

1. Istorijski razvoj automatizovane obrade podataka

60-tih godina u oblasti automatske obrade podataka tada se govorilo o datotekama i skupovima podataka. U vrijeme 60-tih godina javlja se potreba obrade na računaru velike količine podataka koji se susreću prije svega u finansijsko-ekonomskim problemima. Kraj 60-tih karakteriše organizacija podataka nastala korišćenjem novih memorijskih jedinica i metoda pristupa.

Automatizovana obrada (posmatrana u odnosu na podatke) se deli na dva perioda ili dva pristupa:

- Klasična obrada podataka
- Konceptija baza podataka

2. Klasična obrada podataka

Klasična obrada podataka ne isključuje i korišćenje zajedničkih datoteka. Između skupova datoteka koje se koriste samo za pojedine aplikacije postoji dosta istih podataka obično sa različitim formatom zapisa ili metodama pristupa.

- **Prednosti**
 - Jednostavnost projektovanja i realizacije
- **Nedostaci**
 - Nepovezanost aplikacija
 - Redundantnost podataka
 - Čvrsta povezanost programa i datoteka
- Način "nastajanja" modela podatka – strukture datoteka
 - Model podataka (struktura datoteka) je nastajao na osnovu zahteva programa

3. Konceptija baza podataka

- **Kad se pojavila?**
 - Krajem 60ih godina.
- **Šta je bio cilj?**
 - Sve podatke jednog informacionog sistema treba integrisati u jednu fizičku strukturu podataka. Dolazi se do koncepta šeme baze podataka
 - Svi aplikacioni programi pri obradi baze podataka koriste standardizovane softverske rutine nazvane **Sistem za Upravljanje Bazom Podataka**
 - Model podataka nastaje nezavisno od programa
 - Konceptualni model nezavisan od tekućih obrada – pogodan i za nepredviđene buduće obrade

4. SUBP

- Sve obrade podataka koriste standardizovane softverske rutine SUBP. SUBP je softverski sistem koji obezbeđuje osnovne funkcije:
 - Pretraživanje
 - Pamćenje
 - Ažuriranje
 - Višestruko paralelno korišćenje istog skupa podataka
 - Pouzdanost
 - Sigurnost
- Neki poznatiji komercijalni SUBP:
 - Oracle
 - MS SQL Server
 - DB II
 - Postgres

- MySQL

5. Poslovni zadaci Administratora baze podataka

- Administrator baze podataka je osoba odgovorna za specificiranje, projektovanje, implementiranje, efikasan rad i održavanje baze podataka.
- Administrator baze podataka radi na sledećim poslovima:
 1. Sa korisnicima pri ustanovljavanju zahtjeva za bazu podataka
 2. Koristi jezik za opis podataka za definisanje baze podataka
 3. Radi sa programerima čiji program treba da pristupi bazi
 4. Odgovoran je za punjenje baze podacima
 5. Nadgleda rad baze podataka
- *Stvaranje uslova za efikasnu obradu podataka*
- *Konzistentno sprovođenje promena*
- *Redovna izrada sigurnosnih kopija baze podataka*
- *Regulisanje prava pristupa podacima i sl.*
- Osnovni ciljevi administratora BP su:
 1. Integritet BP
 2. Bezbednost i efikasnost.

6. Integritet BP

Stanje u kojem su sve vrednosti podataka korektne, u smislu da:

- Odslikavaju stanje realnog sveta i to do zadate tačnosti i pravovremenosti
- Poštuju pravila uzajamne konzistentnosti

Održavanje integriteta BP podrazumijeva proveru integriteta i oporavak iz bilo kojeg nekorektnog stanja koje može biti otkriveno.

7. Bezbednost BP

Podrazumeva sprečavanje ili zaštitu od:

- Neovlašćenog pristupa podacima
- Namernog i neovlašćenog uništavanja ili menjanja istih

Bezbednost BP je potrebna zbog zaštite kako od nenamjernih, tako i od namernih pokušaja pristupa poverljivim podacima.

8. Principi rada sistema za upravljanje bazom podataka

- Ako aplikacioni program traži ili želi da smesti jedan podatak u bazu podataka, tada se to ostvaruje naredbom u jeziku koji vrši manipulaciju podataka. Rutine za upravljanje podacima analiziraju tu naredbu i dopunjuju je potrebnim vrednostima na osnovu opisa cele baze podataka. Zatim se OS-u daju IO uputstva, koji stupa u kontakt sa spoljnom memorijom i prenosi podatke u međumemoriju sistema. Odavde SUBP prenosi podatke u radne zone programa. Izvršiće se konverzija podataka ukoliko je to potrebno. Rezultat IO operacije SUBP signalizira aplikacionom programu i u slučaju povoljnog signala obrađuje podatke, odnosno nastavlja sa izvršavanjem. U interesu obezbeđenja dalje obrade podataka SUBP vodi računa o sadržaju IO bafera.

9. Modeli podataka

- Model podataka je skup međusobno povezanih podataka koji opisuju objekte, njihove veze i osobine, realnog sistema. Opis realnog sistema čiji se informacioni sistem projektuje pomoću termina modela podataka naziva se model baze podataka.

- Osnovni zadatak istraživanja u području modeliranja podataka je naći koncepte pomoću kojih će se izgraditi model koji reprezentira naše znanje o sistemu.
- **Redundansa**
 - Ponavljanje podataka u jednom informacionom sistemu.

10. Apstrakcije podataka

Pri izradi modela podataka koristi se sledeće tri apstrakcije:

- **Klasifikacija i uzorkovanje**
 - Apstrakcija u kojoj se skup sličnih objekata predstavlja jednom klasom objekata. Slični objekti su oni koji imaju ista obeležja, koji mogu da stupe u iste veze sa drugim objektima u sistemu i na koje se mogu primeniti iste operacije.
 - Npr. tip objekta STUDENT(BROJ_IND, PRE_IME, FAKULTET) reprezentuje sve studente jednog univerziteta.
 - Postupak detaljizacije za ovu apstrakciju se naziva **uzorkovanjem**, i predstavlja prikazivanje jednog pojavljivanja (uzorka) datog tipa ili klase.
- **Generalizacija i specijalizacija**
 - Apstrakcija gde se skup sličnih tipova objekata tretira kao novi generički (izvedeni) tip objekta na višem nivou.
 - Npr. STUDENT, RADNIK, PENZIONER se mogu generalizovati u tip objekta GRAĐANIN.
 - Inverzan postupak generalizaciji je specijalizacija.
- **Agregacija i dekompozicija**
 - Apstrakcija gde se skup tipova objekata i njihovih veza predstavlja novim agregiranim (izvedenim) tipom objekta na višem nivou.
 - Inverzan postupak je dekompozicija. Sam tip objekta može se posmatrati kao najniži nivo ove apstrakcije, kao agregacija njegovih tipova obeležja.

11. Komponente modela podataka

- **Strukturna**
 - Strukturna komponenta modela sadrži skup primitivnih koncepata i skup pravila za izgradnju složenijih koncepata.
- **Integritetna**
 - Integritetna komponenta modela podataka čini skup uslova integriteta (skup ograničenja). Uslovi integriteta ograničavaju sve pojave baze podataka nad jednom šemom na skup dozvoljenih (neprotivrečnih) pojava baze podataka isključujući one pojave koje ne zadovoljavaju neko od ograničenja.
- **Operacijska**
 - Operacijsku komponentu modela podataka čini skup koncepata koji omogućuju interpretaciju dinamičkih karakteristika skupa podataka. Koncept strukture i ograničenja reprezentuju statičke osobine realnog sistema, a operatori omogućuju da se stanje podataka u bazi podataka menja u skladu sa promenom stanja u realnom vremenu.

12. Generacije modela podataka

- **Modeli podataka I generacije**
 - Svaki konvencionalni programski jezik je zaseban model podataka.
- **Modeli podataka II generacije**
 - Generaciji pripadaju sledeći modeli:
 1. Funkcionalni
 2. Hijerarhijski
 3. Mrežni
 4. Klasični relacioni model

5. Warnier-ovi dijagrami

- **Modeli podataka III generacije**

Ovoj generaciji pripadaju sledeci modeli podataka:

- MOV
- Binarni model podataka
- Prošireni model podataka (RM-T)
- Model podataka relacionih mreža
- Petri-jeve mreže
- Model semantičkih asocijacija
- D-Graph

13. Model Objekti-Veze (ER - Model)

- **Konceptualna šema**

- Predstavlja model realnog sistema i njegove baze podataka koji ne zavise od konkretnog sistema za upravljanje BP. KŠ baze podataka prema MOV može se lako prevesti u šemu BP na kojoj je SUBP zasnovan.
- Po pravilu se predstavlja uz pomoć dijagrama objekti-veze (DOV).

- **Strukturna komponenta**

Primitivni koncepti u MOV su: objekat, veza i obilježje objekata. Objekti koji se indetifikuju nezavisno od ostalih objekata u sistemu nazivaju se osnovni objekti (karnel, jaki objekti).

- **Koncept veze**

- Veze u modelu opisuju način povezivanja objekata. Broj tipova objekata koji učestvuju u vezi definiše **red veze**. Postoje:
 - **Rekurzivna** je veza između dva objekta istog tipa.
 - **Binarne** čine povezani objekti različitog tipa.
 - **Veze višeg reda**- mogu se predstaviti pomoću binarnih veza ili se takva veza tretira kao agregacija.

- **Kardinalnost**

- Informacija o prirodi odnosa između objekata iz realnog svega u modelu podataka iskazuje se **kardinalitetom tipa veze**.

- **Nadtip – podtip**

- Generički tip objekta naziva se **nadtipom objekta**.
- Specijalizacijom se za neki tip objekta dobija njegov **podtip**. Podtipovi imaju neka njima specifična obeležja. Nadređenom tipu se pripisuju sva obeležja koja imaju njegovi podtipovi, dok se pojedinim podtipovima pripisuju obeležja karakteristična za taj podtip objekta.

- **Slabi objekat**

- Tip slabog objekta je takav tip objekta koji nema vlastitih obeležja za identifikaciju, već koristi i obeležja nekog drugog nadređenog tipa objekta koji može biti osnovni ili slabi objekat. Kažemo da je tip slab objekat ako **identifikaciono zavisi** od nekog drugog tipa objekta.

- **Agregacija**

- Kada je potrebno da tip veze ima obeležje ili da se uspostavi veza između tipa veze i tipa objekta tada tip veze postaje mešoviti tip **objekat-veza**.

14. Proces izrade MOV

Izrada MOV za dati realni sistem obično se odvija u sledećim koracima:

1. Izrada MOV po dijelovima
 2. Integracija dijelova u jednu cijelinu koju nazivamo **globalni model podataka**
 3. Uključivanje drugih sematičkih detalja u modelu
- **Integrisanje podmodela**

Kada je završena izrada DOV po delovima tj. podmodelima pristupa se integrisanju svih podmodela u jedan globalni (integralni) model podataka.

U procesu integracije dolazi se do novih saznanja i vrlo često do potrebe izmene u pojedinim podmodelima.

- **Kakav treba da bude globalni model podataka?**
 - Kompletan (sadrži sva znanja iz svih podmodela);
 - Neredundantan (ne sadrži višestruke predstave istih znanja);
 - Konzistentan (ne sadrži međusobno protivrečna znanja).

15. Relacioni model podataka?

- **Kad se pojavio, ko je tvorac?**
 - Konceptija relacionog modela podataka je prvi put objavljena u Codd-ovom članku iz 1970 godine.
- **Osnovni pojmovi**
 - Osnovni pojam relacionog modela podataka je **relacija**. Relacija se može posmatrati sa sledeća dva aspekta: **značenje i sadržaj**. Značenje relacije naziva se **intenzijom**, i formalno se iskazuje **šemom relacije**. Sadržaj relacije naziva se **ekstenzijom**, a iskazuje se **tabelom podataka** ili **pojavom šeme relacije**.
- **Šta je relaciona šema (šema relacije)?**
 - Značenje relacije naziva se **intenzijom**, i formalno se iskazuje **šemom relacije**
- **Šta je relacija?**
 - U relacionom modelu podataka **relacija** odgovara dvodimenzionalnoj tabeli u kojoj svaki red sadrži jednu n-torku, a svaka kolona elemente jednog domena.
- **Koje uslove mora da zadovolji skup obeležja da bi bio kandidat ključ šeme?**
 - Jedinственost (ne postoje dve n-torke koje imaju jednake vrednosti svih obeležja iz X)
 - Minimalnost (Ne postoji takav skup X' , $X' \subset X$)
- **Spoljni ključ**
 - Zajedničko obeležje u više tabela takvo da se pomoću njega iz date n-torke jedne relacije može pristupiti odgovarajućoj n-torci druge relacije.

16. Integritetna komponenta relacionog modela podataka

Integritetna komponenta relacionog modela podataka - Model baze podataka polazi od pretpostavke da su sve relevantne informacije poznate i da se mogu uneti u bazu podataka. U praksi ova pretpostavka često nije ispunjena.

Uslove integriteta koji se javljaju u jednoj relacionoj bazi podataka možemo podeliti u 2 grupe:

1. **Integritet objekta**- vezan je za pojam primarnog ključa šeme relacije. Definicija integriteta: vrednost primarnog ključa ne sme biti null;
2. **referencijalni integritet** - mora postojati n-torka koju referencira neko obeležje) i korisnička pravila integriteta.

17. Operacijska komponenta relacionog modela

- **Na čemu se zasniva?**
 - Operacijska komponenta omogućava manipulisanje podacima u bazi podataka, što se pre svega odnosi na postavljanje upita i ažuriranje baze podataka.
 - Smisao operacija nad relacionom BP sastoji se u promjeni pojave šeme baze podataka ili u formiranju novih relacija.
 - Svi postojeći jezici RBP razvijeni su na osnovu jednog od sledećih operatora:

1. Relacione algebre je model proceduralnog jezika i sastoji se od skupa operatora za rad sa relacijama.
 2. Relacioni račun -zasnovana na racunu predikata prvog reda i spada u neproceduralni jezik
- **Definicija operacija relacione algebre**
 - Operacije relacione algebre se mogu podeliti u dve grupe. Prvu grupu čine operacije matematičke teorije skupova:
 - Unija
 - Razlika
 - Presek
 - Dekartov proizvod
 - Drugu grupu operacija čine operacije koje su posebno razvijene za relacioni model podataka:
 - Selekcija
 - Projekcija
 - Spoj
 - **SQL**
 - SQL nije samo upitni jezik, već predstavlja kompletan jezik podataka koji sadrži jezike za:
 - Definisanje podataka
 - Ažuriranje
 - Kontrolu konzistentnosti
 - Konkurentni rad
 - Jezik za održavanje rečnika podataka
 - **Osnovne karakteristike SQL-a**
 - **Jednostavnost i jednoobraznost pri korišćenju**
Tebele se kreiraju jednom naredbom i odmah po kreiranju spremna je za korišćenje. Podaci se prikazuju u obliku tabele.
 - **Mogućnost interaktivnog i klasičnog programiranja**
SQL blokovi ugradjuju se u neki viši programski jezik (FORTRAN, COBOL, C...)
 - **Neproceduralnost**
SQL je u velikoj mjeri neproceduralan jer se njime definiše ŠTA se želi dobiti a ne KAKO se do rezultata dolazi.

18. Cilj relacione baze podataka

Osnovni ciljevi koncepcije baze podataka su:

- Podatke IS treba integrirati u jednu fizičku strukturu podataka. Dolazi se do koncepta **šeme baze podataka**.
- Sve obrade koriste standardizovane softverske rutine (**SUBP**)

19. Anomalije održavanja podataka

Jednostavno korišćenje i mjenjanje podataka podrazumijeva prije svega srečavanje **anomalije održavanja podataka**. Pod anomalijom održavanja podataka podrazumijevamo:

- **Anomalija dodavanja** javlja se kada su informacije o svojstvima jednog objekta memorisane u bazi podataka kao deo opisa nekog drugog objekta.
- **Anomalija brisanja** je inverzija anomalije dodavanja.
- **Anomalija izmjene** javlja se u slučaju kada promene podataka o jednom objektu treba izvršiti na više od jedne kopije podataka.
- U relacionom modelu podataka anomalije održavanja podataka se prevazilaze **normalizacijom** šeme relacija.

20. Normalizacija

- U najopštijem smislu **normalizacija** je postupak kojim se proizvoljna nenormalizovana relacija transformiše u skup manjih normalizovanih relacija.
- Bitna osobina koja se očekuje od normalizacije je **reverzibilnost**, odnosno, ne sme doći do gubitka informacija sadržanih u polaznoj relaciji.
- **Metode normalizacije**
 - **Vertikalna normalizacija** – iz relacione šeme se izdvajaju obeležja koja stoje u nedozvoljenim odnosima sa ostalim obeležjima u šemi.
 - Zasnovana na operacijama **projekcija** i **prirodni spoj**.
 - Operacijom projekcije relaciju razbijamo na dve ili više relacija.
 - Operacija **prirodni spoj** se koristi da bi se dokazala **reverzibilnost**, odnosno da je moguće rekonstruisati polaznu nenormalizovanu relaciju.
 - **Horizontalna normalizacija** – relacija se rastavlja na podskupove n-torki, odnosno fragmente relacije koji zadovoljavaju određene uslove.
 - Zasniva se na operacijama **selekcije** i **unije**.
 - Koristi se kod distribuiranih baza podataka.
- **Normalizacija dekompozicijom**
 - Započinje se od proizvoljne nenormalizovane relacione šeme i izvodi se u koracima.
 - Svakim korakom normalizacije relaciona šema prevodi se u višu normalnu formu
 - Tako da se polazni skup obeležja deli u dva podskupa i od svakog formira posebna šema relacije.
 - Svaki korak normalizacije mora biti reverzibilan.
- **Uslovi dekompozicije bez gubitka informacija**
 - Da bi dekompozicija relacije $p(P)$ na $r(R)$ i $s(S)$ bila reverzibilna, presek projekcija $r(R)$ i $s(S)$ ne sme biti prazan skup i obeležja koja čine presek moraju biti kandidati za ključ u bar jednoj od dve šeme relacija.
- **Vertikalna normalizacija dekompozicijom**
 - Zadatak normalizacije je da relacionu šemu prvo transformiše u 1NF, zatim u 2NF, 3NF i tako redom. Što je redni broj normalne forme veći to su i uslovi koje se postavljaju strožiji.
- **Uslov ispunjenosti 1NF**
 - Za relaciju čije sve domene sadrže samo atomarne vrednosti kažemo da je normalizovana, tj. nalazi se u 1NF.
- **Uslov ispunjenosti 2NF**
 - Relaciona šema R nalazi se u 2NF ako je svako neključno obeležje od R potpuno funkcionalno zavisno od kandidata ključa.
- **Uslov ispunjenosti 3NF**
 - Relaciona šema R nalazi se u 3NF ako je u 1NF i ako nijedno neključno obeležje u R nije tranzitivno zavisno od ključa od R .
- **Postupak svođenja šeme relacije koja nije u 2NF u skup šeme relacije koje su u 2NF**
 - Relacija se dekompozicijom rastavlja na dve ili više relacija, takvih da sva neključna obeležja tih relacija **potpuno** funkcionalno zavise od kandidata za ključ u datim relacijama.
- **Postupak svođenja šeme relacije koja nije u 3NF u skup šeme relacije koje su u 3NF**
 - Ukoliko šema relacije nije u 2NF, prije svođenja u 3NF prvo se svodi u 2NF. Ukoliko nema **tranzitivne zavisnosti** za šemu relacije se kaže da je u 3NF, u suprotnom šema se razbija na dve ili više šema, takvih da neključna obeležja **tranzitivno ne zavise** od kandidata ključa.

21. Prevođenje MOV u relacioni model podataka?

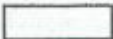






- **Zašto se vrsi prevođenje?**
 - S obzirom da za sada ne postoji komercijalno raspoloživ SUBP zasnovan na MOV to je u praktičnoj realizaciji potrebno model podataka izrađen prema MOV prevesti u neki od modela za koji postoji SUBP.

- **Opšta pravila**
 - Deo strukture MOV, odnosno DOV se predstavlja relacionom šemom.
 - Ograničenja, operacije i deo strukture MOV se predstavljaju operacijama za očuvanje integriteta definisanim nad relacionim modelom, a implementiraju se korišćenjem sredstava odabranog relacionog SUBP.
- **Pojedinačna pravila za objekte**
 - Svaki objekat iz DOV postaje šema relacije. Ime tipa objekta postaje ime šeme relacije. Obeležja objekta su obeležja šeme relacije. Za osnovne objekte identifikator objekta postaje primarni ključ šeme relacije.
 - Svaki **slab objekat** takođe postaje šema relacije. Identifikator slabog objekta čini identifikator nadređenog objekta i obeležja slabog objekta koja jedinstveno identifikuju pojavljivanje slabog objekta u okviru pojavljivanja njemu nadređenog objekta.
 - Objekat **nadtip** postaje šema relacije. ID nadtipa postaje ključ šeme relacije.
 - Objekat **podtip** takođe postaje šema relacije. ID **nadtipa** postaje obeležje šeme relacije podtipa i predstavlja ključ šeme relacije.
- **Pojedinačna pravila za veze**
 - Veze tipa **1:1** po pravilu nemaju obeležja. Sva obeležja koja bi eventualno mogla biti pripisana samoj vezi, zapravo su obeležja jednog od objekata koji čine tu vezu.
 - Veze se mogu ostvariti između dve ili više šema tako da spoljni ključevi budu sadržani u obeležjima samih relacija, ili pomoću dodatne relacija čija će obeležja biti strani ključevi relacija između koji se stvara veza.

22. Osnovne obrade transakcija

- **Sta je transakcija?**
Transakcija je vremenski urađeni niz nedeljivih radnji nad bazom podataka koje u celini ne remete uslove integriteta.
Transakcija predstavlja logičku jedinicu rada nad bazom podataka.
- **Naredbe SQL-a za rukovanje transakcijama**
Transakciona obrada odvija se pod kontrolom administratora transakcija kao dela programske podrške čije naredbe:
 1. **COMMIT** signalizira administratoru transakcija o uspešnom završetku transakcije
 2. **ROLLBACK** signalizira administratoru transakcija o neuspešnom završetku transakcije
 u potpunosti određuju način njegovog funkcionisanja.
Tačka sinhronizacije predstavlja graničnu tačku između dve serijske transakcije.
Sistem za upravljanje BP treba da omogući paralelni rad većem broju korisnika nad istom BP što znači paralelno izvršavanje više transakcija.
Javljaju se sledeći problemi:
 1. **Paralelno izvršavanje transakcija**
Uobičajeni način rješavanja problema PIT su protokoli zaključivanja. Osnovna ideja zaključavanja sastoji se u tome da transakcija koja zahteva garancije da neki objekat za koji je ona zainteresovana ne može biti ažuriran u nekom vremenu. Takav objekat transakcija zaključava i sprečava njegovo ažuriranje od strane drugih transakcija.
 2. **Problem izgubljenog azuriranja**
Transakcija A čita zapis R u trenutku t1. Transakcija B čita zapis R u trenutku t2. Transakcija A ažurira zapis R u trenutku t3 na osnovu vrijednosti pročitane u trenutku t1. Transakcija B ažurira zapis R u trenutku t4 na osnovu vrijednosti pročitane u trenutku t2 koje su iste sa vrijednostima pročitanim u trenutku t1 od strane transakcije A. Ažuriranje od strane transakcije A biva poništeno u trenutku t4, jer transakcija B ne uzima u obzir ažuriranja od strane transakcije A.
 3. **Problem zavisnosti od privremenih azuriranja**

- Osobine transakcija
- Protokoli zaključavanja
- Restauracija konzistentnog stanja baze podataka

GRAFIČKI PRIKAZ (simbol)	OPIS
	osnovni objekt
	slab objekt
	veza
	veza nadtipa i podtipa
	mešoviti tip objekta
	obeležje
	linija za povezivanje

Slika 7.1. Grafički simboli za crtanje DOV