze podataka. Skup tih konceptualnih sredstava radi opisivanja podataka njihovih odnosa, integritetskih ograničenja, kao i apstraktnih operacija u cilju postavljanja upita bazi naziva se model podataka. Svaki DBMS zasnovan je na nekom modelu. Međutim, model sam po sebi ne zavisi od samog DBMS-a, što znači da je on nešto generalnije, uopštenije.

Ovu razliku između modela podataka, s jedne strane, i DBMS-a, s druge strane, treba do kraja izvesti. Naime, DBMS je konkretan softver koji podržava rad sa bazom u cilju definisanja baze i postavljanja upita. To sa svoje strane podrazumijeva prisustvo računarskog sistema i samog softvera za upravljanje bazom podataka. S druge strane, putem modela podataka projektant baze podataka i korisnici prenose svoje ideje na papir, i to:

- informacije o podacima i njihovim odnosima;

- način predstavljanja tih informacija putem osnovnih **informativnih struktura** modela (na primjer: osnovna informaciona struktura relacionog modela jeste relacija);
- način postavljanja upita bazi putem apstraktnih operatora modela (u relacionom modelu to su operatori, selekcije, projekcije, spajanja, itd.).

U skladu sa tim, model podataka treba da sadrži tri ključna elementa:

1. jedinstven način grafičkog predstavljanja sadržaja baze;
2. jasno definisane osnovne strukture informacionog modela;
3. jasno definisane apstraktne operatore modela koji služe za postavljanje upita bazi.

Međutim, različiti modeli postavljaju različite akcente na ova tri aspekta. Na primjer, u relacionom modelu razrađeni su aspekti 2 i 3, dok je u potpunosti zanemaren aspekt 1. Na sličan način mrežni model se uglavnom bavi aspektom 2, dok u priličnoj mjeri zanemaruje aspekte 1 i 3.

Nas će u početku najviše interesovati aspekti 1 i 2, a baš ti aspekti su naglašeni u modelu entiteti–odnosi (E-R Entity-Relationship model). Predložio ga je Peter Chen 1976. godine. Ovaj model se koristi za grafičko prezentiranje podataka u relacionom modelu podataka (prva komponenta modela podataka koju relacioni model nema).

Relacioni model je danas najviše primjenjivan model podataka. Većina poznatih DBMS sistema zasnovana je na relacionom modelu podataka (ORACLE, SQL SERVER, IBM DB2, MS ACCES itd.). Kao glavne prednosti ovog modela navode se jednostavnost strukture, što omogućava primjenu matematičkih formalizama u postupku projektovanja na konceptualnom nivou, kao i lak interaktivni pristup uz pomoć nekih od poznatih upitnih jezika (kao što je SQL).

2.4. MODEL ENTITETI–ODNOSI (E-R MODEL)

Osnovni koncepti E-R modela jesu skupovi entiteta, atributi, skupovi odnosa, kardinalni broj odnosa, egzistencijalne zavisnosti, primarni ključevi i jaki i slabi skupovi entiteta. Grafički se predstavljaju putem E-R dijagrama.

2.4.1. ENTITETI I SKUPOVI ENTITETA

Entitet je svaki objekat koji se može razlikovati od svih ostalih objekata u nekom univerzumu, što znači da se može jednoznačno identifikovati. Npr. određena osoba, događaj, firma itd. jesu različiti entiteti. Entitet može biti nešto konkretno (kao što je određeni čovjek ili određena knjiga) ili nešto apstraktno (kao istorijska kategorija, društveno uređenje, dio neke misaone klasifikacije). Jedino što takav objekat treba da zadovolji da bi bio entitet jeste da se može jednostavno identifikovati u određenom svijetu koji se modelira.

Grupa entiteta istog tipa naziva se skupom entiteta. Npr. skup studenata univerziteta predstavlja skup entiteta. Ovi skupovi entiteta imaju neka pridružena svojstva ili attribute putem kojih se može testirati da li neki entitet pripada datom skupu entiteta. Međutim, skupovi entiteta ne moraju biti potpuno razdvojeni u smislu njihovih svojstava. Npr. ako imamo 2 skupa entiteta *zaposleni* i *čovjek*, onda sve osobine (atributi) skupa entiteta *čovjek* jesu atributi iz skupa entiteta *zaposleni*.

2.4.2. ATRIBUTI

Atributi su svojstva entiteta i njihovih skupova. Svaki skup entiteta je u potpunosti određen skupovima atributa kojima pripadaju. Npr. skup entiteta *zaposleni* može da ima attribute: IME, PREZIME, BR. L. K, STAŽ. Za svaki atribut postoji skup dozvoljenih vrijednosti koji se naziva domenom atributa. Formalno gledano, atribut predstavlja funkciju koja preslikava skup entiteta u odgovarajući domen. Npr. ako je A_j atribut odgovarajućeg skupa entiteta E_i sa domenom D_j , onda se A_j može definisati kao funkcija $A_j: E_i \rightarrow D_j$. U skladu sa tim možemo da pišemo $A_j(ek) = v_i$, gdje ek pripada E_i , a v_i pripada D_j . Npr. ako je ek entitet iz skupa entiteta *zaposleni*, tj. $ek = \langle \text{Petar Petrović, 11123, 15} \rangle$, tada možemo pisati:

IME (ek) = Petar Petrović

BR. L. K. (ek) = 11123

STAŽ (ek) = 15.

Napomenimo da se iz ove definicije da zaključiti da svaki atribut sadrži atomsku vrijednost.

2.4.3. ODNOSI I SKUPOVI ODNOSA

Odnos je neka vrsta asocijacije ili veze između dva ili više entiteta. Pri tom, entiteti ne moraju pripadati različitim skupovima entiteta. Npr. odnos *otac–sin* definiše odnos između dva entiteta istog skupa *čovjek*.

Pod skupom odnosa podrazumijevamo više odnosa istog tipa. U suštini to je matematička relacija nad dva ili više skupova entiteta, pri čemu skupovi entiteta ne moraju biti različiti. Drugim riječima, skup odnosa R jeste podskup Dekartovog proizvoda skupova entiteta, odnosno:

$$R = \{ \langle e_{i1}, e_{i2}, \dots, e_{ik} \rangle \mid e_{i1} \in E_{j1}, \dots, e_{ik} \in E_{jk} \},$$

gdje su $e_{i1}, \dots, e_{ik}$ pojedinačni entiteti, a $E_{j1}, \dots, E_{jk}$ – odgovarajući skupovi entiteta. Već smo rekli da skupovi entiteta ne moraju biti različiti sa aspekta skupova odnosa. Tako npr. možemo imati sljedeći skup odnosa:

$$R = \{ \langle e_{i1}, e_{i2} \rangle \mid e_{i1} \in \text{čovjek}, e_{i2} \in \text{čovjek} \},$$

gdje su oba entiteta iz istog skupa entiteta *čovjek* (već pomenuti odnos *otac–sin*).

Putem E-R modela najlakše se izražavaju binarni odnosi, dok se proizvoljni n -arni odnosi teže izražavaju.

Kao i skupovi entiteta, tako i skupovi odnosa mogu imati svoje atribute. Npr. atribut skupa odnosa *student–predmet* može biti OCJENA.

Razlikujemo više tipova odnosa koji su određeni tzv. **kardinalnim brojem** odnosa. Svaki binarni skup odnosa može imati i jedan od sljedećih tipova:

- odnos jedan prema jedan (1:1), gdje nekom entitetu iz skupa entiteta E_i može odgovarati tačno jedan entitet iz skupa entiteta E_j , i obratno;
- odnos jedan prema mnogo (1:n), tj. odnos kod koga nekom entitetu iz skupa entiteta E_i može odgovarati više entiteta iz skupa E_j , ali nekom entitetu iz skupa E_j može odgovarati tačno jedan entitet iz skupa E_i ;

- odnos mnogo prema jedan ($n:1$) inverzan je prethodnom.
- odnos mnogo prema mnogo ($n:n$), u kome nekom entitetu iz skupa E_i može odgovarati više entiteta iz skupa E_j i obratno.

Npr. odnos *brak* između skupova entiteta *muškarac* i *žena* je odnos tipa $1:1$, jer svaki muškarac može imati samo jednu ženu i obratno. Odnos *otac–sin* je odnos tipa $1:n$, jer otac može imati više sinova, a svaki sin ima samo jednog oca. Skup odnosa *student–predmet* jeste skup tipa $n:m$, jer jedan student sluša više predmeta, jedan predmet pohađa više studenata.

Kardinalni broj odnosa je jedno integritetsko ograničenje, koje specifikuje projektant baze podataka i to u skladu sa stvarnošću koja se modelira. Greške projektanta prilikom specifikacije kardinalnog broja mogu dovesti do:

- povećanja dupliranosti podataka i nepotrebnog uslovljavanja i
- nametanja ograničenja koja ne odgovaraju stvarnosti, a time i nemogućnosti predstavljanja određene informacije.

2.4.4. EGZISTENCIJALNA ZAVISNOST

Pored kardinalnog broja odnosa, egzistencijalne zavisnosti jesu druga značajna vrsta integritetskih ograničenja, koja se mogu specifikovati putem E-R modela.

Ako postojanje nekog entiteta X u bazi zavisi od prisutnosti nekog entiteta Y u bazi, onda kažemo da je X egzistencijalno zavistan od Y . U operacionom smislu to znači da ako se izbriše Y iz baze, onda se mora izbrisati i X . U ovakvom odnosu entitet Y jeste dominantni entitet, a X podređeni entitet. Na istom principu zasnivaju se i dominantni i podređeni skupovi entiteta. Npr. u nekom preduzeću se vodi evidencija o izdržavanim članovima porodice zaposlenih u tom preduzeću. Ako izbrišemo zaposlenog, onda moramo izbrisati i članove njegove porodice.

2.4.5. PRIMARNI KLJUČEVI

Jedan od najvažnijih zadataka sa aspekta modeliranja baze podataka jeste da se odredi kako se entiteti, a isto tako i odnosi, međusobno razlikuju unutar skupova entiteta i skupova odnosa. Uopšteno gledano, entiteti su po

definiciji različiti jedni od drugih. Međutim, sa tačke gledišta DBMS, te razlike se moraju izraziti putem njihovih atributa, ili tačnije putem vrijednosti koje ti atributi čuvaju unutar različitih entiteta. Zato se insistira da za svaki skup entiteta postoji neki podskup njegovih atributa, putem kojih se jednoznačno može odrediti svaki entitet unutar tog skupa. Taj podskup atributa putem koga se jednoznačno mogu odrediti entiteti u skupu entiteta naziva se **natključ** (super ključ) tog skupa entiteta. Npr. za skup entiteta zaposleni koji imaju attribute IME, BR. L. K, STAŽ, natključ može biti BR. L. K.

Međutim, natključ može biti i IME ako sa sigurnošću možemo tvrditi da svi zaposleni imaju različita imena.

Formalno gledano, natključ se može definisati kao grupa atributa datog skupa entiteta koja omogućava jednoznačno preslikavanje entiteta iz datog skupa na Dekartov proizvod domena tih atributa, odnosno:

$$NK(E_i) = (A_{i1}, A_{i2}, \dots, A_{ik}) = f_i : E_i \rightarrow D_{i1} \times D_{i2} \times \dots \times D_{ik}.$$

Sam pojam natključa ne podrazumijeva minimum broja atributa koji ga specificiraju. Npr. u datom skupu entiteta *zaposleni* natključ je BR. L. K, ali isto tako i skup atributa BR. L. K, IME. Uopšteno govoreći, ako je X natključ, onda je i svaki nadskup atributa iz X, tj. skup atributa koji kao pravi podskup sadrži X, takođe natključ.

Koncept natključa nije dovoljan za naše potrebe. Nas interesuje onaj najmanji natključ koji ne sadrži suvišne attribute. Takav natključ naziva se **kandidatski ključ**. Prema tome, kandidatski ključ K jeste skup atributa:

1. koji omogućava jednoznačno preslikavanje sa skupa atributa na Dekartov proizvod domena tih atributa, odnosno: $\forall e_i, e_j \in E_i, K(e_i) \neq K(e_j)$
2. u kome nijedan pravi podskup tog skupa atributa koje sačinjavaju kandidatski ključ nema tu osobinu, tj. ako je $K' \subset K$, onda postoje $e_i, e_j \in E_i$, takvi da $K'(e_i) = K'(e_j)$.

Sam pojam kandidatskog ključa podrazumijeva da je, kao i kod natključa, sposobnost jednoznačnog identifikovanja putem njega vremenski nepromjenljivo svojstvo, tj. bez obzira na trenutni sadržaj skupa entiteta E_i ,

ovo svojstvo mora važiti. Npr. BR. L. K. jeste kandidatski ključ skupa entiteta *zaposleni* zato što zadovoljava svojstva 1 i 2 kandidatskog ključa.

Kao što smo vidjeli, postoji mogućnost da jedan skup entiteta ima više kandidatskih ključeva (BR. L. K. i IME su dva kandidatska ključa podskupa *zaposleni*).

Međutim, nas interesuje samo jedan od mogućih kandidatskih ključeva koji bi služio kao primarni ključ za dati skup entiteta. Iz više kandidatskih ključeva za dati skup entiteta projektant baze podataka bira onaj koji je po svojoj semantici, udobnosti ili sasvim proizvoljno bolji da bude **primarni ključ**. Na primjer, u tom smislu za dati skup entiteta *zaposleni* možemo izabrati BR. L. K. Znači, primarni ključ nekog skupa entiteta u oznaci PK(Ei) jeste kandidatski ključ koji je administrator baze podataka po svojoj volji izabrao kao glavno sredstvo za jednoznačno identifikovanje entiteta u tom skupu.

Kod nekih skupova atributa ne postoji dovoljan broj atributa koji bi mogli da sačinjavaju primarni ključ. Takvi skupovi nazivaju se **slabim skupovima entiteta**. Za razliku od njih, **jaki skupovi entiteta** jesu oni koji imaju primarni ključ. Na primjer, skup entiteta *izdržavani* može da ima samo dva atributa *IME* i *STAROST*. Ovakav skup nema primarni ključ. Vrlo često se zbog toga slabim skupovima entiteta dodaju neki fiktivni atributi putem kojih će se jednoznačno odrediti entiteti u tom slabom skupu. Međutim, ovo nije uvijek moguće uraditi. Pojmovi jakog i slabog skupa su tijesno povezani sa pojmovima egzistencijalne zavisnosti skupova entiteta. Već smo rekli da neki odnos skupova entiteta može biti takav da u njemu razlikujemo dominantni skup, koji je ujedno jaki skup, i podređeni skup, koji je obično slabi skup. Uz to se uvodi ograničenje da odnos između dominantnog i podređenog (jakog i slabog) skupa mora biti 1:1 ili 1:n. Ako svaki skup entiteta ima svoj dominantni skup, onda se primarni ključ slabog skupa određuje na sljedeći način. Uzimamo primarni ključ dominantnog skupa i njemu dodajemo potrebne attribute iz slabog skupa. Tako oformljeni skup atributa predstavlja primarni ključ slabog skupa. Ovaj dodatni skup atributa slabog skupa entiteta naziva se **diskriminator slabog skupa**. Na primjer, ako imamo slabi skup *izdržavani*, sa atributima *IME-I* i *STAROST*, i jaki skup *zaposleni*, sa atributima *IME*, *BR. L. K.*, *STAŽ*, tada primarni ključ od *izdržavani* sadrži attribute *BR. L. K.*, *IME-I*, gdje je *BR. L. K.* primarni ključ od *zaposleni*, a *IME-I* diskriminator.

I skupovi odnosa imaju svoje primarne ključeve. Prema definiciji, primarni ključ skupa odnosa jeste unija primarnih ključeva svih skupova entiteta koji ulaze u taj odnos, odnosno:

$$PK(R) = \cup PK(E_i), \text{ gdje je } E_i \in R.$$

2.4.6. INTEGRITETSKA OGRANIČENJA E-R MODELA

Integritetska ograničenja jesu uslovi koje podaci moraju da zadovoljavaju da bi baza podataka bila konzistentna (tj. da bi odgovarala realnom stanju sistema). Integritetska ograničenja mogu biti **prosta** i **strukturna (referencijalna)**.

Prosta ograničenja kod E-R modela su:

- domenska ograničenja (STAROST 1–100, semestar I–IV, CIJENA realan broj, IME tekst dužine najviše 60 itd.) i
- primarni ključevi (različita pojavljivanja istog objekta moraju se jednoznačno identifikovati).

Strukturna (referencijalna) ograničenja kod E-R modela su:

- egzistencijalna zavisnost (postojanje podređenog objekta zavisi od postojanja dominantnog) i
- kardinalnost (broj pojavljivanja nekog objekta u bazi zavisi od kardinalnog broja)

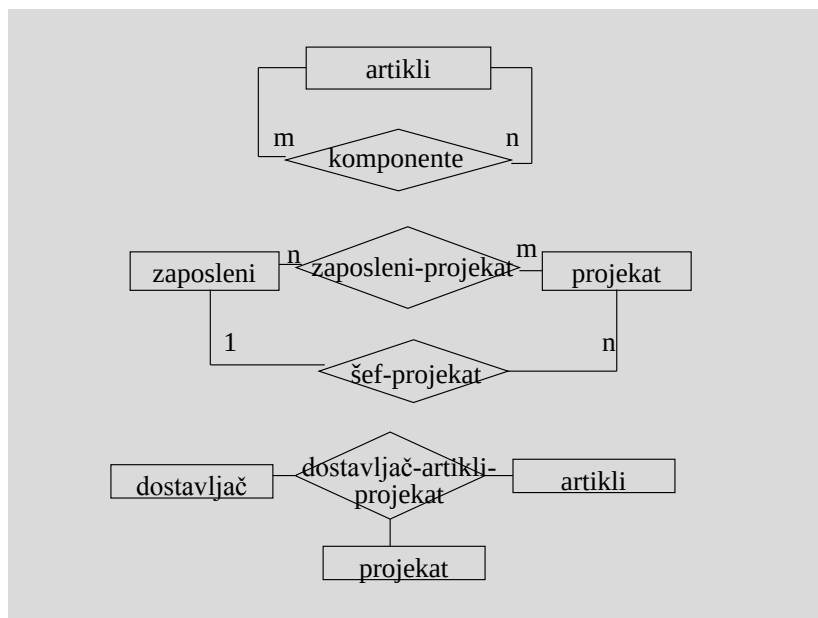
2.4.7. DIJAGRAMI E-R MODELA

Čitava konceptualna (logička) baza može se predstaviti putem dijagrama koji se sastoji od

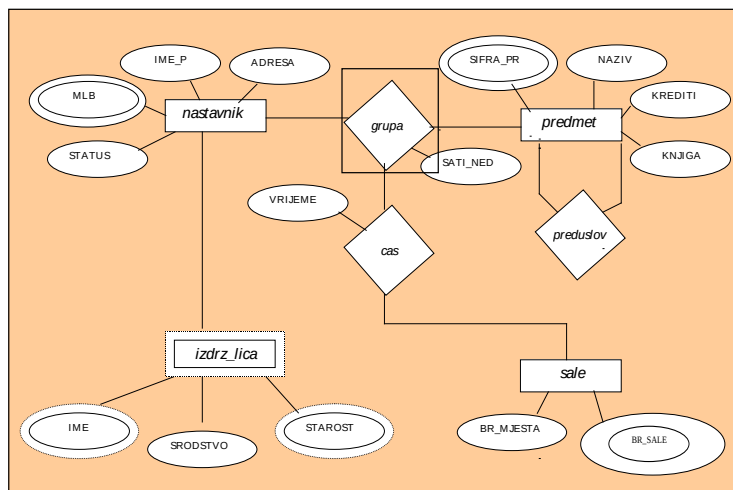
- pravougaonika, koji predstavljaju skupove entiteta;
- elipsi, koje predstavljaju attribute;
- rombova, koji predstavljaju skupove odnosa i
- linija, koje povezuju odgovarajuće attribute, skupove entiteta i odnosa.

Iz datog dijagrama može se uočiti sljedeće:

- Skup odnosa može biti definisan nad jednim te istim skupom entiteta, npr. skup odnosa *komponente* definisan je samo nad skupom entiteta *artikli* (slika 2.1. – gore).
- Više odnosa može da postoji između dva skupa entiteta, npr. između *zaposleni-projekat* mogu postojati odnosi: *zaposleni-projekat* i *šef-projekat*. Ovi skupovi odnosa semantički nisu isti (slika 2.1. – u sredini). Kardinalni broj odnosa *zaposleni-projekat* je $n:m$, što znači da jedan zaposleni može da radi na više projekata i jedan projekat može imati više zaposlenih. Kardinalni broj odnosa *šef-projekat* je $1:n$, što znači da jedan zaposleni može biti šef za više projekata ali jedan projekat ima samo jednog šefa.
- U principu, skup odnosa može postojati između više skupova entiteta. Npr. to bi se desilo kada bismo od odnosa *dostavljač-artikli* povukli liniju prema skupu entiteta *projekat*. Tada bismo dobili trajni odnos *dostavljač-artikli-projekat* (slika 2.1. – dolje).
- Na osnovu E-R dijagrama može se na jednostavan način uočiti koji su jaki, a koji slabi skupovi entiteta. Na slici 2.2. dat je skup entiteta *izdrz_lica* kao slabi skup, i predstavljen je grafički dvostrukim pravougaonikom.
- Jednoznačno se mogu uočiti ključevi skupova entiteta, a samim tim i ključevi skupova odnosa. Primarni ključevi se predstavljaju dvostrukom elipsom. Na slici 2.2. atribut *MLB* jeste primarni ključ skupa entiteta *nastavnik*. Primarni ključ odnosa *grupa* jeste *MLB*, *SIFRA_PR*.
- U principu, skupovi odnosa mogu povezivati ne samo skupove entiteta već i same skupove odnosa. Na slici 2.2. skup odnosa *cas* povezuje odnos *grupa* sa entitetom *sale*. Odnos *grupa*, koji predstavlja kraj veze, stavlja se u pravougaonik i takav odnos se zove odnos–entitet.
- Mogu se specifikovati domeni atributa.



Slika 2.1. E-R dijagrami



Slika 2.2. E-R dijagram dijela baze podataka fakulteta

2.4.8. INFORMACIONA STRUKTURA E-R MODELA

Entiteti, odnosi i njihovi atributi samo su apstraktni objekti u našoj svijesti, koje putem E-R dijagrama možemo grafički da predstavimo. U ovom poglavlju upoznaćemo se sa načinom predstavljanja tih konceptualnih (apstraktnih) objekata putem tzv. osnovne informacione strukture.

U E-R modelu postoji samo jedna informaciona struktura, a to je **tabela**. Dakle, skupovi entiteta (jaki i slabi) i skupovi odnosa predstavljaju se putem ovih tabela, a čitava baza podataka predstavlja jedan običan skup tabela. Za svaki skup entiteta ili skup odnosa radi se posebna tabela. Svaka tabela ima određeni broj kolona, i to onoliko koliko atributa ima skup. Ove kolone dobijaju imena odgovarajućih atributa i ona moraju biti jednoznačna.

Razmotrimo, na primjer, skup entiteta *zaposleni*, koji ima attribute *IME*, *BR. L. K.* i *STAŽ*. Ovom skupu atributa odgovara tabela koja ima naziv ***zaposleni***:

<u>BR. L. K.</u>	IME	STAŽ
112	Petar Petrović	10
173	Marko Ivanović	20

2.4.8.1. PREDSTAVLJANJE JAKIH SKUPOVA ENTITETA

Neka je E jaki skup entiteta sa atributima $A_1, A_2, \dots, A_n$; tada se ovaj skup može predstaviti tabelom E , koja ima n kolona, pri čemu svakoj koloni odgovara tačno jedan atribut. Svakoj vrsti tabele E odgovara jedan entitet iz skupa entiteta E . Vrste tabele u E-R modelu zovu se zapisi ili slogovi. Zapisi/slogovi se sastoje od polja koja sadrže vrijednosti odgovarajućih atributa za taj entitet.

Entiteti se u tabelu mogu unositi/brisati, tj. ažurirati. Tabele imaju svoje primarne ključeve, koji se poklapaju sa primarnim ključevima skupa entiteta kojem odgovara ta tabela. Atributi primarnog ključa u tabeli se podvlače.

2.4.8.2. PREDSTAVLJANJE SLABIH SKUPOVA ENTITETA

Neka je S slabi skup entiteta sa atributima $A_1, A_2, \dots, A_n$. Ako postoji jaki skup entiteta E , od kojeg S egzistencijalno zavisi, i ako je primarni ključ od E , $PK(E) = \{B_1, B_2, \dots, B_k\}$, tada se slabi skup entiteta S predstavlja tabelom S čije kolone odgovaraju sljedećem skupu atributa: $\{B_1, B_2, \dots, B_k\} \cup \{A_1, A_2, \dots, A_n\}$. Ovdje se pretpostavlja da svi atributi imaju različita imena.

Primarni ključ tabele S dobija se na sljedeći način:

$$PK(S) = PK(E) \cup DESCR(S), \text{ isto kao kod slabog skupa } S.$$

Na primjer, slabi skup entiteta *izdržavani* ima attribute *IME*, *STAROST*. Ovaj skup zavisi od jakog skupa entiteta *zaposleni*, čiji je primarni ključ: $PK(\text{zaposleni}) = BR. L. K.$, pa tabela *izdržavani* izgleda ovako:

izdržavani

<u>BR. L. K.</u>	<u>IME IZDRŽ</u>	<u>STAROST</u>
111	Marko Ivanović	7
111	Marko Ivanović	77
222	Sofija Ilić	9

2.4.8.3. PREDSTAVLJANJE SKUPOVA ODNOSA

Neka je R odnos između skupova entiteta $E_1, E_2, \dots, E_k$. Skup odnosa R predstavlja se tabelom R , čije kolone odgovaraju sljedećem skupu atributa:

$\bigcup_{i=1}^k PK(E_i) \cup \{A_1, A_2, \dots, A_n\}$, pri čemu su $A_1, A_2, \dots, A_n$ sopstveni atributi skupa odnosa.

Primarni ključ ove tabele predstavlja uniju $\bigcup_{i=1}^k PK(E_i)$ po Chenovoj definiciji, mada to nije uvijek tako u praksi.

Na primjer, skup odnosa *dostavljač-artikli*, koji ima sopstveni atribut *CIJENA*, može biti predstavljen sljedećom tabelom:

$$PK(\text{dostavljač}) = \{IME_D, ADRESA\}$$

$$PK(\text{artikli}) = BR_ART$$

dostavljač-artikli

<u>IME_D</u>	<u>ADRESA</u>	<u>BR_ARTIKLA</u>	<u>CIJENA</u>
NEG	SR_NJ	596	100 DEM
EI	NIŠ	120	250 DEM
:	:	:	:

Ako se radi o slabom skupu odnosa, tj. skupu odnosa između jakog i slabog skupa, onda osim PK jakog skupa u atribute tabele datog skupa odnosa ulazi samo diskriminator.

2.4.9. PREDNOSTI I NEDOSTACI E-R MODELA

Prednosti E-R modela su sljedeće:

- E-R dijagrami su pogodno sredstvo za predstavljanje grafičkih modela baze;
- putem E-R modela na jednoznačan način mogu se izraziti integritetska ograničenja, kao što su kardinalni broj odnosa i egzistencijalna zavisnost, a one se vrlo teško ili nikako ne mogu izraziti u drugim modelima;
- jasno su razgraničeni pojmovi entiteta i njihovih odnosa;
- jasno je definisan način predstavljanja jakih i slabih skupova entiteta, kao i skupova odnosa;
- smatra se da E-R model može da pokupi više semantičke informacije iz realnog svijeta koji se modelira, nego ostali modeli, kao što je recimo relacioni.

Osnovni nedostaci E-R jesu:

- E-R nema apstraktno definisanih apstraktnih operatora;
- višestruki odnosi (između tri ili više skupova entiteta) teško se mogu predstaviti putem ovog modela zbog kardinalnog broja odnosa i drugog.

Ipak, izgleda da je nedovoljna primjena ovog modela u praksi više posljedica nekih eksternih faktora nego samih nedostataka ovog modela.

2.4.10. PRIMJER: DIO BAZE PODATAKA BIBLIOTEKE

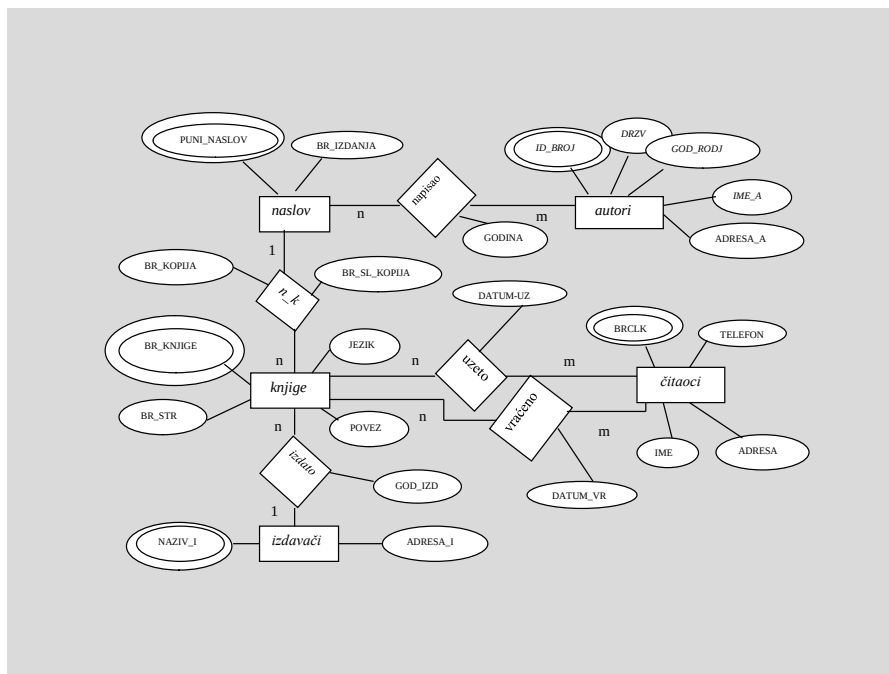
Potrebno je nacrtati E-R dijagram i predstaviti tabelama bazu podataka za biblioteku. Dati su sljedeći podaci:

- a) za čitaoce: BRCLK, IME, ADRESA, TELEFON
- b) za knjige: BR_KNJIGE, BR_STR, POVEZ, JEZIK
- c) za naslove: PUNI_NASLOV, BR_IZDANJA
- d) za autore: ID_BROJ, IME, ADRESA, DRZ, GOD_RODJ
- e) za izdavače: NAZIV_I, ADRESA_I

Odgovarajući E-R dijagram predstavljen je na slici 2.3.

Neke od tabela koje odgovaraju entitetima i odnosima sa slike 2.3 date su na slici 2.4. Može se primijetiti da primarni ključ tabele za skupove odnosa *uzeto* i *vraćeno* nije određen po Chenovoj definiciji. Nekada unija primarnih ključeva skupova entiteta nije dovoljna za jednoznačnu identifikaciju slogova u tabeli veze, pa se mora uzeti dodatni atribut. Tako se na primjer kod tabele *uzeto* može desiti da jedan čitalac istu knjigu uzima više puta. Iz tog razloga u PK ulazi i datum uzimanja knjige.

Veza naslov-knjige n_k jeste veza tipa 1:n. Prema originalnoj Chenovoj definiciji, ova veza se predstavlja posebnom tabelom kao na slici 2.5. – gore. Međutim, ovakve veze se mogu predstaviti i jednostavnim dodavanjem spoljnog ključa podređenoj tabeli (onoj kod koje je veza označena sa n) (slika 2.5. – dolje). Kako će veza biti predstavljena, zavisi od toga da li ona ima svoje veze sa drugim entitetima ili odnosima u bazi. Ako ima sopstvene veze, onda se predstavlja kao posebna tabela.



Slika 2.3. E-R dijagram dijela baze podataka biblioteke

knjige			
<u>BR_KNJIJE</u>	BR_STR	JEZIK	POVEZ
125	130	Engleski	tvrdi
156	250	Engleki	tvrdi
138	157	Ruski	meki

uzeto			
<u>BR_KNJIJE</u>	<u>BRCLK</u>	<u>DATUM_U</u>	VRIJEME
125	1111	1.03.2011	15 dana
156	1211	1.04.2011	10 dana
138	1213	1.05.2011	11 dana

vraćeno		
<u>BR_KNJIJE</u>	<u>BRCLK</u>	<u>DATUM_V</u>
125	1111	15.03.2011
125	1123	28.04.2011

Slika 2.4. Predstavljanje entiteta i odnosa tabelom

n_k

<u>BR_KNJIGE</u>	<u>PUNI_NASLOV</u>	BR_KOPIJA	BR_SL_KOPIJA
123	Antigona	50	45
124	Antigona	50	45
139	Romeo i Julija	150	90

knjige

<u>BR_KNJIGE</u>	BR_STR	JEZIK	POVEZ	PUNI_NASLOV
123	50	Engleski	meki	Antigona
124	65	Španski	tvrdi	Antigona

Slika 2.5. Predstavljanje veze 1:n tabelom

2.5. RELACIONI MODEL

Relacioni model se najviše primjenjuje za modeliranje poslovnih podataka.

2.5.1. RELACIJA – OSNOVNA INFORMACIONA STRUKTURA RELACIONOG MODELA

Osnovna informaciona struktura relacionog modela jeste tabela, a baza podataka predstavlja jedan običan skup tabela. Kao i kod E-R modela, kolone tabele odgovaraju atributima, a redovi pojedinim entitetima. Međutim, ovo je samo jedan od načina posmatranja relacione baze podataka, i to način na koji naivni korisnici gledaju na relacionu bazu. Obrazovaniji korisnici, a u prvom redu projektanti baze podataka, posmatraju relacionu bazu kao jedan skup matematičkih formalizama i u svjetlu čitave algebarske teorije u vezi sa relacijama.

Ovi različiti pogledi na jednu te istu relacionu bazu (uopšteno gledište naivnih korisnika i daleko složenije gledište obrazovanih korisnika)

predstavlja jednu od najvećih prednosti relacionog modela. Zato ćemo u prvom redu formalno definisati osnovnu informacionu strukturu relacionog modela.

S formalne strane, osnovna informaciona struktura relacionog modela jeste relacija. U matematičkim terminima relaciju r definišemo kao podskup Dekartovog proizvoda domena, odnosno:

$$r \subset D_1 \times D_2 \times \dots \times D_n,$$

gdje su D_i , $i = 1 \dots n$, domen, tj. konačni i beskonačni skupovi vrijednosti.

Postoji jedna razlika između relacije definisane na čisto matematički način i relacija koje se odnose na relacione baze podataka. Naime, dok matematičke relacije mogu imati beskonačan broj elemenata, relacioni model zahtijeva da svaka relacija bude konačna. Elemente relacije nazivamo n -torke. Elementi relacije su vektori oblika: $t = (v_1, v_2, \dots, v_n)$ gdje je n stepen relacije a v_i , $i = 1 \dots n$ – odgovarajuće vrijednosti iz skupa vrijednosti D_i , $i = 1 \dots n$.

Skupovi n -torki mogu se posmatrati kao obične tabele. Kolonama tih tabela pridružena su atributska imena, tj. atributi. Svakom atributu A_i odgovara domen D_i .

Skup atributa neke relacije r označen sa $R = \{A_1, A_2, \dots, A_n\}$ naziva se **relaciona šema**. Skup relacionih šema svih relacija naziva se **šema relacione baze**. Šema relacione baze u stvari predstavlja konceptualnu bazu podataka u relacionom modelu. Znači, konceptualna baza podataka u relacionom modelu jeste skup relacionih šema.

Same relacione šeme označavaćemo velikim slovima $R, S, Q, \dots$, a relacije na tim relacionim šemama – malim slovima $r, s, q, \dots$. Relacionu šemu je ispravno pisati kao $R = \{A_1, A_2, \dots, A_n\}$, ali ćemo mi često koristiti sljedeću notaciju $R = A_1, A_2, \dots, A_n$, znajući pri tome da raspored atributa u relacionoj šemi nije bitan, tj. $R = ABC$ isto je što i $R = CBA$. Na sličan način odgovarajuću relaciju sa tom relacionom šemom označavaćemo sa $r(R)$ ili $R(ABC)$. Attribute ćemo označavati velikim slovom $A_1, A_2, \dots, A_n$, a vrijednosti tih atributa sa $a_1, a_2, \dots, a_n$. Kao i kod E-R modela, relacije imaju natključeve, kandidatske ključeve i primarne ključeve, koji se definišu na isti način. Napomenimo da u relacionom modelu postoji samo tabela, tj.

relacija i nema nikakve razlike između skupova entiteta i skupova odnosa. Između relacije u relacionom modelu i skupova odnosa u E-R modelu postoji fundamentalna razlika. Naime, relacije su asocijacije između prostih domena, dok su odnosi u E-R modelu asocijacije između skupova entiteta.

Primjer: Sastaviti šemu za relacionu bazu podataka za biblioteku koja je data E-R dijagramom na slici 2.3.

naslovi (PUNI NASLOV, BR_IZDANJA)

autori (ID_BROJ, IME_A, ADRESA_A, DRZV, GOD_RODJ)

napisao (ID_BROJ, PUNI_NASLOV, GODINA)

knjige (BR_KNJIGE, BR_STR, POVEZ, JEZIK)

n-k (PUNI_NASLOV, BROJ_KNJIGE, BROJ_KOPIJA, BR_SL_KOPIJA)

čitaoci (BRCL, IME, ADRESA, TELEFON)

uzeto (BR_KNJIGE, BRCL, DATUM_UZ, VRIJEME)

vraćeno (BR_KNJIGE, BRCL, DATUM_VR)

izdavači (NAZIV_I, ADRESA_I)

izdato (BR_KNJIGE, NAZIV_I, GOD_IZD)

2.5.2. AŽURIRANJE RELACIJA

Do sada smo posmatrali relacije kao statičke objekte. Međutim, relacije predstavljaju neku vrstu apstrakcije određenog dijela stvarnosti koji se mijenja i prelazi iz jednog stanja u drugo. Na isti način i relacije mijenjaju svoja stanja. Stanje relacije određeno je njenim sadržajem, a dodavanjem, brisanjem i ažuriranjem n-torki to stanje se može promijeniti.

Dodavanje n-torke

Pretpostavimo da relaciji $r(R)$, $R = A_1, A_2, \dots, A_n$ želimo da dodamo novu n-torku. Ovo radimo pomoću operatora *dodaj* koji ima sljedeći oblik:

ADD(r; A1 = v1, A2 = v2, ..., An = vn) ili

ADD(r; v1, v2, ..., vn), ukoliko je fiksiran raspored atributa, gdje su vi vrijednosti iz odgovarajućih domena Di.

Ova operacija može da propadne ukoliko je:

- navedena n-torka nije specifikovana u skladu sa relacionom šemom;
- neke vrijednosti ne pripadaju domenima odgovarajućih atributa;
- navedena n-torka koja se dodaje ima isti ključ kao neka već unijeta n-torka;
- vrijednosti za neke attribute (prvenstveno za ključne) nijesu definisane.

U sva četiri slučaja operacija se abortira (prekida) i data n-torka se ne unosi u relaciju.

Brisanje n-torke

Često je potrebno izbrisati neku prethodno unesenu n-torku. Pretpostavimo da iz relacije $r(R)$, $R = A1, A2, \dots, An$ želimo da izbrišemo n-torku. To radimo pomoću sljedećeg operatora;

DEL(r; A1 = v1, A2 = v2, ..., An = vn) ili

DEL (r; v1, v2, ..., vn), ako je raspored atributa fiksiran.

Međutim, ako je poznat ključ relacije r $PK = \{B1, B2, \dots, Bk\} \subseteq R$, onda se ne moraju specificovati vrijednosti svih atributa, već samo ključnih:

DEL(r; B1 = b1, B2 = b2, ..., Bk = bk)

Operacija brisanja propada samo u slučaju ako se odgovarajuća n-torka ne nalazi u datoj relaciji.

Ažuriranje n-torke

Pored operacija dodavanja i brisanja, postoji i operacija ažuriranja pojedinih vrijednosti n-torke. Npr. pretpostavimo da u relaciji $r(R)$, $R = A1, A2, \dots$,

An želimo da izmijenimo vrijednosti atributa iz skupa $(C1, C2, \dots, C_m) \subset (A1, A2, \dots, A_n)$. To onda radimo pomoću sljedećeg operatora:

MODIFY(r; A1 = v1, A2 = v2, ..., An = vn; C1 = c1, C2 = c2, ..., Cm = cm)
ili

MODIFY(r; v1, v2, ..., vn; C1 = c1, C2 = c2, ..., Cm = cm)

Međutim, ako je poznat primarni ključ relacije r : $PK = \{B1, B2, \dots, B_k\} \subseteq \{A1, A2, \dots, A_n\}$, onda se operacija može opisati na sljedeći način:

MODIFY(r; B1 = b1, B2 = b2, ... Bk = bk; C1 = c1, C2 = c2, ..., Cm = cm)

Na ovu operaciju možemo da gledamo kao na operaciju brisanja, a zatim dodavanja. Zato će ova operacija biti abortirana u onim slučajevima u kojima ne uspijevaju operacije brisanja odnosno dodavanja.

Znači, operacije ADD, DEL, MODIFY služe za promjenu stanja jedne relacije. Svako trenutno stanje relacije uslovljeno je nizom prethodnih operacija nad n -torkama te relacije. Međutim, trenutno stanje te relacije nema pamćenje, tj. na osnovu njega ne možemo da rekonstruišemo redosljed prethodnih operacija nad n -torkama. U tom smislu relacije se mogu posmatrati i kao statički fiksirane u vremenu.

2.5.3. RELACIONA ALGEBRA

Operacije ADD, DEL, MODIFY jesu standardne operacije nad n -torkama te relacije. Nas će sada interesovati operatori koji kao svoje argumente dobijaju same relacije, a ne njihove n -torke. Skup ovih operatora sačinjava relacionu algebru. Relaciona algebra predstavlja apstraktan jezik relacionog modela, putem kojeg se na apstraktan način predstavljaju upiti bazi podataka. Sastoji se od nekoliko operacija nad relacijama. Kombinovanjem operacija mogu se izraziti skoro svi upiti nekoj bazi. Relacija je zatvorena u odnosu na ove operacije, tj. svaka od njih uzima kao argument relaciju, i kao rezultat opet daje relaciju. Prema [13], pored skupovnih operacija kao što su unija, presjek i razlika, u relacionu algebru spadaju i neke specifične operacije, kao što su selekcija, projekcija i spajanje.

2.5.3.1. SKUPOVNE OPERACIJE

Skupovne operacije ćemo razmatrati na primjeru sljedećih relacija:

$r(A \ B \ C)$	$s(A \ B \ C)$
a1 b2 c1	a1 b2 c2
a1 b1 c2	a2 b2 c2
a2 b1 c2	a1 b2 c1

a) unija $r(R) \cup s(R) = \{t \mid t \in r \vee t \in s\}$

$r(ABC) \cup s(ABC) = q(A \ B \ C)$
a1 b2 c1
a1 b1 c2
a2 b1 c2
a1 b2 c2
a2 b2 c2

b) presjek $r(R) \cap s(R) = \{t \mid t \in r \wedge t \in s\}$

$r(ABC) \cap s(ABC) = q(A \ B \ C)$
a1 b2 c1

c) razlika $r(R) / s(R) = \{t \mid t \in r \wedge t \notin s\}$

$r(ABC) / s(ABC) = q(A \ B \ C)$
a1 b1 c2
a2 b1 c2

2.5.3.2. OPERATOR SELEKCIJE

Definicija: Predikatskom formulom F nad relacijom $r(R)$ nazivamo izraz koji se sastoji od:

- atributa i konstanti iz domena tih atributa (iz relacione šeme R);
- operacija upoređivanja iz skupa $\{<, >, =, < >, < =, > =\}$
- logičkih operatora iz skupa $\{\wedge, \vee, T\}$
- otvorene i zatvorene zagrade $(,)$.

Primjer: $(A < a1 \wedge A \geq B) \wedge \neg (A = a1 \vee A = a2)$

Primjer: $(IME = \text{" MARKO " } \wedge STAZ < 10) \vee BR \text{ GODINA} = 25$

Ako se upoređuju dva atributa, onda ta dva atributa moraju biti kompatibilna, tj. moraju imati isti domen.

Definicija: Neka je $r(R)$ relacija sa relacionom šemom R . Pod **operatorom selekcije** nad relacijom r , u oznaci $\sigma_F(r)$ podrazumijeva se unarna operacija čiji je rezultat relacija $r'(R)$ sa istom relacionom šemom R koja sadrži n -torke iz r koje zadovoljavaju predikat F .

Osobine operatora selekcije su:

- **komutativnost**

$$\sigma_F(\sigma_G(r)) = \sigma_G(\sigma_F(r))$$

- **distributivnost u odnosu na skupovne operacije**

$$\sigma_F(r \varphi s) = \sigma_F(r) \varphi \sigma_F(s) \text{ za } \varphi \in \{\cup, \cap, /\}$$

Primjer: Neka je data relacija r sa relacionom šemom ABC koja ima sljedeće n -torke:

r	A	B	C
	$a1$	$b1$	$c1$
	$a2$	$b1$	$c2$
	$a2$	$b2$	$c2$

Naći $\sigma_{B=b1}(r)$.

$\sigma_{B=b1}(r)$	r'	A	B	C
		$a1$	$b1$	$c1$
		$a2$	$b1$	$c2$

Ovo je unarna operacija jer kao argument ima samo jednu relaciju (a ne n -torke).

Primjer: $r(A \ B \ C)$

	1	$b1$	3
	3	$b2$	2
	4	$b3$	6

Naći $\sigma_{A \leq C}(r)$.

$$\sigma_{A \leq C}(r) = r' \begin{pmatrix} A & B & C \\ 1 & b1 & 3 \\ 4 & b3 & 6 \end{pmatrix}$$

2.5.3.3. OPERATOR PROJEKCIJE

Definicija: Neka je $r(R)$ relacija sa relacionom šemom R i neka je $X \subset R$. Pod operatorom projekcije nad relacijom r u oznaci $\Pi_X(r)$ podrazumijeva se unarna operacija čiji je rezultat relacija r' sa relacionom šemom X koja ne sadrži istovjetne n -torke.

Osobine operatora projekcije su:

- **sublimacija**

Ako je $X_1 \subseteq X_2 \subseteq \dots \subseteq X_n$, tada $\Pi_{X_1}(\Pi_{X_2}(\dots \Pi_{X_n}(r) \dots)) = \Pi_{X_1}(r)$

- **komutativnost sa selekcijom**

$$\Pi_X(\sigma_F(r)) = \sigma_F(\Pi_X(r))$$

Primjer: Neka je data relacija r sa relacionom šemom $A \ B \ C$ i n -torkama

$$r \begin{pmatrix} A & B & C \\ a1 & b1 & c1 \\ a1 & b2 & c1 \\ a2 & b1 & c2 \end{pmatrix}$$

$$\Pi_{AC}(r) = r' \begin{pmatrix} A & C \\ a1 & c1 \\ a2 & c2 \end{pmatrix}$$

2.5.3.4. OPERATOR SPAJANJA

Definicija: Neka su $r(R)$ i $s(S)$ relacije. Pod operatorom spajanja nad relacijama r i s u oznaci $r \times s$ podrazumijeva se binarna operacija čiji je rezultat relacija $q(R \cup S)$ za čije n -torke t važi da su dobijene iz nekih n -torki $t_r \in r$ i $t_s \in s$, takvih da $t_r(R \cap S) = t_s(R \cap S)$.

Osobine operatora spajanja su:

- **komutativnost** $r \times s = s \times r$

- **asocijativnost** $r \times (s \times q) = (r \times s) \times q$
- **distributivnost selekcije u odnosu na spajanje**
 $\sigma_F(r \times s) = \sigma_F(r) \times \sigma_F(s)$
- **distributivnost spajanja u odnosu na uniju**
 $r \times (s \cup q) = (r \times s) \cup (r \times q)$

Primjer: $r(A \ B \ C) \quad s(A \ C \ D)$

a1 b1 c1	a1 c1 d1
a1 b2 c1	a1 c2 d2
a2 b1 c2	a2 c2 d3

$r \times s = q(A \ B \ C \ D)$

a1 b1 c1 d1
a1 b2 c1 d1
a2 b1 c2 d3

2.5.3.5. PRIMJERI ZA RELACIONU ALGEBRU

Primjer 1. Provjeriti da li baza podataka za biblioteku čiji je E-R dijagram dat na slici 2.3, može da odgovori na sljedeće upite (pomoću relacione algebre):

1. Naći imena i adrese svih autora:

$$\Pi_{IME_A, ADRESA}(\text{autori})$$

2. Naći imena i adrese svih jugoslovenskih autora:

$$\Pi_{IME_A, ADRESA}(\sigma_{DRZV='Yu'}(\text{autori}))$$

3. Naći naslove i broj izdanja svih djela koje je napisao Miroslav Antić:

$$\Pi_{PUNI_NASLOV, BR_IZDANJA}(\sigma_{IME_A='Miroslav Antić'}(\text{naslovi} \mid X \mid \text{napisao} \mid X \mid \text{autori}))$$

4. Naći imena svih korisnika koji su pozajmili knjige poslije 10. 02. 2000. godine:

$$\Pi_{IME}(\text{CITAOCI} \mid X \mid \sigma_{DATUM_UZ > 10.02.2000}(\text{uzeto}))$$

5. Naći brojeve knjiga i naslove koje je pozajmio čitalac sa članskom kartom 793, a nije ih vratio:

$$\prod_{BROJ_KNJIGE, PUNI_NASLOV} (n - k |X| \sigma_{BRCL = 793} (uzeto))$$

6. Naći sve knjige koje je do sada uzimao čitalac sa članskom kartom 128:

$$\prod_{BR_KNJIGE, PUNI_NASLOV} (n - k |X| \sigma_{BRCL = 128} (\prod_{BRCL, BR_KNJIGE} (uzeto) \cup \prod_{BRCL, BR_KNJIGE} (vraceno)))$$

7. Naći sve godine izdanja i izdavače za naslov “Braća Karamazovi”:

$$\prod_{GOD_IZD, NAZIV_I} (izdato |X| \sigma_{PUNI_NAZIV = 'Braca Karamazovi'} (n - k))$$

8. Naći broj slobodnih kopija naslova “Bludni sin”:

$$\prod_{BROJ_SL_KOPIJA} (\sigma_{PUNI_NASLOV = 'Bludni sin'} (n - k))$$

9. Naći brojeve knjiga, broj strana i povež za knjige pod naslovom “Alhemičar”:

$$\prod_{BR_KNJIGE, BR_STR, POVEZ} (\sigma_{PUNI_NASLOV = 'Alhemicar'} (n - k) |X| knjige)$$

10. Naći broj članske karte i imena čitalaca koji trenutno čitaju djela Milorada Pavića:

$$\prod_{BRCL, IME} (citaoci |X| uzeto |X| n - k |X| napisao |X| \sigma_{IME_AUTORA = 'Milorad Pavic'} (autori))$$

Primjer 2. Baza podataka bankarskog sistema sadrži sljedeće relacije:

1. stranka (#L_K, IME_S, GRAD_S)
2. poslovnica (IME_P, GRAD_P)
3. štednja (#L_K, #RAČUNA, IME_P, STANJE)
4. zajam (#L_K, #ZAJMA, IME_P, IZNOS)

Atributi #L_K i IME_P predstavljaju spoljne ključeve u relacijama *štednja* i *zajam*. Prema Chenovoj definiciji unija ova dva atributa trebalo bi da bude PK u ovim relacijama. Međutim, s obzirom na to da

ista stranka može u jednoj poslovnici imati više računa, PK je #RACUNA.

1. Naći sve stranke poslovnice blok 5:

$$r = \prod_{\#L_K, IME_S} (stranka |X| \sigma_{IME_P = 'Blok5'} (stednja))$$

$$S = \prod_{\#L_K, IME_S} (stranka |X| \sigma_{IME_P = 'Blok5'} (zajam))$$

$$q = r \cup S$$

2. Naći sve stranke poslovnice Blok 5 koji samo štede i imaju pozajmice:

$$q = r \cap S$$

3. Naći sve stranke poslovnice Blok 5 koji samo štede, ali nemaju pozajmice:

$$q = r \setminus S$$

4. Naći sve poslovnice koje imaju štediše koje žive u Podgorici:

$$\prod_{IME_P} (stednja |X| \sigma_{GRAD_S = 'Podgorica'} (stranke))$$

5. Naći sve stranke koje imaju štedne uloge van gradova u kojima žive:

$$\prod_{\#LK, IME_S} (\sigma_{GRAD_S \neq GRAD_P} (poslovnica |X| stednja |X| stranka))$$

6. Naći sve stranke koje ne štede u gradu u kojemu žive:

$$r = \prod_{\#LK, IME_S} (\sigma_{GRAD_S = GRAD_P} (poslovnica |X| stednja |X| stranka))$$

$$S = \prod_{\#LK, IME_S} (stranka |X| stednja)$$

$$q = S \setminus r$$

2.5.4. UPITNI JEZIK SQL

SQL (Structured Query Language) jeste standardni upitni jezik relacionih DBMS. Može se reći da se koristi u dvije svrhe, i to za pretraživanje podataka i za ažuriranje podataka.

2.5.4.1. PRETRAŽIVANJE PODATAKA

Osnovni izraz SQL za pretraživanje ima sljedeći izgled:

```
SELECT A1, A2, ..., An
FROM r1, r2, ..., rk
WHERE P
```

gdje su $r_1, r_2, \dots, r_k$ relacije; $A_1, A_2, \dots, A_n$ atributi tih relacija i P logički predikat. Ovakav izraz ekvivalentan je sljedećem izrazu relacione algebre:

$\Pi_{A_1, A_2, \dots, A_n}(\sigma_P(r_1 \times r_2 \times \dots \times r_k))$, gdje je $\times$ Dekartov proizvod.

U SELECT klauzuli navodi se spisak atributa rezultirajuće relacije koju treba izlistati. SELECT klauzula može da sadrži i neke specijalne funkcije, kao što su:

1. * – zamjenjuje spisak svih atributa rezultirajuće relacije;
2. DISINCT – selektuje samo različite n-torke rezultirajuće relacije i
3. derivacione funkcije, kao što su: SUM, MAX, MIN, AVG i COUNT.

U FROM klauzuli specificuju se relacije koje učestvuju u upitu, a može biti navedena i jedna ili više promjenljivih tipa n-torke. Ove promjenljive često se nazivaju nadimci relacije.

Pod izrazom WHERE specificuje se selekcija n-torki rezultirajuće relacije koje zadovoljavaju predikat P .

U predikatu P figuriraju:

1. atributi relacija iz podizraza FROM;
2. konstante iz domena tih atributa;
3. relacije ili njihovi nadimci iz podizraza FROM;
4. znaci poređenja iz skupa: $\{=, <, >, <>, <=, >=\}$
5. operacije poređenja alfanumerika, kao što su: $\{\text{LIKE, CONTAINING, STARTING, WITH}\}$;
6. logički operatori iz skupa $\{\text{AND, OR, NOT}\}$;
7. operator pripadnosti skupu $\text{IN } (\in)$;
8. operator podskupa $\text{CONTAINS } (\subseteq)$;

9. operatori NOT IN i NOT CONTAINS();
10. operatori za ispitivanje nula polja: IS NULL, IS NOT NULL;
11. konstrukcije ANY i ALL, koje omogućavaju da se jedna vrijednost uporedi sa bilo kojom ili sa svim vrijednostima iz nekog skupa;
12. operator pripadnosti intervalu BETWEEN;
13. tačka, otvorena mala i zatvorena mala zagrada{., (,)}.

Osim ovog, SQL koristi i ključne riječi UNION, INTERSECT i MINUS za skupovne operacije unija, presjek i razlika.

Neki operatori relacione algebre mogu se direktno izraziti SQL-om:

1. Projekcija

Neka je $r(R)$ relacija, a $R = A_1, A_2, \dots, A_n$ i neka je $X \subset R$ za $X = A_1, A_2, \dots, A_m$. Tada se $\Pi_X(r(R))$ može izraziti sa:

```
SELECT A1, A2, ..., Am
FROM r
```

2. Selekcija

Neka je $r(R)$ relacija, a $R = A_1, A_2, \dots, A_n$. Tada se $\sigma_P(r(R))$ može izraziti sa:

```
SELECT *
FROM r
WHERE P
```

3. Spajanje

Neka su $r(AB)$ i $s(BC)$ dvije relacije. Tada se $r(AB) \times s(BC)$ može izraziti kao:

```
SELECT A,B,C
FROM r,s
WHERE r.B=s.B
```

4. Unija

Neka su $r(R)$ i $s(R)$ relacije, tada se $r(R) \cup s(R)$ može izraziti sa:

```
(SELECT *  
FROM r)  
UNION  
(SELECT *  
FROM s)
```

5. Presjek

Neka su $r(R)$ i $s(R)$ relacije, tada se $r(R) \cap s(R)$ može izraziti sa:

```
(SELECT *  
FROM r)  
INTERSECT  
(SELECT *  
FROM s)
```

6. Razlika

Neka su $r(R)$ i $s(R)$ relacije, tada se $r(R) - s(R)$ može izraziti sa:

```
(SELECT *  
FROM r)  
MINUS  
(SELECT *  
FROM s)
```

Primjer: Za bazu podataka bankarskog sistema:

stranka (BR_LK, IME_S, GRAD)
poslovnica (IME_P, GRAD_P)
štednja (BR_LK, BR_RAC, IME_P, STANJE)
zajam (BR_LK, BR_ZAJMA, IME_P, IZNOS)

1. Naći ukupan iznos uloga štediše Marka Jovanovića.

```
SELECT SUM(STANJE)  
FROM stednja U, stranka S  
WHERE U.BR_LK=S.BR_LK AND  
S.IME_S="Jovanović Marko"
```


2. Naći imena svih stranaka koje štede u ovoj banci.

```
SELECT DISTINCT IME_S  
FROM stranka X, stednja Y  
WHERE X.BR_LK=Y.BR_LK
```

3. Naći broj štediša ove banke.

```
SELECT COUNT (DISITNCT BR_LK)  
FROM stednja
```

4. Naći najveći zajam koji je banka dala.

```
SELECT MAX(IZNOS)  
FROM zajam
```

5. Naći minimalno stanje na nekom računu poslovnice Blok V.

```
SELECT MIN(STANJE)  
FROM stednja  
WHERE IME_P="Blok V"
```

6. Naći imena poslovnica koje imaju štedište iz Podgorice.

```
SELECT DISTINCT IME_P  
FROM stranka, stednja  
WHERE stranka.BR_LK=stednja.BR_LK AND  
stranka.GRAD_S="Podgorica"
```

7. Naći sve štediše čiji broj računa ima prefiks 12/3-.

```
SELECT DISTINCT BR_LK  
FROM stednja  
WHERE BR_RAC STARTING WITH "12/3-"
```

8. Naći sve stranke kojima grad u kojem žive nije evidentiran.

```
SELECT BR_LK, IME_S  
FROM stranka  
WHERE GRAD_S IS NULL
```

9. Naći sve štediše čiji su ulozi između 20000 i 30000.

```
SELECT BR_LK  
FROM stednja  
WHERE STANJE BETWEEN 20000 AND 30000
```

10. Naći one štediše koji nemaju uloge u iznosu 20000 i 30000.

```
SELECT BR_LK  
FROM stednja  
WHERE STANJE NOT IN (20000, 30000)
```

2.5.4.2. AŽURIRANJE PODATAKA I POGLEDI

1. Unošenje podataka

Osnovna naredba SQL-a za unošenje podataka ima oblik:

```
INSERT INTO naziv_relacije [(nazivi_atributa)]  
VALUES (spisak_vrijednosti)
```

U uglastim zagradama se nalazi dio naredbe koji može biti izostavljen ukoliko smo sigurni kakav je redosljed atributa u bazi. Unošenje podataka obavlja se po principu odozgo prema dolje, što znači da se podaci prvo unose u nadređene, pa tek onda u podređene relacije. Podređene relacije jesu relacije koje predstavljaju entitete koji su u vezi 1:n sa nadređenim entitetima ili relacije koje predstavljaju veze između entiteta.

Primjer: Stari štediša sa brojem lične karte 121 u poslovnici Blok V uložio je 45000 dinara, na računu broj 12/3-003. Ažurirati bazu podataka iz prethodnog primjera.

```
INSERT INTO stednja (BR_LK, BR_RAC, IME_P, STANJE)
VALUES (121, "12/3-003", "Blok V", 45.000)
```

Kod ovog primjera bitno je da je riječ o starom štediši, što znači da je već unesen u relaciju stranke, pa ga možemo unijeti u relaciju stednja (koja predstavlja vezu između stranke i poslovnice), poštujući princip unošenja odozgo prema dolje.

2. **Brisanje podataka**

Osnovna naredba SQL-a za brisanje podataka ima oblik:

```
DELETE FROM naziv_relacije
WHERE uslov_selekcije
```

Brisanje podataka se obavlja po principu odozdo na gore, što znači da se prvo brišu podaci iz podređenih, pa tek onda iz nadređenih relacija.

Primjer: Štediša sa brojem lične karte 353 ugasio je račun broj 12/4-222. Ažurirati bazu podataka iz prethodnog primjera.

```
DELETE FROM stednja
WHERE BR_LK=353 AND BR_RAC="12/4-222"
```

Kod ovog primjera je izbrisan jedan račun stranke. Ukoliko stranka ima još računa, ne možemo je izbrisati prije nego izbrišemo sve njene račune.

3. **Ispravka podataka**

Osnovna naredba SQL-a za ispravku podataka ima oblik:

```
UPDATE naziv_relacije
SET (naziv_atributa=vrijednost_atributa,...)
WHERE uslov_selekcije
```

Kod ispravke treba imati u vidu da se ne mogu ispravljati vrijednosti atributa primarnih ključeva koji su spuštene kao spoljni ključevi u neku drugu relaciju.

Primjer: Stranka sa brojem lične karte 323 preselila se u Podgoricu. Ažurirati bazu podataka iz prethodnog primjera.

```
UPDATE stranka  
SET GRAD_S="Podgorica"  
WHERE BR_LK=323
```

4. Pogledi

Pogledi su SQL upiti koji kreiraju jedan korisnički pogled na podatke. Obično objedinjuju podatke iz više relacija u okviru jednog pogleda. Pogledi nisu materijalizovani. Za njih se ne kreira objekat u memoriji. Oni ostaju u obliku upita. Mogu se koristiti za kreiranje novih upita isto kao i relacije. Na taj način korisnik može koristiti za svoje upite strukturu koja mu je na neki način bliža od samih relacija. Osim toga, pogled ograničava korisnika na one relacije koje su mu potrebne. Osnovna naredba SQL-a za kreiranje pogleda podataka ima oblik:

```
CREATE VIEW naziv_pogleda  
[(nazivi_atributa_u_pogledu)]  
AS definicija pogleda (SELECT SQL izraz)
```

Primjer: Za bazu podataka biblioteke sa slike 2.3, kreirati pogled autori-naslovi

```
CREATE VIEW autori_naslovi  
(IME_A, PUNI_NASLOV)  
AS  
SELECT IME_A, PUNI_NASLOV  
FROM naslov N, autor A, napisao N  
WHERE N.PUNI_NASLOV=P.PUNI_NASLOV AND  
A.ID_BROJ=P.ID_BROJ
```

Ovakav pogled skriva od korisnika da su autori i naslovi u bazi podataka smešteni u različitim relacijama.

2.5.4.3. PRIMJER: DIO BAZE PODATAKA STUDENTSKE SLUŽBE – SQL UPITI

Dio baze podataka studentske službe dat je sljedećim relacijama:

student (BR_IND, IME, PREZIME, KUĆNI BROJ, TELEFON)
predmet (SIF_PR, NAZIV_PR)
prijave (BR_IND, SIF_PR, DATUM, OCJENA).

1. Prikazati sve podatke o studentima koji se zovu Jovan ili Luka, a stanuju u Jevrejskoj ulici.

```
SELECT *  
FROM student  
WHERE (IME="Jovan" OR IME="Luka") AND  
       ULICA="Jevrejska"
```

2. Prikazati ime, prezime i adresu studenata koji stanuju u ulicama čiji nazivi sadrže riječ "knez".

```
SELECT IME, PREZIME ULICA, KUCNI_BROJ  
FROM studenti  
WHERE ULICA CONTAINING "knez"
```

3. Navesti sve podatke o studentima čije prezime počinje sa Jov, a stanuju u Knez Mihajlovoj ulici.

```
SELECT *  
FROM studenti  
WHERE PREZIME STARTING WITH "Jov" AND  
       ULICA="Knez Mihajlova"
```

4. Prikazati sve podatke o predmetima čiji naziv počinje riječju teorija.

```
SELECT *  
FROM predmeti  
WHERE NAZIV_PR LIKE "teorija%"
```

5. Prikazati sve podatke o studentima čije prezime počinje sa Hadži, a završava se sa ić.

```
SELECT *  
FROM studenti  
WHERE PREZIME LIKE "Hadzi%ić"
```

6. Prikazati imena svih studenata koji nemaju telefon.

```
SELECT IME, PREZIME  
FROM studenti  
WHERE TELEFON IS NULL
```

7. Izračunati srednju ocjenu studenata na ispitu iz Baza podataka u januarskom roku.

```
SELECT AVG (OCJENA)  
FROM prijave  
WHERE SIF_PR="4BP" AND  
DATUM="januar"
```

8. Izračunati ukupan broj kandidata na ovom ispitu.

```
SELECT COUNT (*)  
FROM prijave  
WHERE SIF_PR="4BP" AND  
DATUM="januar"
```

9. Prikazati ime, prezime i broj indeksa studenata koji su položili neki ispit u januarskom roku.

```
SELECT IME, PREZIME, BR_IND  
FROM studenti  
WHERE BR_IND=ANY  
(SELECT BR_IND  
FROM prijave  
WHERE OCJENA>5 AND  
DATUM="januar")
```

2.5.4.4. PRIMJER: RELACIONA ALGEBRA – SQL UPITI

Za relacije:

$r(A \ B \ C)$	$s(A \ B \ D)$
$a_1 \ b_1 \ c_1$	$a_1 \ b_1 \ d_3$
$a_1 \ b_1 \ c_2$	$a_2 \ b_1 \ d_1$
$a_2 \ b_1 \ c_1$	$a_2 \ b_1 \ d_3$
$a_1 \ b_1 \ c_3$	
$a_3 \ b_2 \ c_1$	
$a_2 \ b_1 \ c_2$	

naći $\Pi_{A,B,D} (\sigma_{D=d_3}(r \times s))$. Predstaviti ovaj upit putem SQL-a.

$r \times s = (A \ B \ C \ D)$

$a_1 \ b_1 \ c_1 \ d_3$
$a_1 \ b_1 \ c_2 \ d_3$
$a_2 \ b_1 \ c_1 \ d_1$
$a_2 \ b_1 \ c_1 \ d_3$
$a_1 \ b_1 \ c_3 \ d_3$
$a_2 \ b_1 \ c_2 \ d_1$
$a_2 \ b_1 \ c_2 \ d_3$

$\sigma_{D=d_3}(r \times s) = (A \ B \ C \ D)$

$a_1 \ b_1 \ c_1 \ d_3$
$a_1 \ b_1 \ c_2 \ d_3$
$a_2 \ b_1 \ c_1 \ d_3$
$a_1 \ b_1 \ c_3 \ d_3$
$a_2 \ b_1 \ c_2 \ d_3$

$\Pi_{A,B,D} (\sigma_{D=d_3}(r \times s)) = (A \ B \ D)$

$a_1 \ b_1 \ d_3$
$a_2 \ b_1 \ d_3$

```

SELECT A, B, D
FROM r,s
WHERE r.A=s.A AND
      r.B=s.B AND
      s.D=d3

```

2.5.5. LOGIČKO PROJEKTOVANJE RELACIONIH BAZA

Kada se projektuje neka relaciona baza, često se nameće više alternativnih načina dekompozicije cjelokupne baze u različite relacione šeme. Neke od tih dekompozicija su pravilne (valjane, dok druge nisu. U okviru ovog poglavlja razmatraćemo problem logičkog projektovanja baze podataka i dati nekoliko algoritama koji formalizuju taj proces projektovanja. Međutim, prije toga treba da vidimo šta to konstituše loše projektovanu bazu podataka. Ovo ćemo razmotriti na primjeru:

isporučilac (IME_I, ADRESA, #ARTIKLA, CIJENA)

U ovakvoj bazi podataka možemo uočiti nekoliko anomalija:

1. Dupliciranje informacija: za svaki artikal koji neki isporučilac dostavlja moramo upisati adresu tog isporučioca.
2. Potencijalna nekonzistentnost baze (anomalija ažuriranja).
Ove anomalije dolaze kao posljedica dupliciranja informacija. Naime, prilikom promjene adrese nekog isporučioca, može se desiti da se ta promjena unese u jednu n-torku za tog isporučioca, a da pri tom u ostalim n-torkama za istog isporučioca adresa ostane ista. Time se dobija nekonzistentnost baze (baza ne odražava realno stanje).
3. Anomalija unošenja: ne možemo predstaviti informaciju o imenu i adresi isporučioca ukoliko on ne šalje nijedan artikal. Mogli bismo vještački stvoriti n-torku sa definisanim vrijednostima za IME_I, ADRESA i nedefinisanim vrijednostima #ARTIKLA, CIJENA i takvu n-torku unijeti u relaciju isporučilac. Međutim postojanje nedefinisanih vrijednosti ne dozvoljava se za ključne attribute, a #ARTIKLA jeste ključni atribut ove relacije. Ovo važi stoga što semantika algebarskih relacionih operatora, kao što je spajanje, nije definisana za neke vrijednosti. Dalje, ukoliko bi isti takav isporučilac za koga smo unijeli n-torku sa nepoznatim vrijednostima počeo da isporučuje artikle, prilikom unošenja informacije o tome često bi bilo zaboravljeno da se eliminiše prethodno unesena n-torka sa nepoznatim vrijednostima.
4. Anomalija brisanja inverzna je anomaliji unošenja. Naime, ako izbrišemo sve artikle koje isporučilac šalje, tada nesvjesno gubimo informaciju o samom isporučiocu i njegovoj adresi.

U ovom primjeru svi gore navedeni problemi nestaju ukoliko se data relacija r predstavi pomoću dvije manje relacije.

isporučilac (IME_I, ADRESA)

isporučuje (IME_I, #ARTIKLA, CIJENA)

Kod ovakve dekompozicije ne postoji dupliciranje adrese, jer za svakog isporučioca adresa se drži u jednoj n-torki relacije isporučilac. Dakle, nema anomalije ažuriranja. Možemo predstaviti informaciju o imenu i adresi nekog isporučioca iako taj isporučilac ne šalje nijedan artikal (dakle, nema anomalije unošenja). Na kraju možemo izbrisati sve artikle datog isporučioca, a da pri tome zadržimo informaciju o njegovom imenu i adresi, čime se izbjegava anomalija brisanja.

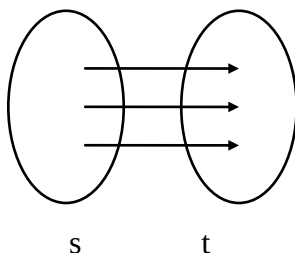
Međutim, ova dekompozicija ima takođe svoje nedostatke. A nedostaci se ogledaju u tome što sada za upite moramo vršiti operaciju spajanja, koja je veoma kompleksna i spora, dok smo kod ranije baze sve upite mogli da postavimo isključivo putem operatora selekcije i projekcije. Zbog svega ovoga u ovom poglavlju ćemo razmatrati probleme optimalne dekompozicije šeme relacione baze i probleme određivanja kada se neka dekompozicija isplati, a kada ne.

2.5.5.1. FUNKCIJSKE ZAVISNOSTI

Neformalno funkcijska zavisnost $X \rightarrow Y$ važi na relacionoj šemi R , pri čemu $X, Y \subseteq R$, ako vrijednosti atributa iz X jednoznačno određuju attribute iz Y .

Prema [13], formalno pojam funkcijske zavisnosti $X \rightarrow Y$ može se definisati na sljedeći način.

Definicija 1: Funkcijska zavisnost $X \rightarrow Y$ važi na R ako $\forall r(R)$ i $s = \prod_x(r(R))$ i $t = \prod_y(r(R))$ postoji jednoznačno preslikavanje iz skupa n -torki s u skup n -torki t .



Definicija 2: Funkcijska zavisnost $X \rightarrow Y$ važi na R ako $\forall r(R)$ $\Pi_y(\sigma_{x=x}(r(R)))$ proizvodi najviše jednu n -torku.

Primjer: Sadržaj relacije *raspored*, koja sadrži podatke o rasporedu letjenja, dat je kao:

raspored (#LETA, DATUM, POLAZAK, PILOT)

3	27.03.'99.	18 ^h	MM
7	29.03.'99.	8 ^h	SS
3	03.04.'99.	18 ^h	RR

Potrebno je utvrditi da li važi funkcijska zavisnost $\{\#LETA\} \rightarrow \{POLAZAK\}$ na relaciji sa ovim n -torkama.

Napravimo selekciju po broju leta 3 nad ovom relacijom. Tada dobijamo relaciju:

raspored_let3 (#LETA, DATUM, POLAZAK, PILOT)

3	27.03.'99	18 ^h	MM
3	03.04.'99	18 ^h	RR

Zatim, napravimo projekciju na POLAZAK. Tada dobijamo relaciju:

raspored_let3_polazak (POLAZAK)

18^h

Prema drugoj definiciji funkcijske zavisnosti, možemo utvrditi da data funkcijska zavisnost važi na ovakvoj relaciji (dobili smo samo jednu n -torku).

Međutim, funkcijska zavisnost $\{\#LETA\} \rightarrow \{PILOT\}$ ne važi na ovakvoj relaciji jer bi se za let broj 3 projektovanjem na atribut PILOT dobile dvije različite n -torke.

Ali zato važi funkcijska zavisnost $\{\#LETA, DATUM\} \rightarrow \{PILOT\}$. Inače, svi atributi su funkcijski zavisni od primarnog ključa, jer je primarni ključ jedinstven u relaciji, tj. ne ponavlja se.

2.5.5.1.1. OSOBINE FUNKCIJSKIH ZAVISNOSTI

Za funkcijske zavisnosti važi:

1. ako funkcijska zavisnost $X \rightarrow Y$ važi na R tada $\forall r(R)$ i $\forall t_1, t_2 \in r$
 $t_1(X) = t_2(X) \rightarrow t_1(Y) = t_2(Y)$
2. funkcijske zavisnosti su vremenski nepromjenljiva svojstva, tj. ako $X \rightarrow Y$ važi na R , tada $\forall r(R)$, funkcijska zavisnost važi i za svako stanje te relacije r .

Postojanje funkcijskih zavisnosti na nekoj relacionoj šemi R ne može se utvrditi na osnovu trenutnog sadržaja neke relacije $r(R)$. Naime, ako je sadržaj relacije r takav da $\forall t_1, t_2 \in r, t_1(X) = t_2(X) \rightarrow t_1(Y) = t_2(Y)$ ne možemo tvrditi da $X \rightarrow Y$ važi na R jer se kasnije može unijeti n -torka koja će narušiti ovo svojstvo. Međutim, na osnovu trenutnog sadržaja neke relacije r može se zaključiti da funkcijska zavisnost $X \rightarrow Y$ ne važi na R . Naime, ako je sadržaj relacije r takav da $\exists t_1, t_2 \in r, t_1(X) = t_2(X)$ i $t_1(Y) \neq t_2(Y)$ tada sigurno možemo tvrditi da X ne povlači Y .

Primjer : Ako je data relacija r sa sljedećim n -torkama:

$$\begin{array}{l} r(A \ B \ C \ D) \\ a_1 b_1 c_1 d_2 \\ a_2 b_1 c_2 d_1 \\ a_1 b_1 c_1 d_1 \end{array}$$

Funkcijska zavisnost $AB \rightarrow C$ možda važi, dok $AB \rightarrow D$ sigurno ne važi.

Dakle, funkcijske zavisnosti jesu tvrđenja koja se ne mogu dokazati, ali za očekujemo da će ona uvijek biti ispunjena. Jedini način da se utvrdi postojanje neke funkcijske zavisnosti jeste pažljivo posmatranje atributa neke relacione šeme i značenja tih atributa.

Primjer: isporučilac (IME_I, ADRESA, #ARTIKLA, CIJENA)

Na ovoj relaciji važi funkcijska zavisnost $\{IME_I\} \rightarrow \{ADRESA\}$ jer ime isporučioca jednoznačno određuje adresu. Drugim riječima, ne može se desiti da se u ovoj relaciji pojavi n-torka koja ima isto ime isporučioca kao neka druga, a različitu adresu.

Isto tako važi funkcijska zavisnost $\{IME_I, \#ARTIKLA\} \rightarrow \{CIJENA\}$ jer je IME_I, #ARTIKLA primarni ključ ove relacije.

2.5.5.1.2. LOGIČKE IMPLIKACIJE FUNKCIJSKIH ZAVISNOSTI

Definicija 1: Za relaciju $r(R)$ kažemo da zadovoljava funkcijsku zavisnost na R ako se u njoj ne mogu naći dvije n -torke t_1, t_2 takve da $t_1(X) = t_2(X)$ i $t_1(Y) \neq t_2(Y)$.

Definicija 2: Neka je F skup funkcijskih zavisnosti koje važe na relacionoj šemi R . Kažemo da F logički implicira funkcijsku zavisnost $X \rightarrow Y$ na R , u oznaci $F \models X \rightarrow Y$, ako svaka relacija koja zadovoljava funkcijske zavisnosti iz F zadovoljava i funkcijsku zavisnost $X \rightarrow Y$. Tada kažemo da funkcijska zavisnost $X \rightarrow Y$ logički slijedi iz F .

Primjer: Dokazati da za $F = \{X \rightarrow Y, Y \rightarrow Z\}$ važi $F \models X \rightarrow Z$

Dokaz: Pretpostavimo suprotno, neka X ne povlači Z . Tada za neku relaciju r koja zadovoljava funkcijske zavisnosti iz F važi $\exists t_1, t_2 \in R$ takve da $t_1(X) = t_2(X)$ i $t_1(Z) \neq t_2(Z)$.

Da li tada $t_1(Y) = t_2(Y)$? Ne, jer tada Y ne povlači Z .

Da li tada $t_1(Y) \neq t_2(Y)$? Ne, jer tada X ne povlači Y .

Dobili smo kontradikciju.

Definicija3: Pod zatvorenjem skupa funkcijskih zavisnosti u oznaci F^+ , podrazumijeva se skup funkcijskih zavisnosti koje logički slijede iz F .

$$F^+ = \{X \rightarrow Y \mid F \models X \rightarrow Y\}$$

Pomoću zatvorenja skupa funkcijskih zavisnosti može se definisati primarni ključ neke relacije.

Definicija4: $X \subseteq R$ jeste primarni ključ na R , na kome važi skup funkcijskih zavisnosti F , ako:

1. $X \rightarrow R \in F^+$
2. nijedan $Y \subset X$ ne zadovoljava svojstvo $Y \rightarrow R \in F^+$

Pravila na osnovu kojih se iz datog skupa funkcijskih zavisnosti F izvode druge funkcijske zavisnosti koje logički slijede iz F , nazivaju se **aksiome Armstronga**. Dokazuje se da su aksiome Armstronga saglasne (što znači da se na osnovu njih izvode samo funkcijske zavisnosti iz F^+) i kompletne (što znači da se na osnovu njih mogu izvesti sve funkcijske zavisnosti iz F^+).

(A1) pravilo refleksivnosti

$X, Y \subseteq R$, $Y \subseteq X$ tada $X \rightarrow Y$ važi na R

(A2) pravilo aditivnosti

$X, Y, Z \subseteq R$ tada – ako važi $X \rightarrow Y$, onda važi i $XZ \rightarrow YZ$

(A3) pravilo tranzitivnosti

$X, Y, Z \subseteq R$ tada $X \rightarrow Y$ i $Y \rightarrow Z \Rightarrow X \rightarrow Z$

(A4) pravilo proširenja

$X, Y, Z \subseteq R$ tada – ako važi $X \rightarrow Y$ i $X \rightarrow Z$, onda važi i $X \rightarrow YZ$

(A5) pravilo pseudotranzitivnosti

$X, Y, Z, W \subseteq R$ i važi $X \rightarrow Y$ i $YW \rightarrow Z$, onda $XW \rightarrow Z$

(A6) pravilo dekompozicije

$X, Y, Z \subseteq R$ i $Z \subset Y$ tada $X \rightarrow Y$, onda $X \rightarrow Z$

Aksiome A4–A6 mogu se izvesti iz A1–A3.

Teorema: X je natključ na $r(A_1, A_2, \dots, A_n)$ ako i samo ako $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$

Dokaz:

$(\Rightarrow)$ X je natključ, tada po definiciji 4 iz prethodnog paragrafa važi $X \rightarrow A_1, A_2, \dots, A_n$. Na osnovu (A6) važi $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$.

$(\Leftarrow)$ $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$, na osnovu (A4) važi $X \rightarrow A_1, A_2, \dots, A_n$, a onda na osnovu definicije 4 slijedi da je X natključ.

Primjer: $R=ABC$, $F=\{AB \rightarrow C, C \rightarrow B\}$, pokazati da $AC \rightarrow B$, $AC \rightarrow ABC$, $AC \rightarrow C$

1) $C \rightarrow B$, (A2): $AC \rightarrow AB$, (A6): $AC \rightarrow B$

2) $AC \rightarrow B$, (A6): $AC \rightarrow ABC$

3) (A1): $AC \rightarrow C$

Definicija 5: Zatvorenjem skupa atributa $X \subseteq R$, u odnosu na skup funkcijskih zavisnosti F koji važi na R, u oznaci $(X^+)_F$ nazivamo skup svih atributa $A \in R$ za koji se funkcijske zavisnosti $X \rightarrow A$ mogu izvesti iz F putem aksioma Armstronga.

Osim putem aksioma Armstronga, postoji i algoritam putem koga se može efektivno naći zatvorenje skupa atributa:

Algoritam1: Izračunavanje $(X^+)_F$

ULAZ: relaciona šema R, skup funkcijskih zavisnosti F na R, skup atributa $X \subseteq R$

IZLAZ: $(X^+)_F$

METODA: iterativna

1) $X^{(0)} = X$

2) $X^{(i+1)} = X^{(i)} \cup \{A \mid \exists Y, \exists Z (Y \rightarrow Z \in F \text{ i } Y \subseteq X^{(i)} \text{ i } A \in Z)\}$

3) ponavljati korak dva sve dok se nijedan novi atribut ne može unijeti u $X^{(i)}$

Primjer: $R = ABCD$ $F = \{A \rightarrow C, B \rightarrow CD, AC \rightarrow B\}$

$$1) X = \{A\} (X^+)_F = ?$$

$$X^{(0)} = X = \{A\}$$

$$X^{(1)} = \{A\} \cup \{C\} = \{AC\}$$

$$X^{(2)} = \{AC\} \cup \{B\} = \{ABC\}$$

$$X^{(3)} = \{ABC\} \cup \{CD\} = \{ABCD\}$$

$$(X^+)_F = \{ABCD\}$$

$$2) X = \{B\}$$

$$X^{(0)} = X = \{B\}$$

$$X^{(1)} = \{B\} \cup \{CD\} = \{BCD\}$$

$$X^{(2)} = \{BCD\} \cup \emptyset = \{BCD\}$$

$$(X^+)_F = \{BCD\}$$

Pomoću zatvorenja skupa atributa može se utvrditi da li važi neka funkcijska zavisnost čija je lijeva strana taj skup atributa.

Da li važi $A \rightarrow BD$? Da jer

$$B, D \in A^+ = ABCD$$

$$A \rightarrow B, A \rightarrow D, (A4): A \rightarrow BD$$

Da li važi $B \rightarrow A$? Ne jer

$$A \text{ ne pripada } B^+ = BCD$$

2.5.5.1.3. PRIMJERI

1. $R = GUP$, $F = \{P \rightarrow G, GU \rightarrow P\}$, koristeći aksiome Armstronga dokazati da $PU \rightarrow PGU$.

Rješenje: $P \rightarrow G$, (A2) $PU \rightarrow PGU$

2. $R = ABC$, $F = \{AB \rightarrow C, C \rightarrow B\}$, koristeći aksiome Armstronga pokazati da $AB \rightarrow BC$, $AC \rightarrow C$, $AC \rightarrow B$, $ABC \rightarrow AB$, $AB \rightarrow ABC$.

Rješenje: a) $AB \rightarrow C$, (A2) $AB \rightarrow BC$

- b) (A1) $AC \rightarrow C$
- c) $C \rightarrow B$, (A2) $AC \rightarrow AB$, (A6) $AC \rightarrow B$
- d) (A1) $ABC \rightarrow AB$
- e) (A2) $AB \rightarrow ABC$

3. $F = \{AB \rightarrow C, C \rightarrow A, BC \rightarrow D, ACD \rightarrow B, D \rightarrow EG, BE \rightarrow C, CG \rightarrow BD, CE \rightarrow AG\}$ Za $X = BD$ naći $(X^+)_F$.

Rješenje: $X^{(0)} = X = \{BD\}$
 $X^{(1)} = \{BD\} \cup \{EG\} = \{BDEG\}$
 $X^{(2)} = \{BDEG\} \cup \{C\} = \{BCDEG\}$
 $X^{(3)} = \{BCDEG\} \cup \{A\} = \{ABCDEG\}$
 $(X^+)_F = \{ABCDEG\}$

4. $F = \{A \rightarrow C, B \rightarrow DC, AC \rightarrow B\}$, $(A^+)_F = ?$

Rješenje: $X^{(0)} = X = \{A\}$
 $X^{(1)} = \{A\} \cup \{C\} = \{AC\}$
 $X^{(2)} = \{AC\} \cup \{B\} = \{ABC\}$
 $X^{(3)} = \{ABC\} \cup \emptyset = \{ABC\}$
 $(X^+)_F = \{ABC\}$

2.5.5.2. DEKOMPOZICIJA RELACIONIH ŠEMA

Dekompozicija relacije šeme R jeste zamjena te šeme skupom relacionih šema:

$$\rho = \{R_1, R_2, \dots, R_n\}$$

pri čemu $R_1 \cup R_2 \cup \dots \cup R_n = R$, $(R_i \cap R_j)$ može da bude (a ne mora) prazan skup za $i, j = 1, \dots, n, i \neq j$.

Motivacija za vršenje dekompozicije jeste da se eliminišu problemi dupliciranja informacija, anomalija ažuriranja, unošenja i brisanja kod relacija. Ove dekompozicije obavlja projektant baze podataka prilikom logičkog projektovanja baze.

Međutim, nije svaka dekompozicija dobra niti poželjna. Generalno gledano, postoje dva problema prilikom dekompozicije, a to su:

1. mogućnost gubljenja informacija i
2. mogućnost gubljenja funkcijskih zavisnosti prilikom dekompozicije.

2.5.5.2.1. NORMALNE FORME

Normalna forma predstavlja neku vrstu redukcije nad relacijama ili relacionim šemama u cilju eliminisanja anomalija. Pod anomalijama podrazumijevamo: anomalije unošenja, ažuriranja, brisanja, kao i dupliciranje informacija.

Sve relacije koje smo do sada razmatrali zadovoljavale su uslov da svaki zapis za svako svoje polje sadrži isključivo prostu vrijednost, tj. atom. Drugim riječima, pojedine vrijednosti n-torki nisu ni skupovi ni relacije. Ovakvo ograničenje nad relacijama naziva se **prva normalna forma – 1NF**. Relaciona baza je u 1NF ako je svaka njena relacija u 1NF. Prva normalna forma je značajna iz dva razloga. U prvom redu, funkcijske zavisnosti ne bi bilo moguće izraziti ukoliko relacije nisu u 1NF. Ponovićemo još jednom značaj funkcijskih zavisnosti koje predstavljaju integritetska ograničenja između različitih atributa. Značajne su za pravilno modeliranje stvarnosti. Druga prednost 1NF jeste u tome što omogućava pogodno i efikasno ažuriranje.

Primjer: Uzmimo jednu relaciju koja nije u 1NF

IME	GOD_ROĐ
(Marija, Ivan)	1967
(Petar)	1969
(Janko, Ana)	1970

Ukoliko bi se ispostavilo da Ivan nije rođen 1967, već 1969. Godine, sa aspekta DBMS-a nastupila bi konfuzija. Naime, DBMS bez dodatne informacije ne bi mogao da odredi da li da promijeni 1967 u 1969 ili da Ivana prebaci iz jedne grupe u drugu.

Za razliku od ovoga, relacija koja je u 1NF-u izgleda ovako:

IME	GOD_ROĐ
Marija	1967
Ivan	1967
Petar	1969
Janko	1970
Ana	1970

Ažuriranje Ivanove godine rođenja zahtijeva promjenu vrijednosti u samo jednoj n-torci baze.

Nedostatke prve normalne forme već smo pomenuli na primjeru relacije isporučuje (IME_I, ADRESA, #ARTIKLA), a to su: dupliciranje informacija i anomalija ažuriranja, unošenja i brisanja. Uočićemo da su anomalije koje nastupaju na ovoj relaciji posljedica jedne funkcijske zavisnosti koja važi na toj relaciji, a to je $\text{IME_I} \rightarrow \text{ADRESA}$

Prisustvo gore navedene funkcijske zavisnosti u stvari označava da su u relaciji *isporučuje* prisutna dva nezavisna odnosa:

- 1) odnos između isporučioaca i adrese, koji je $n:1$;
- 2) odnos između isporučioaca i artikla, koji je $n:m$.

Činjenica da su u relaciji *isporučuje* prisutna dva različita odnosa ogleda se u tome što atribut ADRESA zavisi i od ključa te relacije tj. IME_I, #ARTIKLA i od jednog njegovog dijela, tj. IME_I. Ovakva vrsta zavisnosti u kojoj desna strana te zavisnosti zavisi i od dijela lijeve strane funkcijske zavisnosti naziva se **parcijalna funkcijska zavisnost**

Parcijalna funkcijska zavisnost formalno se definiše ovako:

Def. 1: Neka su $X, Y \subseteq R$, ukoliko važi netrivialna funkcijska zavisnost $X \rightarrow Y$ i postoji $X' \subset X$ tako da $X' \rightarrow Y$ onda za $X \rightarrow Y$ kažemo da je u pitanju **parcijalna** funkcijska zavisnost.

Def. 2: Ukoliko funkcijska zavisnost $X \rightarrow Y$ nije parcijalna, tj. $\forall X' \subset X$ važi X' ne povlači Y , onda za funkcijsku zavisnost $X \rightarrow Y$ kažemo da je **potpuna**.

Napomena: Za funkcijsku zavisnost kažemo da je **netrivialna** ako desna strana nije podskup lijeve.

Primjer: Funkcijska zavisnost $\text{IME_I}, \# \text{ARTIKLA} \rightarrow \text{ADRESA}$ iz prethodnog primjera je parcijalna jer i $\text{IME_I} \rightarrow \text{ADRESA}$.

Rekli smo već da je funkcijska zavisnost $X \rightarrow X'$ trivijalna ako je $X' \subseteq X$. Tada parcijalnu funkcijsku možemo predstaviti kao kompoziciju $X \rightarrow X' \rightarrow Y$, gdje je $X \rightarrow X'$ trivijalna, a $X' \rightarrow Y$ netrivijalna zavisnost.

Druga normalna forma – 2NF uvedena je u cilju eliminisanja parcijalnih zavisnosti putem dekompozicije relacionih šema. Neka je $Y \subset X$, R neka relaciona šema, takva da nijedan njen atribut nije dio nijednog ključa nad R (ove attribute možemo zvati **sporednim**, tj. Y je skup sporednih atributa). Kažemo da je relacija $r(R)$ u 2NF ako svaki atribut iz Y (sporedni atribut) potpuno funkcijski zavisi od svakog ključa relacije r . Relaciona baza se nalazi u 2NF ukoliko je svaka njena relacija u 2NF.

Primjer: Relacija *isporučuje* nije u 2NF jer sporedni atribut ADRESA parcijalno zavisi od IME_I , $\# \text{ARTIKLA}$ zato što postoji funkcijska zavisnost $\text{IME_I} \rightarrow \text{ADRESA}$.

Međutim, ukoliko bismo izvršili dekompoziciju relacije *isporučuje* na dvije relacije

isporučilac (IME_I , ADRESA)
šalje (IME_I , $\# \text{ARTIKLA}$),

onda bi takva baza bila u 2NF, jer su obje ove relacije u 2NF. Prva relacija *isporučilac* je u 2NF zato što je njoj jedini ključ IME_I , a *šalje* je u 2NF jer nema sporednih atributa.

Uopšteno gledano, svaka relacija čiji se ključevi sastoje od samo jednog atributa ili nema sporednih atributa, automatski je u 2NF.

Sama definicija 2NF ne zabranjuje postojanje parcijalnih zavisnosti ključa od ključa.

Primjer: Neka je data relacija $r(\text{ABCDEF})$ sa skupom f.z. $F = \{ \text{ABC} \rightarrow \text{DE}, \text{DE} \rightarrow \text{ABC}, \text{DE} \rightarrow \text{F}, \text{AB} \rightarrow \text{D}, \text{E} \rightarrow \text{C} \}$.

Ključevi ove relacije su $K_1=ABC$ jer $(ABC)^+ = ABCDEF$, $K_2 = DE$ jer $(DE)^+ = ABCDEF$, $K_3 = ABE$ jer $(ABE)^+ = ABCDEF$.

Kod ovakve relacije F je jedini sporedni atribut koji potpuno funkcijski zavisi od svakog ključa relacije R.

Međutim, u ovoj relaciji postoje parcijalne zavisnosti atributa ključa od ključa. Npr. važi $ABC \rightarrow D$ po (A6) iz $ABC \rightarrow DE$. S druge strane, imamo funkcijsku zavisnost $ABC \rightarrow DE$; dakle, funkcijska zavisnost $ABC \rightarrow D$ je parcijalna zavisnost. Isto tako postoji parcijalna zavisnost $DE \rightarrow C$ jer $E \rightarrow C$.

Ako neka relacija $r(R)$ nije u 2NF, onda postoji takva dekompozicija relacije $r(R)$ u kojoj su sve relacije šeme u 2NF, pri čemu nema gubitka funkcijskih zavisnosti, niti gubitka informacija. Proces normalizacije u 2NF je iterativan. Naime, relacija $r(R)$ razbija se u dvije nove relacije $r_1(R_1)$ i $r_2(R_2)$ da bi se eliminisala postojeća parcijalna zavisnost. U slučaju da neka od relacija r_1 ili r_2 nije u 2NF, nastavljamo normalizaciju sve dok se ne dobije skup relacija od kojih je svaka u 2NF. Ovaj postupak je konačan jer je skup atributa konačan.

Algoritam 2: 2NF normalizacija

ULAZ: relacija $r(R)$ koja nije u 2NF i skup funkcijskih zavisnosti F na R

IZLAZ: dekompozicija $\rho = \{r_1(R_1), r_2(R_2), \dots, r_i(R_i)\}$, gdje su $\forall R_i$ u 2NF.

METODA: ako relacija $r(R)$ nije u 2NF, onda postoji parcijalna funkcijska zavisnost $X \rightarrow Y$, gdje se Y sastoji od sporednih atributa, a X je neki ključ relacije $r(R)$. Neka je Z skup svih atributa koji nisu ni u X ni u Y, (Z može biti prazan skup). X možemo predstaviti kao $X = X'X''$, gdje je $X' \rightarrow Y$ zavisnost koja $X \rightarrow Y$ čini parcijalnom. Tada se relacija $r(XYZ)$ zamjenjuje sa dvije nove relacije $r_1(XZ)$ i $r_2(X'Y)$. Ukoliko bilo koja od relacija r_1 ili r_2 nije u 2NF, tada isti proces ponavljamo za r_1 ili r_2 ili obje. Proces se ponavlja sve dotle dok sve relacije ne budu u 2NF.

Primjer : *isporučuje (IME_I, ADRESA, #ARTIKLA)*

$F = \{IME_I \rightarrow ADRESA\}$

PFZ: IME_I, #ARTIKLA→ADRESA

$$X = \{ \text{IME_I}, \# \text{ARTIKLA} \} \quad X' = \{ \text{IME_I} \}$$

$$Y = \{ \text{ADRESA} \}$$

$$Z = \emptyset$$

$r_1(\text{IME_I}, \# \text{ARTIKLA})$ 2NF – nema sporednih

$r_2(\text{IME_I}, \text{ADRESA})$ 2NF – PK od jednog atributa

Nažalost, ograničenja koja nameće 2NF nisu dovoljno stroga, pa relacije koje su u 2NF i dalje ispoljavaju anomalije.

Primjer: Relacija *radnik* (#RADNIKA, PREDUZEĆE, ADRESA) na kojoj važe funkcijske zavisnosti :

$$F = \{ \# \text{RADNIKA} \rightarrow \text{PREDUZEĆE}, \text{PREDUZEĆE} \rightarrow \text{ADRESA} \}.$$

Ovo znači da neki radnik može da radi u samo jednom preduzeću i da svako preduzeće ima samo jednu adresu. Sa druge strane u jednom preduzeću može raditi više radnika, pa važi da PREDUZEĆE ne povlači #RADNIKA.

Jedini ključ ove relacije jeste atribut #RADNIKA, pa je relacija u 2NF. Međutim, date funkcijske zavisnosti svjedoče da se unutar relacije radnik suštinski razlikuju dva skupa entiteta:

radnik (#RADNIK, PREDUZEĆE)

preduzeće (PREDUZEĆE, ADRESA).

Ova činjenica da u jednoj te istoj relaciji imamo dva različita skupa entiteta dovodi do anomalija u početnoj relaciji *radnik*. Dupliciranje informacija je posljedica toga što adresu preduzeća moramo ponoviti za svakog radnika u tom preduzeću. Anomalija ažuriranja nastaje ako adresu promijenimo u jednoj n-torci, a zaboravimo da to uradimo za ostale n-torke koje se odnose na to preduzeće. Anomalija unošenja nastaje ukoliko želimo da predstavimo preduzeće i adresu, a nemamo informaciju o radnicima tog preduzeća. Anomalija brisanja nastaje ako izbrišemo sve radnike, jer tada gubimo informacije o samom preduzeću i njegovoj adresi.

Anomalije koje još uvijek postoje u relacijama koje su u 2NF jesu posljedica tranzitivnih funkcijskih zavisnosti, kao što su:

#RADNIKA $\rightarrow$ PREDUZEĆE $\rightarrow$ ADRESA,

u prethodnom primjeru.

Def. 3: Neka su $X, Y, Z \subseteq R$, a R relaciona šema. Kažemo da postoji **tranzitivna funkcijska zavisnost** $X \rightarrow Y$, ako postoji funkcijska zavisnost $X \rightarrow Y$, pri čemu Y ne povlači X , i postoji funkcijska zavisnost $Y \rightarrow Z$ koja je netrivialna.

Ovim nismo rekli da funkcijska zavisnost $X \rightarrow Y$ mora biti netrivialna. Ukoliko je $X \rightarrow Y$ trivialna funkcijska zavisnost, onda je $X \rightarrow Z$ parcijalna funkcijska zavisnost.

Relacija se nalazi u **trećoj normalnoj formi – 3NF** ukoliko nijedan sporedni atribut nije tranzitivno zavisn od bilo kojeg ključa te relacije. Baza podataka se nalazi u 3NF ukoliko je svaka njena relacija u 3NF.

Lema: Ako je relacija u 3NF, onda je ona neminovno u 2NF.

Dokaz: Pretpostavimo da je neka relacija u 3NF, a nije u 2NF. Pošto se relacija ne nalazi u 2NF, onda postoji parcijalna funkcijska zavisnost $X \rightarrow Y$, gdje je X neki ključ relacije, a Y skup sporednih atributa. Tada postoji $\exists X' \subset X$ takav da $X' \rightarrow Y$. Pošto je $X' \subset X$, to važi da $X \rightarrow X'$. Takođe važi da X' ne povlači X jer bi u suprotnom X' bio ključ. Dakle, na ovoj relaciji imamo sljedeći niz funkcijskih zavisnosti $X \rightarrow X' \rightarrow Y$, pa je $X \rightarrow Y$ tranzitivna zavisnost na ovoj relaciji, što je u kontradikciji sa pretpostavkom da je relacija u 3NF.

Sa druge strane, ukoliko je relacija u 2NF, ona ne mora biti u 3NF. Primjer toga imali smo kod relacije :

radnik (#RADNIKA, PRDUZEĆE, ADRESA).

Ukoliko relacija nije u 3NF, onda postoji takva dekompozicija te relacije koja ne uzrokuje gubitak informacije niti funkcijskih zavisnosti, a sve relacije u njoj su u 3NF.

Algoritam 3: 3NF normalizacija

ULAZ: relacija $r(R)$ i skup funkcijskih zavisnosti F na R

IZLAZ: dekompozicija $\rho = \{r_1(R_1), r_2(R_2), \dots, r_n(R_n)\}$, gdje su sve $r_i(R_i)$, $i = 1, n$ u 3NF

METODA: Ako $r(R)$ nije u 3NF, onda postoji $X, Y, Z \subseteq R$ i $X \rightarrow Y$, Y ne povlači X , $Y \rightarrow Z$, pri čemu je X ključ relacije r , a Z podskup sporednih atributa. Neka je W skup svih atributa koji nisu ni u X ni u Y ni u Z . Tada se relacija $r(XYZW)$ zamjenjuje sa dvije relacije $r_1(XYW)$ i $r_2(YZ)$. Ukoliko bar jedna od ovih relacija i dalje nije u 3NF, onda vršimo njenu dekompoziciju na isti način sve dok ne dobijemo skup relacija od kojih je svaka u 3NF.

Primjer: *radnik* (#RADNIKA, PREDUZEĆE, ADRESA)

$F = \{ \#RADNIKA \rightarrow PREDUZEĆE, PREDUZEĆE \rightarrow ADRESA \}$

$\#RADNIKA \rightarrow ADRESA$ – tranzitivna funkcijska zavisnost

$X = \{ \#RADNIKA \}, Z = \{ ADRESA \}, Y = \{ PREDUZEĆE \}, W = \emptyset$

$r_1 = (\#RADNIKA, PREDUZEĆA)$

$r_2 = (PREDUZEĆE, ADRESA)$

Primjer: Za bazu podataka biblioteke iz poglavlja 2.4.10. izvršiti 3NF normalizaciju.

1. *napisao* (ID_BROJ, PUNI_NASLOV, GODINA), važe sljedeće funkcijske zavisnosti

$ID_BROJ, PUNI_NASLOV \rightarrow GODINA$ – p.f.z

$PUNI_NASLOV \rightarrow GODINA$

Relacija nije u 2NF, a samim tim nije ni u 3NF. Zato vršimo dekompoziciju.

napisao (ID_BROJ, PUNI_NASLOV)
godina (PUNI_NASLOV, GODINA)

2. *n-k* (PUNI_NASLOV, BROJ_KNJIGE, BROJ_KOPIJA,
BROJ_SLOB_KOP)

PUNI_NASLOV, BROJ_KNJIGE → BROJ_KOPIJA, BROJ_SLOB_KOP – p.f.z.

PUNI_NASLOV → BROJ_KOPIJA, BROJ_SLOB_KOP

dekompozicija:

n-k (PUNI_NASLOV, BROJ_KNJIGE)
kopije (PUNI_NASLOV, BROJ_KOPIJA, BROJ_SLOB_KOP)

3. *izdato* (BROJ_KNJIGE, NAZIV_I, GOD_IZDANJA)

BROJ_KNJIGE, NAZIV_I → GOD_IZDANJA – p.f.z.

BROJ_KNJIGE → GOD_IZDANJA, $W = \emptyset$

dekompozicija:

izdato (BROJ_KNJIGE, NAZIV_I)
god_izd (BROJ_KNJIGE, GOD_IZDANJA)

2.5.6. NEDOSTACI RELACIONOG MODELA

Relacioni modeli osim prednosti, koje smo već naveli, imaju i svoje nedostatke. Jednostavna struktura znači da je relacioni model semantički siromašan, te da se svojstva stranog sistema koji se modelira bazom teško predstavljaju konceptima takvog modela. Idealno normalizovan relacioni model trebalo bi da se sastoji od međusobno nezavisnih relacija, što naravno nije moguće, jer izolovani objekti praktično ne postoje. Veze između objekata su teško uočljive. Isto tako, očuvanje uslova integriteta prepušta se aplikacionim programerima koji nedovoljno dobro poznaju logičku bazu.

U cilju prevazilaženja ovih nedostataka u novije vrijeme relacioni model se proširuje tako da se oslanja na semantički bogatije modele (E-R dijagrami su grafička komponenta relacionog modela) i ima mogućnosti upotrebe objektnih koncepata za upravljanje kompleksnim tipovima podataka. Takođe se na nivou modela uvode dinamički koncepti u okviru kojih su definisana pravila za očuvanje integritetskih ograničenja i poslovne logike. Ova ograničenja i pravila ugrađuju se u samu bazu, pa su programi oslobođeni od kompleksnih procedura zaštite integriteta, a integritet baze ne zavisi od programa.

2.5.7. FUNKCIJE RELACIONOG DBMS-A

Relacioni DBMS je zadužen da vodi računa o sljedećim funkcijama:

1. upravljanje datotekama i strukturama koje omogućavaju brz i efikasan pristup podacima;
2. upravljanje baferima koji sadrže tekuću informaciju o bazi podataka, prenošenju sa diska u memoriju;
3. omogućavanje restauracije konzistentnosti baze u slučaju kada nastupi krah sistema;
4. omogućavanje konkurentnog pristup podacima i sinhronizacija rada korisnika sa istim podacima;
5. prevođenje korisničkih upita u termine razumljive fizičkoj organizaciji baze podataka;
6. vršenje optimizacije redosljeda izvršenja operacija u upitima;
7. zaštita baze podataka od neautorizovanog pristupa.

U cilju ostvarivanja ovih funkcija DBMS održava više različitih datoteka:

1. **Glavne datoteke**, tj. datoteke zapise koje sadrže samo podatke.
2. **Indekse**, koji omogućavaju brz pristup podacima u glavnoj datoteci, i to sekvencijalnim pretraživanjem za vrijednosti nekih atributa.

Obično se indeksi uvode za atribut koji predstavlja ključ tabele, mada se mogu uvesti i za ostale attribute, na osnovu kojih se često obavljaju pretraživanja. Indeks se sastoji od elemenata oblika <ključ, pokazivač>, uređenih po vrijednostima ključa, pri čemu svaki pokazivač u stvari predstavlja adresu odgovarajućeg zapisa u glavnoj memoriji.

Kada se pretražuje neki zapis sa ključem K , prvo se pristupa indeksnoj datoteci, gdje se vrši sekvencijalno pretraživanje za sve $K_i < K$. Kada nađemo na $K_{i0} = K$, pokazivač za ovaj element indeksa ukazuje nam na mjesto traženog zapisa u glavnoj datoteci. Na ovaj način pristup podacima se znatno ubrzava.

3. **Sistemske datoteke** koje se koriste za opis fizičke i logičke strukture baze podataka. Skup ovih datoteka sačinjava **rječnik baze podataka**.
4. **Statističke datoteke** u kojima se čuvaju statističke informacije o drugim datotekama. Ove informacije se koriste za optimizaciju pristupa podacima.

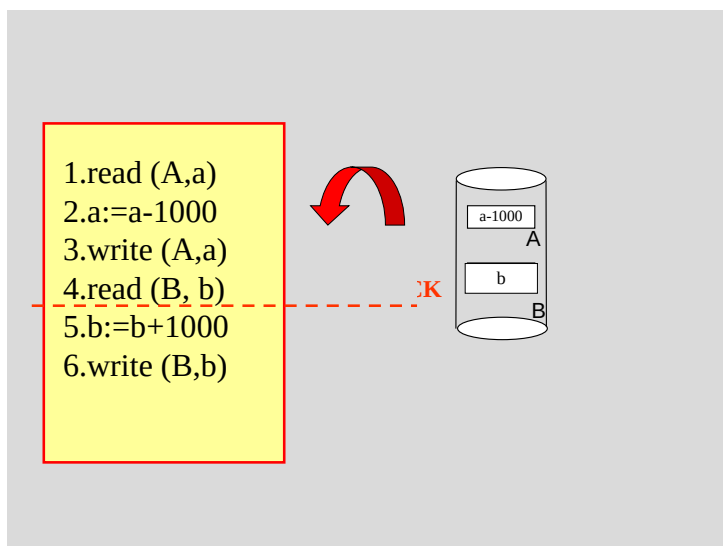
2.5.7.1. POJAM KONZISTENTNOSTI I TRANSAKCIJE

Ako se baza nalazi u stanju koje odgovara realnom stanju sistema koji se modelira, tada kažemo da je ona u **konzistentnom stanju**. Na primjer, ako je stanje na nekom računu u banci različito od onog koji se trenutno nalazi u bazi podataka, onda je baza u nekonzistentnom stanju. Operacija ažuriranja baze podataka prevodi bazu iz jednog stanja u drugo (dodajući, brišući ili ispravljajući podatke u bazi). Stoga su ove operacije sposobne da naruše integritet podataka, to jest da dovedu bazu podataka u nekonzistentno stanje.

Razmotrimo na primjer operaciju ažuriranja koja sa računa A skida određenu svotu i dodaje na račun B.

Pretpostavimo da se usljed nekog razloga operacija prekinula poslije treće instrukcije. Tada bi u bazi podataka sa računa A bila skinuta svota, dok se na računu B iznos ne bi povećao (slika 2.6.). Ovo je jedno nekonzistentno stanje baze podataka, do kojeg je dovela nekompletirana operacija ažuriranja.

Zato se ovakve operacije izvršavaju atomski, to jest ili se izvršavaju u cjelini ili nikako. Ovakav niz akcija koje se izvode zajedno kao cjelina naziva se **transakcija**. Transakcija, dakle, predstavlja operaciju kojom se baza podataka prevodi iz jednog konzistentnog stanja u drugo, takođe konzistentno stanje. Transakcija može da se poništi (ROLLBACK), i tada se u bazi ispostavlja staro konzistentno stanje (stanje prije započinjanja transakcije). Ukoliko se transakcija uspješno završi (COMMIT), uspostavlja se novo konzistentno stanje koje se može razlikovati od prethodnog.

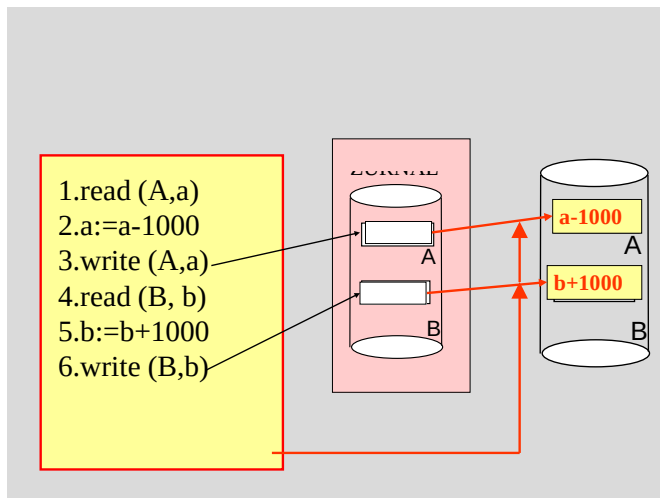


Slika 2.6. Transakcija i krah sistema

2.5.7.2. RESTAURACIJE KONZISTENTNOSTI BAZE PODATAKA

Do prekida operacije u prethodnom primjeru može doći iz više razloga, na primjer zbog nestanka struje (krah sistema). Isto tako, logička greška, to jest greška koju napravi programer u samom programu, može dovesti do pomenutog prekida. Na primjer, nakon upisivanja nove vrijednosti na računu A, ispostavlja se da traženog računa B uopšte nema u bazi podataka. Program se prekida, ali je sa računa A trajno izgubljena svota od 1000 jedinica.

Dio DBMS-a koji se bori sa pomenutim sistemskim krahovima naziva se sistem za restauraciju konzistentnosti baze. U te svrhu DBMS održava jednu sistemsku datoteku koja se naziva **žurnal**. U datoteku se unose sve promjene koje transakcije treba da izvrše u bazi podataka. Tek kada je transakcija u potpunosti završena, sistem će prenijeti te promjene iz sistemskog žurnala na odgovarajuća mjesta u bazi (slika 2.7.). U slučaju da je u međuvremenu nastupio sistemski krah, vrši se ROLLBACK započete transakcije, to jest vrijednosti iz žurnala ne prenose se u bazu podataka, već se poništavaju.

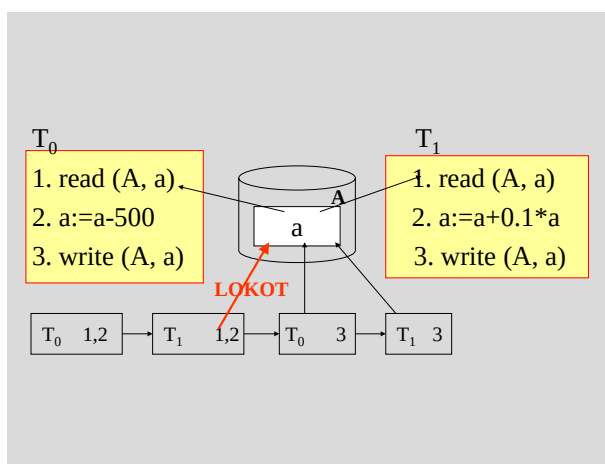


Slika 2.7. Sistem za restauraciju konzistentnosti baze

2.5.7.3. KONKURENTNI PRISTUP

U uslovima multiprogramiranja nekoliko transakcija može da radi istovremeno sa istim podacima u bazi. Usljed toga DBMS mora kontrolisati njihovo izvršavanje u cilju izbjegavanja konfliktnih situacija, tj. u cilju očuvanja konzistentnosti baze.

Na primjer, pogledajmo dvije transakcije T_0 i T_1 na slici 2.8.:



Slika 2.8. Zaključavanje podataka kod konkurentnog pristupa

Pošto je u pitanju multiprogramiranje, obje transakcije se izvršavaju paralelno. Pretpostavimo da su se izvršile prva i druga instrukcija transakcije T_0 , zatim prva i druga instrukcija transakcije T_1 , a zatim treća instrukcija transakcije T_0 , pa treća instrukcija transakcije T_1 (slika 2.8.). Dobili bismo nekonzistentno stanje. Naime, obje transakcije su uzele početnu vrijednost sa računa A, a promjena računa A koju je izvršila transakcija T_0 biće poništena zadatom *write* instrukcijom, koju obavlja transakcija T_1 . Drugim riječima, odraziće se samo promjena koju je izazvala transakcija T_1 , dok će promjena koju je trebalo da izvrši transakcija T_0 biti zauvijek izgubljena. Dakle, potrebno je bilo prvo umanjiti račun za 500, pa tek onda ga povećati za 10%.

DBMS probleme konkurentnog pristupa najčešće rješava **zaključavanjem podataka**. Pri tom se podacima pridružuju takozvani **lokoti** (tj. zajedničke promjenljive u memoriji). Tada transakcija ne može pristupiti podatku ukoliko ne zadobije odgovarajući lokot nad tim podatkom. S druge strane, transakcije mogu zadobiti traženi lokot jedino ako ga druge transakcije otpuste.

Tako u prethodnom primjeru transakcija T_0 zadobija lokot nad podatkom A, i zadržava ga sve dok ne završi svoju *write* instrukciju nad tim podatkom. Za sve to vrijeme transakcija T_1 mora čekati na željeni lokot (slika 2.8.). Kada T_0 izvrši i treću instrukciju, otpušta lokot, i tek tada transakcija T_1 može započeti svoje izvršavanje. Na taj način je zaštićen integritet podataka, tj. Dobija se konzistentno stanje nakon izvršavanja obje operacije.

Napomenimo da je konkurentni pristup vrlo česta pojava u radu sa bazama podataka (na primjer, u banci više operatera istovremeno mijenja neki račun).

2.6. PITANJA I ZADACI ZA VJEŽBU

1. a) Šta je skup odnosa, a šta relacija? U čemu je osnovna razlika?
- b) Neka su $E_1 = \{e_1, e_2, e_3\}$ i $E_2 = \{f_1, f_2, f_3\}$ skupovi entiteta. Definišite jedan skup odnosa R sa kardinalnim brojem $n:m$ između ova dva skupa entiteta.
- c) Ako je skup entiteta E_1 iz prethodne tačke predstavljen sljedećom relacijom:
 $r(A \ B \ C)$, a skup entiteta E_2 relacijom $s(D \ E)$

$a_1 \ b_2 \ c_1$	(e_1)	$d_1 \ e_1$	(f_1)
$a_2 \ b_1 \ c_1$	(e_2)	$d_2 \ e_1$	(f_2)
$a_1 \ b_1 \ c_1$	(e_3)	$d_3 \ e_1$	(f_3) ,

prema sadržaju relacija odrediti domene atributa, primarne ključeve za ove relacije i odgovarajućom relacijom predstaviti skup odnosa koji ste definisali u prethodnoj tački.

2. a) Šta je odnos, a šta skup odnosa? U čemu je osnovna razlika?
- b) Neka su $E_1 = \{e_1, e_2, e_3\}$ i $E_2 = \{f_1, f_2\}$ skupovi entiteta. Definišite jedan skup odnosa R sa kardinalnim brojem $1:n$ između ova dva skupa entiteta.
- c) Eliminišite n -torke iz skupa odnosa za kardinalni broj $1:1$.
3. a) U čemu je osnovna razlika između natključa i kandidatskog ključa?
- b) Neka skup entiteta E_1 ima attribute A, B, C sa domenima $D_1 = \{a_1, a_2\}$, $D_2 = \{b_1, b_2, b_3\}$, $D_3 = [18, 60]$. Neka su entiteti $e_1 = \langle a_1, b_3, 23 \rangle$, $e_2 = \langle a_2, b_2, 14 \rangle$, $e_3 = \langle a_3, b_3, 50 \rangle$. Da li entiteti mogu biti ovako definisani i, ako ne mogu, zašto?
- c) Ako su entiteti $e_1 = \langle a_2, b_3, 20 \rangle$, $e_2 = \langle a_2, b_2, 20 \rangle$, $e_3 = \langle a_1, b_1, 20 \rangle$ da li je $K = \{A, B\}$ kandidatski ključ ovog skupa entiteta. Ako nije, objasnite zašto nije?

4. a) U čemu je osnovna razlika između operacija za ažuriranje relacija i operacija relacione algebre? Koje SQL naredbe (izrazi) odgovaraju jednima, a koje drugima?

b) U kojim slučajevima operacija dodavanja n-torke (INSERT) može da propadne?

c) Date su relacije $r(A \ B \ C \ D)$ i $s(B \ D \ E \ F)$

$a_1 \ b_2 \ c_1 \ d_2$	$b_1 \ d_2 \ e_1 \ f_2$
$a_2 \ b_1 \ c_1 \ d_2$	$b_1 \ d_1 \ e_1 \ f_2$
$a_1 \ b_1 \ c_1 \ d_2$	$b_1 \ d_2 \ e_2 \ f_1$

Naći $\Pi_{ACD}(\sigma_{C=c1}(r \times s))$

d) Predstaviti prethodni upit putem SQL-a.

5. a) Definišite operaciju relacione algebre *spajanje*.

b) Koristeći osobine operatora relacione algebre, dokažite da važi :

$$\sigma_F(\Pi_x(r) \times \Pi_x(s)) = \Pi_x(\sigma_F(r)) \times \Pi_x(\sigma_F(s))$$

d) Proverite tačnost ovog izraza za relacije $r(A \ B \ C)$ i $s(A \ C \ D)$,

$a_1 \ b_2 \ c_1$	$a_1c_2 \ d_1$
$a_1 \ b_3 \ c_2$	$a_1c_2 \ d_3$
	$a_3 \ c_3 \ d_3$

Predikat F glasi $C = c_2$ i $X = AC$.

6. Dio baze podataka proizvodnje u jednom poslovnim informacionom sistemu sadrži sljedeće relacije:

dokumenta (JED_BR_DOK, VRS_DOK, DATUM_DOK)

radna_mj (SIF_RM, NAZIV_RM, BR_RADNIKA)

trebovanje (JED_BR_DOK, SIF_RM, SKLADISTE)

materijali (SIFRA_MAT, NAZIV_MAT, VRSTA_MAT, KOLICINA, JED_MJERE, JED_CIJENA_KOSTANJA)

stavke_trebovanja (JED_BR_DOK, SIF_RM, SIFRA_MAT, KOLIČINA)

- a) Nacrtati odgovarajući E-R dijagram za ovaj dio baze podataka.
- b) Dopuniti E-R dijagram tako da sadrži informaciju o tome koji se proizvodi proizvode na određenom radnom mjestu, kao i o standardnim utrošcima pojedinih materijala za proizvodnju pojedinih proizvoda.
7. Putem SQL-a izraziti sljedeće upite bazi iz prethodnog zadatka:
- Naći sva trebovanja materijala *brašno tip A* u periodu od 10. 03. 2002. do 10. 04. 2002.
 - Naći spisak materijala i njihovu količinu koji su trebovani iz skladišta *S1* za radno mjesto *proizvodnja hljeba* dana 20. 04. 2002.
 - Naći ukupnu količinu materijala *So* koja je iz skladišta *S1* trebovana za radno mjesto *proizvodnja hljeba* u martu 2002.
 - Naći spisak materijala sa standardnim utrošcima za proizvodnju proizvoda *biohljeb*.
 - Trebovanje radnog mjesta sa šifrom *1A* prema dokumentu broj 23-001 treba stornirati (poništiti). Ažurirati bazu podataka.
 - Naći ukupan broj dokumenata putem kojih su trebovani materijali za proizvodnju u toku 2001. godine.
8. Dio baze podataka prodaje u jednom poslovnom informacionom sistemu sadrži sljedeće relacije :

dokumenta (BR_DOK, VRSTA_DOK, DATUM_DOK, ORG_JED_NASTANKA, IZRADIO_ZAPOSLANI)

otpremnica (BR_DOK, VRSTA_DOK, OPIS_OTPREME)

zalihe (SIFRA_ZALIHE, NAZIV_ZALIHE, VRSTA_ZALIHE, KOLICINA, JED_MJERE, JED_CIJENA_KOSTANJA)

stavke_otpremnice (BR_DOK, VRSTA_DOK, SIFRA_ZALIHE, KOLIČINA)

izlazna_faktura (BR_DOK, VRSTA_DOK, OPIS_FAKTURE)

- c) Nacrtati odgovarajući E-R dijagram za ovaj dio baze podataka.
- d) Dopuniti E-R dijagram tako da sadrži informaciju o tome po kojoj otpremnici je faktura nastala, kao i informacije o stavkama fakture (kojem kupcu je fakturisana otprema i po kojoj prodajnoj cijeni su fakturisani odgovarajući proizvodi iz otpremnice).
9. Putem SQL-a izraziti sljedeće upite bazi iz prethodnog zadatka:
- Naći sva dokumenta vrste *otpremnic*a i *faktura* koji su nastali 10. 03. 2001.
 - Naći sve različite vrste zaliha koje su izdate na osnovu otpremnica nastalih u organizacionoj jedinici *Stovarište 1* u periodu od 01. 01. 2001. do 01. 03. 2001.
 - Naći ukupnu cijenu za vrstu zalihe *gotovi proizvodi* i naziv zalihe *šećer u kristalu*.
 - Naći sve stavke fakture nastale na osnovu otpremnice broj 5367400. Stavke treba da sadrže broj fakture, kupca, šifru zalihe, naziv zalihe, jedinicu mjere, količinu i prodajnu cijenu.
 - Otpremnica broj 6367500 je poništena i treba je izbrisati. Ažurirati bazu podataka.
10. Data je relaciona šema $R = PUVQ$ i skup funkcijskih zavisnosti $F = \{PU \twoheadrightarrow V, P \twoheadrightarrow Q, U \twoheadrightarrow Q\}$.
- a) Koristeći aksiome Armstronga, pokazati da važi $PU \twoheadrightarrow Q$.
- b) Koristeći algoritam zatvorenja skupa atributa, pokazati da važi $PU \twoheadrightarrow PUVQ$.
- c) Naći sve kandidatske ključeve na relacionoj šemi R i objasniti zašto su oni KK.
11. Data je relaciona šema $R = ABCDE$ i skup funkcijskih zavisnosti $F = \{E \twoheadrightarrow BD, C \twoheadrightarrow BE, ED \twoheadrightarrow AC\}$.
- d) Koristeći aksiome Armstronga, pokazati da važi $E \twoheadrightarrow A$.

- e) Koristeći algoritam zatvorenja skupa atributa, pokazati da važi $AC \rightarrow ABCDE$.
- f) Naći sve kandidatske ključeve na relacionoj šemi R i objasniti zašto su oni KK.

12. Data je relacija:

zalihe_materijala (SIF_MAGACINA, SIF_MATERIJALA,
OPIS_MATERIJALA, JEDINICA_MJERE, KOLICINA)

- a) Definišite parcijalnu funkcijsku zavisnost. Kada za neku relaciju kažemo da je u 2NF?
- b) Da li je relacija *zalihe_materijala* u 2NF? Ako nije, objasnite zašto nije i izvršite 2NF normalizaciju date relacije.
- c) Objasnite anomaliju brisanja na ovom primjeru.

13. Data je relacija:

stavke_fakture (DOK_ID, SIF_DOB, SIF_ROBE, NAZIV_DOB,
ADRESA, NAZIV_ROBE, KOLIC_ROBE_U_STAVCI, VRIJ_STAVKE)

- a) Kada za neku relaciju kažemo da je u 1NF, 2NF i 3NF?
- b) Da li je relacija *stavke_fakture* u 3NF? Ako nije, objasnite zašto nije i izvršite 3NF normalizaciju date relacije.
- c) Objasnite anomaliju dupliciranja podataka i anomaliju unošenja na ovom primjeru.

14. a) Šta su indeksi i čemu služe?

- b) Na primjeru objasnite pojam transakcije.
- c) Šta je žurnal datoteka i čemu služi?
- d) Šta je konkurentni pristup? Na primjeru objasnite kako SUBP rješava problem konkurentnog pristu