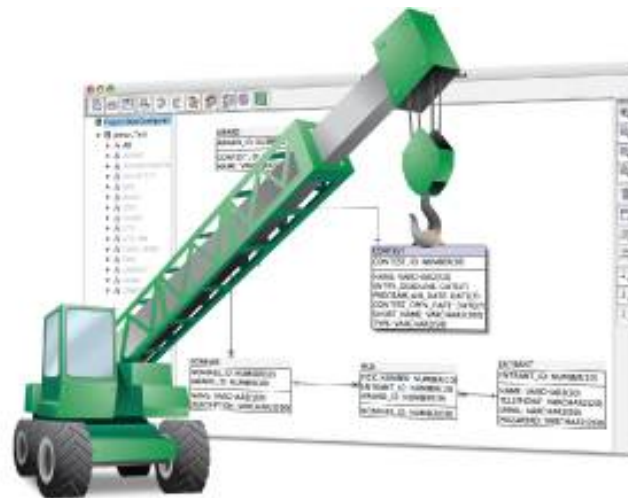
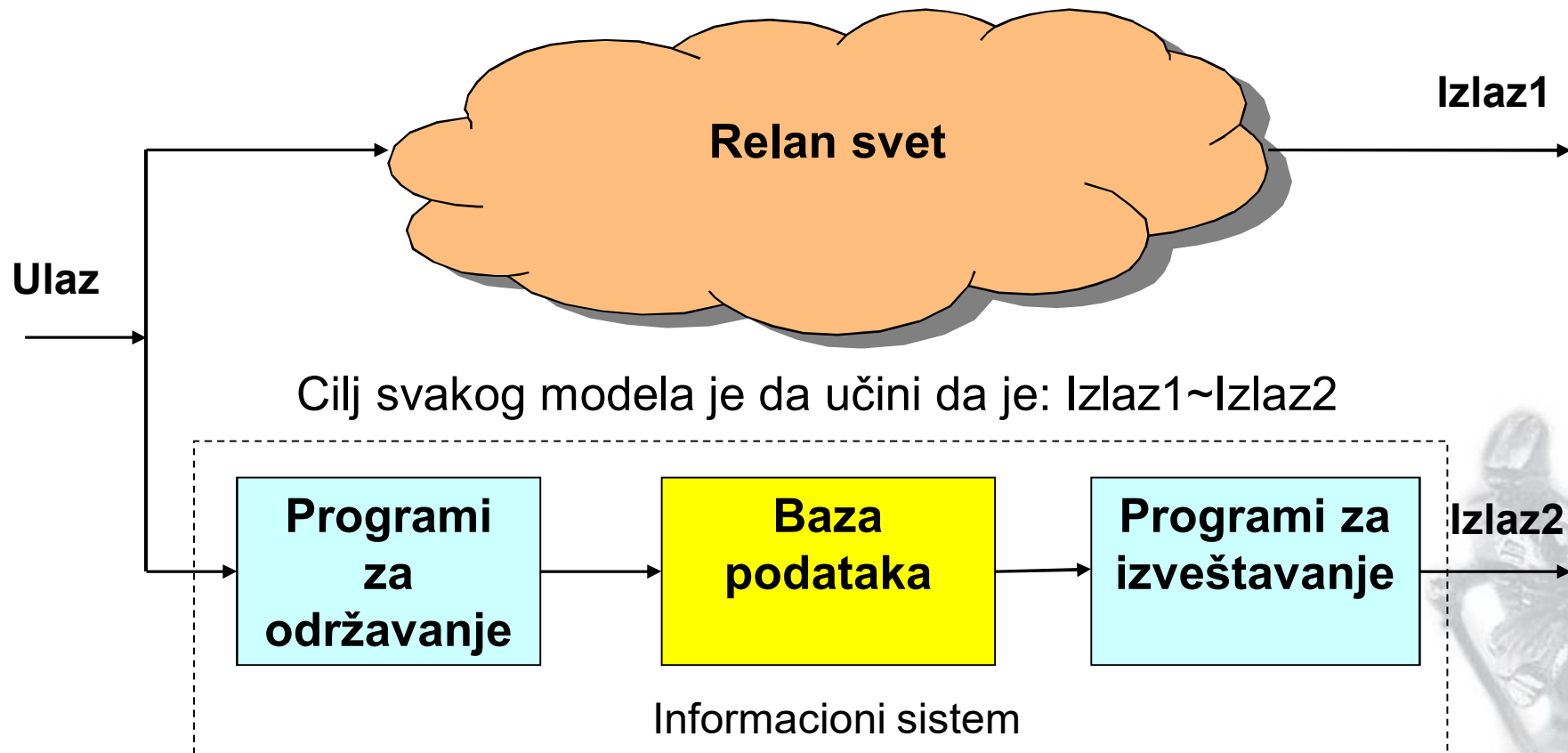


# Baze podataka

## Modelovanje





- Modelovane je postupak kojim se realni svet svodi na određeni broj podataka
- Podaci su apstrakcija realnosti - sredstva za kodiranje osobina objekata iz realnog sveta
- Izbor (selekcija) – izdvajanje bitnih objekata
- Dodeljivanje imena (za svaki objekat, vezu, atribut)
- Klasifikacija – nehomogenih objekata u homogene klase i tipove objekata



- Konceptualni model sadrži:
  - Strukturu podataka – statički opis stanja realnog sveta (**podaci** i **odnosi** između podataka)
  - Operacije – izražavaju dinamiku iz realnog sveta
  - Ograničenja (*constraints*) – Ograničenja u modelu koja su posledica ograničenja iz realnog sveta
- Model ne propisuje fizički oblik u kome se podaci čuvaju
- **Šema** je opis nekog skupa podataka, korišćenjem datog modela podataka



- Pod entitetom se podrazumeva sve što se može jednoznačno odrediti, identifikovati i razlikovati
- Svaki **entitet** ima svoje osobine – **atribute**
- **Domen atributa** je skup svih mogućih vrednosti koje atribut može poprimiti (slično tipu promenljive u programiranju)
- **Primarni ključ** je jedan ili više atributa čija vrednost jednoznačno određuje primerak entiteta



- Primer entiteta:  
STUDENT, FAKULTET, RADNIK, PROIZVOD,  
RAČUNAR, ISPIT, UPIS, itd.
- Instance su pojedinačna pojavljivanja entiteta
  - Instanca – Zapis u tabeli
  - Npr. U entitetu STUDENT instanca je konkretan student
- • Marko Marković, Beograd, Njegoševa 13, 011/234567
- • Petar Petrović, Novi Sad, Drapšinova 2, 021/123456
- ....

- Atribut je zajednička osobina koju poseduju svi entiteti jedne klase
- Svojstva objekta se opisuju preko atributa
- Primer:
  - Mogući atributi entiteta STUDENT:
    - BrInd, Ime, Prezime, Fakultet, Smer, Adresa, Telefon
- Broj atributa nije fiksna
- Relevantne attribute definiše kompetentna osoba
  - U skladu sa željenim zahtevima (informacijama)
  - Od toga zavisi upotrebljivost dobijenih informacija
- Atributi svih entiteta poprimaju određene vrednosti



- Primer: Vrednosti atributa entiteta STUDENT (jedan entitet iz klase studenata):
  - BrInd: 123/08
  - Ime: Marko
  - Prezime: Marković
  - Fakultet: DIR
  - Smer: Osiguranje
  - Adresa: Kneza Miloša 156, 11000 Beograd
- **Domen** – skup dozvoljenih vrednosti koje može da poprimi jedan atribut
- Obeležavanje domena:
  - $\text{Dom}(\text{Fakultet}) = \{\text{FPE}, \text{FTHM}, \text{DIR}, \text{DM}\}$





- Premalo atributa:
  - verodostojnost mala,
  - model jednostavan za predstavljanje i analizu,
  - ograničen je broj upotrebljivih informacija
- Previše atributa:
  - verodostojnost odlična,
  - kompleksnost velika,
  - manipulacija podacima teško izvodljiva,
  - dobijaju se konfuzne informacije.
- Zadatak projektanta: prepoznavanje prave mere pri modelovanju (izbor relevantnih atributa)



- Primer 1:
  - Sa aspekta zarada, atribut “adresa” nije relevantan
  - Sa aspekta organizovanja prevoza, to jeste relevantan atribut
- Primer 2:
  - Sa aspekta odeljenja za plate atribut “vrsta bolesti” nije relevantan
  - Sa aspekta zdravstvenog odeljenja istog preduzeća to je relevantan atribut

- Prosti i složeni atributi
  - Prosti atributi:
    - Visina (cm),
    - Ocena,
    - Smer itd.
  - Složeni atributi:
    - Adresa (Ulica, Broj, Mesto, ...)
    - DatumRođenja (Dan, Mesec, Godina)



- Nekada je teško razlikovati entitete od atributa
- Npr: Adresa može biti entitet i atribut
- Ako je adresa entitet:
  - Prednost – višestruka upotreba istog koda
  - Mana – Različite su potrebe za adresama (kupci, zaposleni, dobavljači itd.)
    - Može se desiti da se neprikladne forme za unos primenjuju na svim mestima – npr. nisu iste potrebe kod kupaca i zaposlenih



- Domen je užiji pojam od tipa podataka
- Ulica i Prezime su tipa Text,
  - ali su im domeni potpuno različiti
- Npr: Naučno zvanje → Text,
  - ali taj tekst može biti iz skupa {docent, vanredni profesor, redovni profesor}
- Za definiciju domena nisu dovoljna ograničenja
  - Npr: Poštanski broj je petocifren, ali nisu svi petocifreni brojevi u igri



- Domeni mogu biti kompatibilni po tipu
  - Moguće je porediti attribute definisane u tim domenima
- Npr: Relacije: Zaposleni i Prodavci
  - Iz uslova:  $Id\_Zaposleni = Id\_Prodavci$  može se dobiti nova informacija
- Ne može se vršiti kombinovanja relacija po atributima koji nemaju iste domene – dobija se besmislena informacija
  - Većina DBMS-ova to omogućava



- Veze, odnosi - *relationship*
- U realnom sistemu objekti nisu međusobno izolovani
  - objekti se nalaze u međusobnoj interakciji
- Npr. veza “studenti polažu predmet”
  - Između entiteta **Student** i **Predmet** postoji veza **Student\_Polaže\_Predmet** koja se može nazvati **Ispit**
- Entiteti između kojih postoji veza zovu se učesnici veze (*participants*)



- Internet i Web:
  - Entiteti: reči, dokumenta, datoteke, hostovi, serveri itd.
  - Veze:
    - reči **se nalaze u** dokumentima,
    - dokument **hyperveza** dokument
    - Reči **u** datoteci, datoteke **raspoložive na** hostovima, serverima





- Učestvovanje nekog entiteta u vezi može biti **potpuno i delimično**
  - Potpuno – zapisi takvog entiteta mogu da postoje samo ako postoje i u drugom (slab entitet)
  - Delimično – zapisi takvog entiteta mogu da postoje bez obzira na drugi entitet (jak entitet)
- Npr. Za IS fakulteta:
  - Zaposleni – jak (nezavisan) entitet
  - Profesor – slab (zavisni) entitet



- Npr. Za IS prodavnice:
  - Dobavljač – jak entitet
  - Artikal – slab entitet
- Zbog toga što Artikal učestvuje potpuno u ovoj vezi:
  - Nije moguće promeniti dobavljača jednog artikla, sve dok se ne izbrišu i svi artikli koje on dostavlja

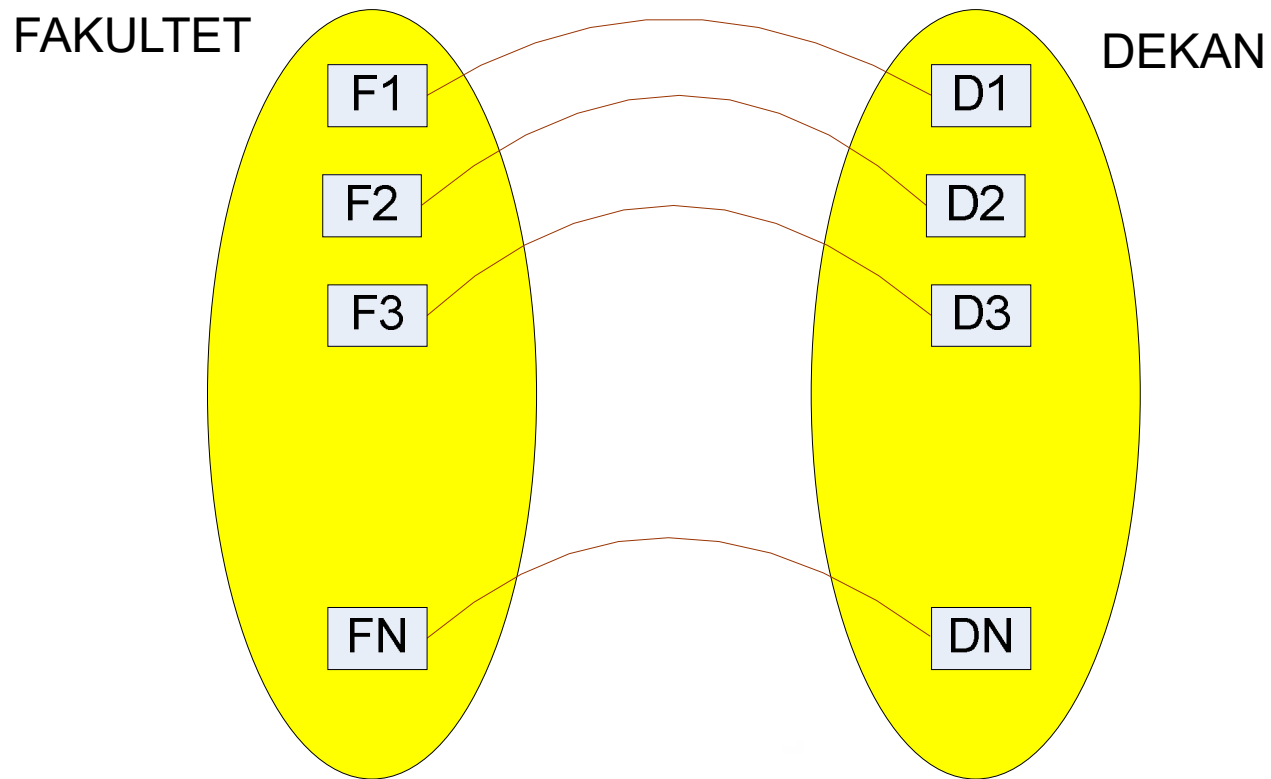


- Npr. Za IS prodavnice:
  - Kupac – jak entitet
  - Porudžbina – slab entitet
- Kupac učestvuje delimično, a porudžbina potpuno. Posledica je:
  - Podaci o kupcu se mogu uneti i pre nego što taj kupac pošalje prvu porudžbinu



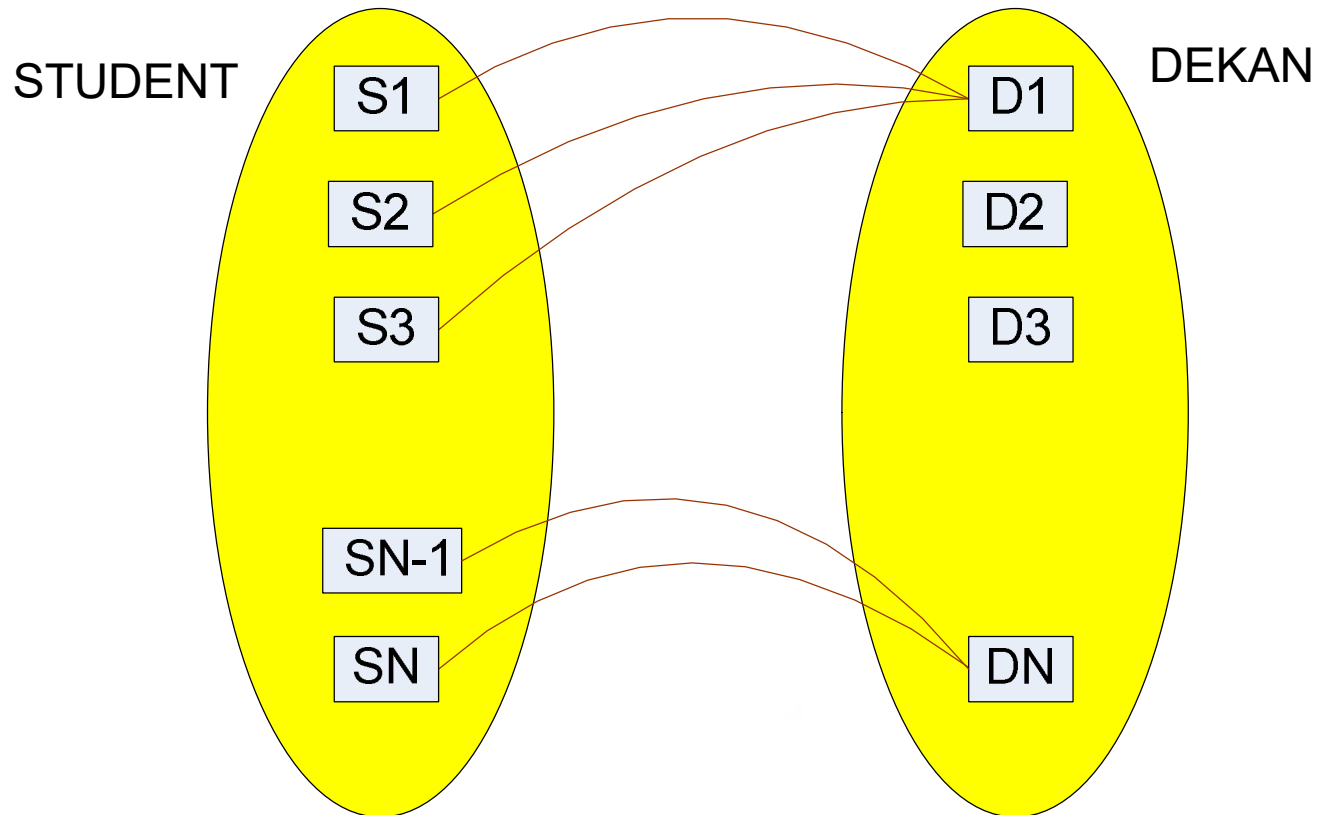
- Odnosi između objekata posmatranja prikazuju se najčešće primenom logike skupova i preslikavanja njihovih elemenata.
- Odnosi između entiteta:
  - 1:1; 1:N; N:M
  - Drugi način prikazivanja (1:1; 1:\*; \*\*)





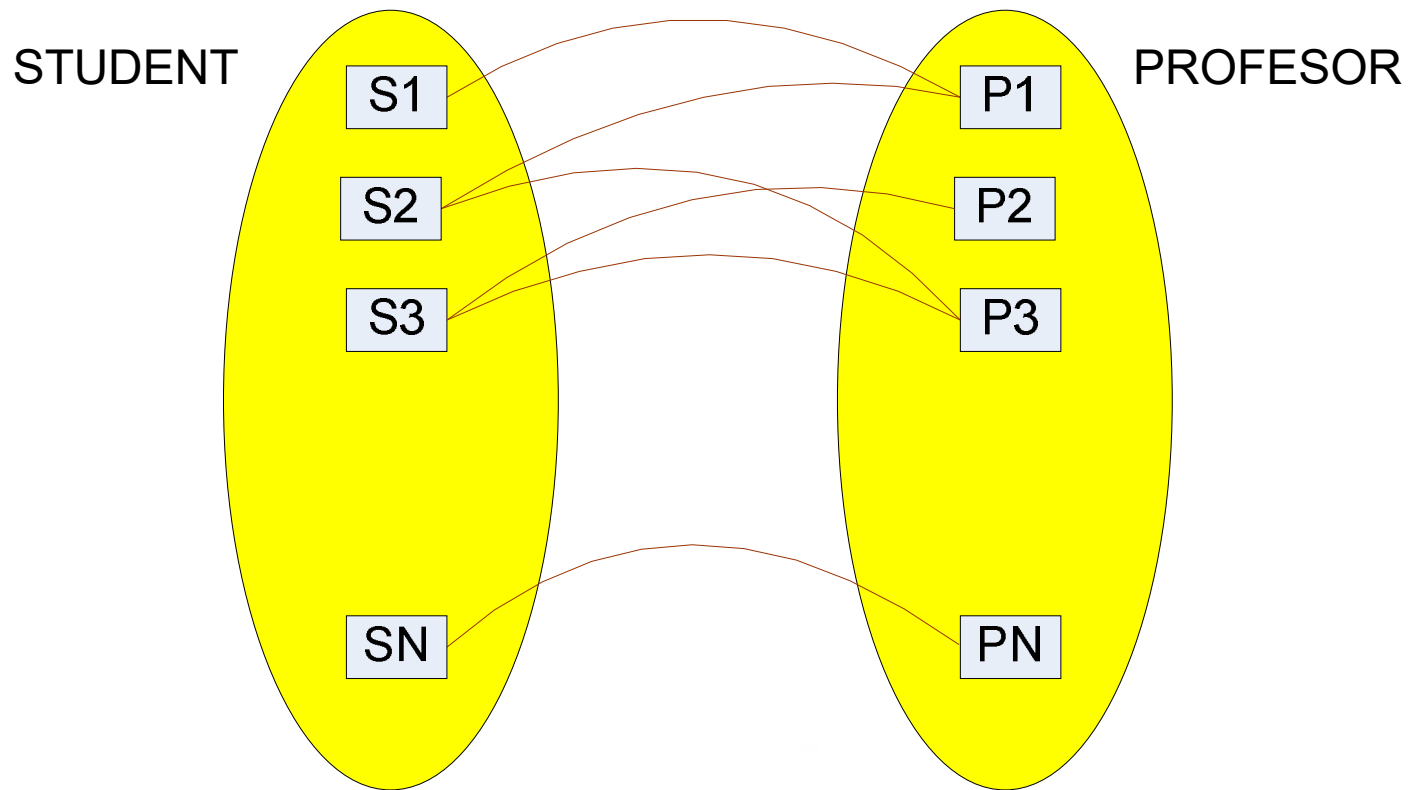
Odnos između entiteta FAKULTET i DEKAN





Odnos između entiteta STUDENT i DEKAN





Odnos između entiteta STUDENT i PROFESOR



- **Različiti modeli BP:**
  - Tabele (*flat file, spreadsheet*)
  - Hijerarhijski model
  - Mrežni model
  - Relacioni model
  - Objektni model
- **Zajednički cilj različitih modela:**
  - Olakšati smeštanje podataka i dobijanje informacija
- **Osnovne razlike kod modela:**
  - Načini uspostavljanja veza između relacija
  - Ograničenja nad podacima i vezama



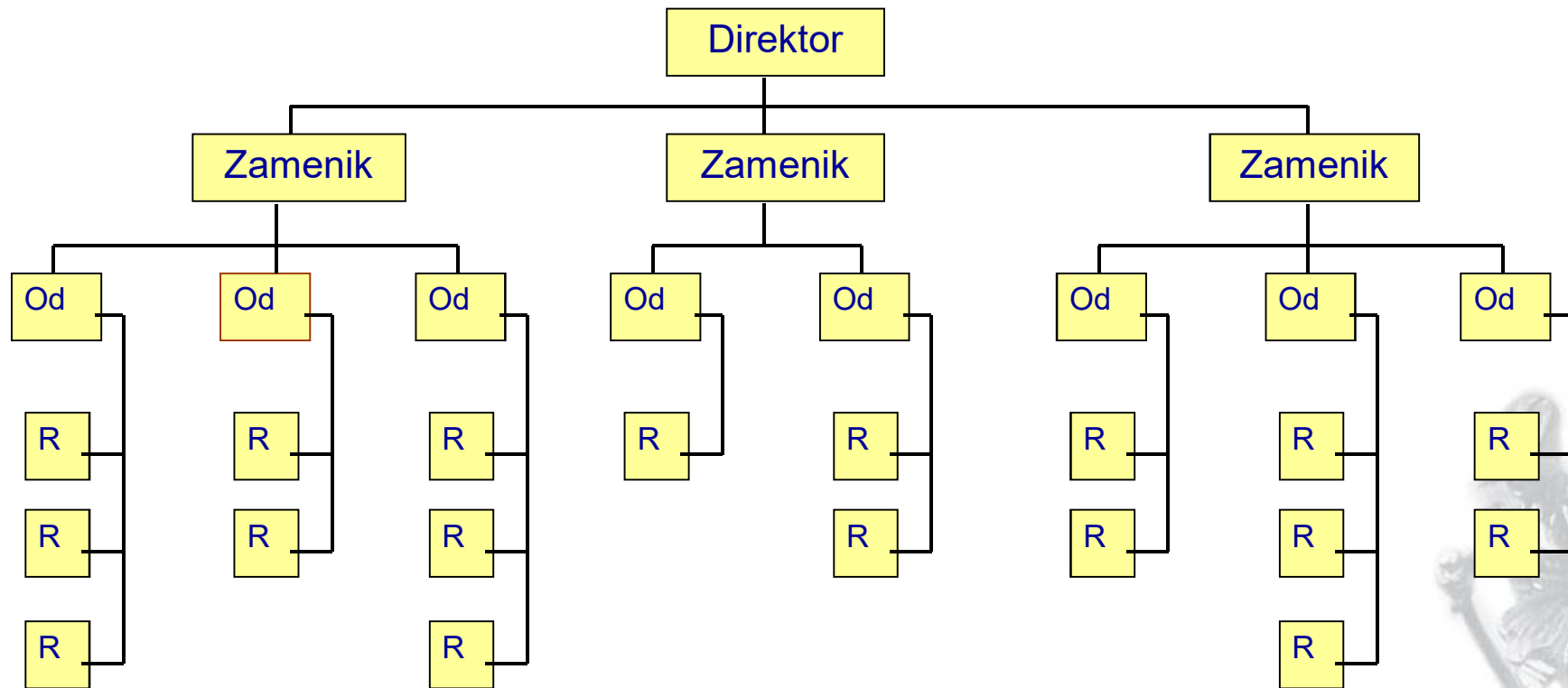


- Tabele - svi podaci u jednoj tabeli
  - Dobro:
    - Jednostavno, za mali broj rekorda i atributa
  - Loše:
    - Redudansa
    - Teškoće kod promena
    - Anomalije ažuriranja
    - Višak podataka u jednoj tabeli

Šifra	Ime	Prezime	Telefon
111-222-333	Milan	Marković	011/123-456
123-333-321	Petar	Petrović	011/543-211
123-333-321	Petar	Petrović	063/8234-567
222-333-444	Vera	Petrović	035/25-666

- Najstariji model
- Kao nasleđe i danas se koristi
- Podaci su smešteni u seriju slogova (zapisa)
- Da bi se uspostavila veza između slogova, hijerarhijski model uspostavlja relaciju:  
roditelj – naslednik (preslikavanje 1:N)
- Roditelji i naslednici su povezani vezama koje se nazivaju pokazivači
- Roditelj ima listu pokazivača za svakog od svojih naslednika





- Hijerarhijski model:
  - Nedovoljan za kompleksnije veze
  - Lako se transformiše u linearnu listu

```
Direktor (ime=Petar, tel=011/234567, itd.)
  Zamenik (ime=Marko, ...)
    Odeljenje (naziv=Plan, ...)
      Radnik (ime=Milan, ...)
      Radnik (ime=Zoran, ...)
      Radnik (ime=Pavle, ...)
    Odeljenje (naziv=Analiza, ...)
      Radnik (ime=Marija, ...)
      Radnik (ime=Jovan, ...)
    Odeljenje .....
  Zamenik .....
  .....
```

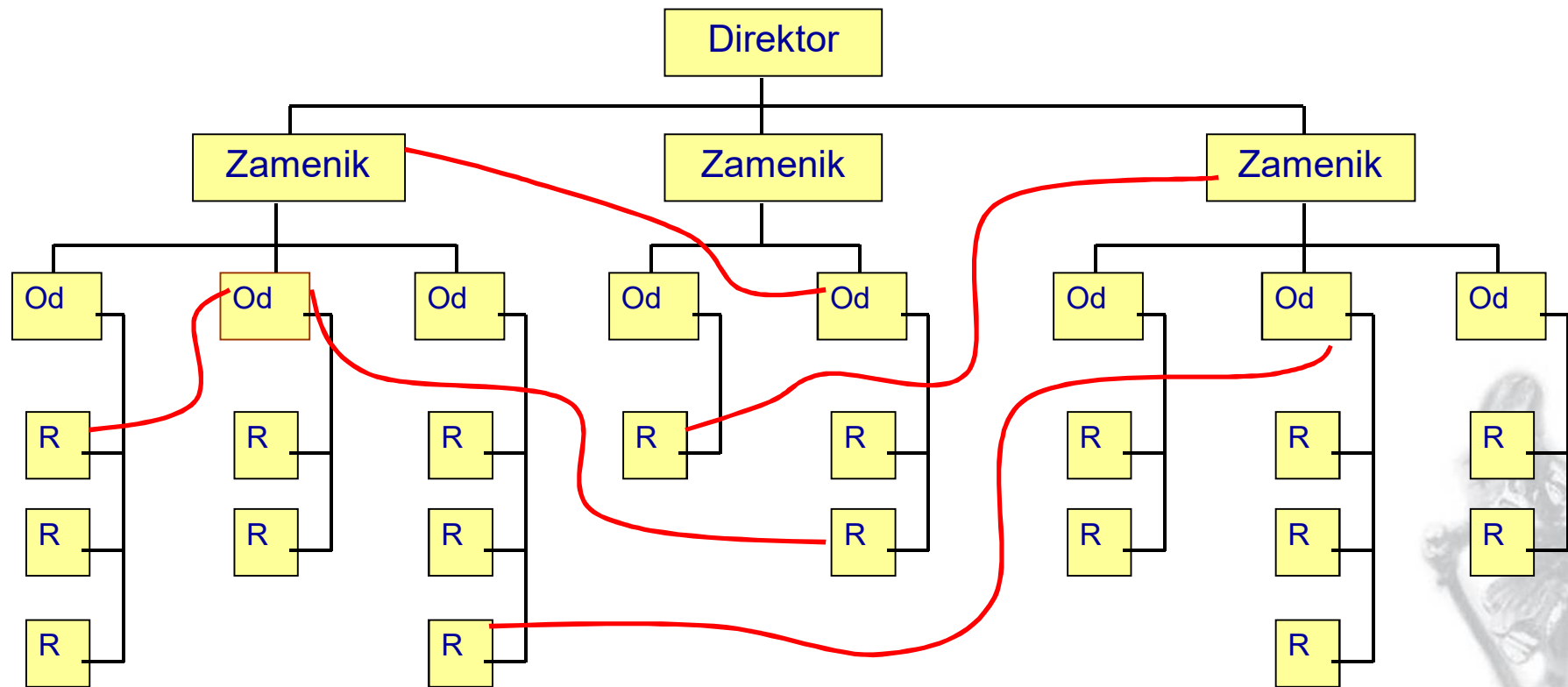


- Nedostaci:
  - Ne može se dodati slog u tabelu naslednika dok se ne uključi u roditeljsku tabelu
  - Ne podržava veze tipa N:M
  - Za kompleksnije veze moraju se pojaviti duplikati (komplikovano održavanje)



- Omogućava da se višestruki skupovi podataka koriste zajedno putem pokazivača (ili pointera)
- Neke kolone sadrže pokazivače na druge tabele umesto samih podataka.
- Tabele su povezane pokazivačima i mogu se posmatrati kao mrežna struktura.
- Omogućava veze tipa N:M
- Slabost: kompleksanost i težina upotrebe





- Osnovne karakteristike:
  - Sve se predstavlja relacijama (tabelama)
  - Zasniva se na strogoj matematičkoj teoriji
  - Minimalna redudansa podataka
  - Jednostavno ažuriranje podataka
  - Izbegnute su anomalije ažuriranja
  - Redosled kolona i redova ne utiče na informacioni sadržaj tabele
  - Ne mogu da egzistiraju dva identična reda (rekorda) u jednoj tabeli
  - Svaki red se može jednoznačno odrediti (postoji primarni ključ)





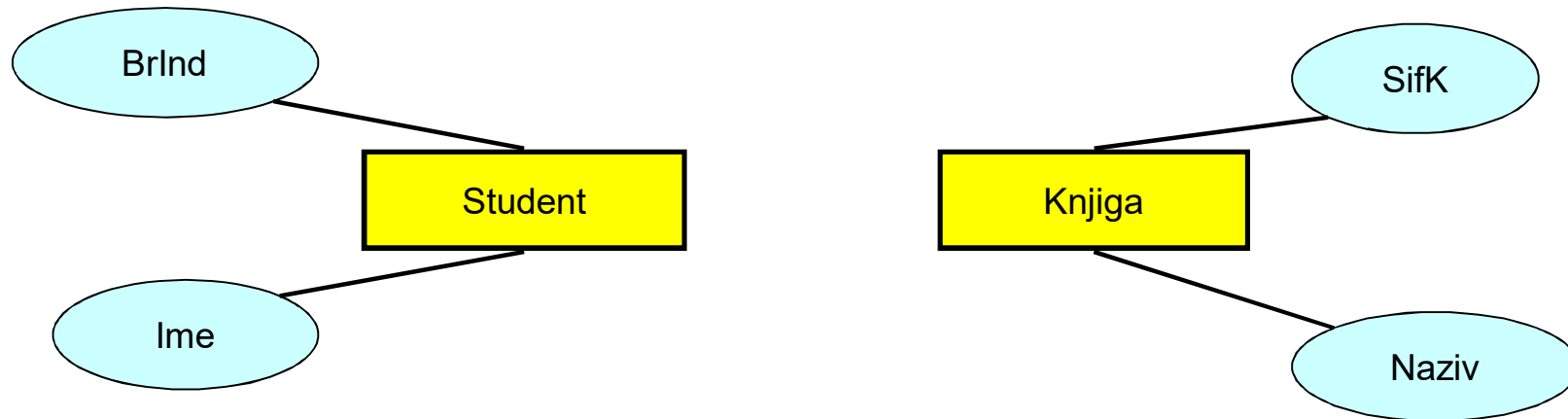
**Student**

BrInd	Ime
75/01	Marko
22/02	Petar
156/03	Milan
112/02	Dragan

**Knjiga**

SifK	Naziv
001	Računovodstvo
002	Baze podataka
003	Osnove finansija
004	Poslovna informatika
005	Marketing

Tabela, sa svojim atributima, je osnovni objekat relacione baze podataka



Grafički prikaz objekata i njihovih atributa  
(posmatra se model, a ne konkretni podaci)



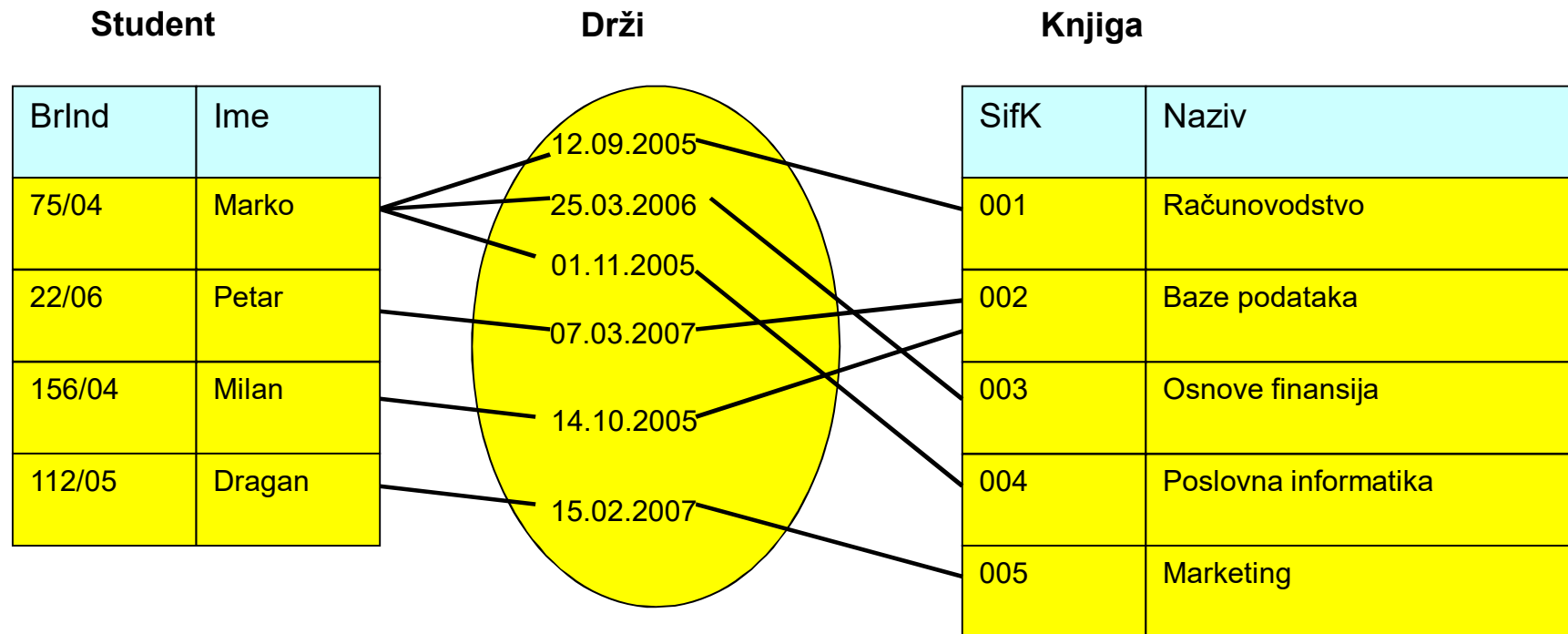
**Student**

BrInd	Ime
75/01	Marko
22/02	Petar
156/03	Milan
112/02	Dragan

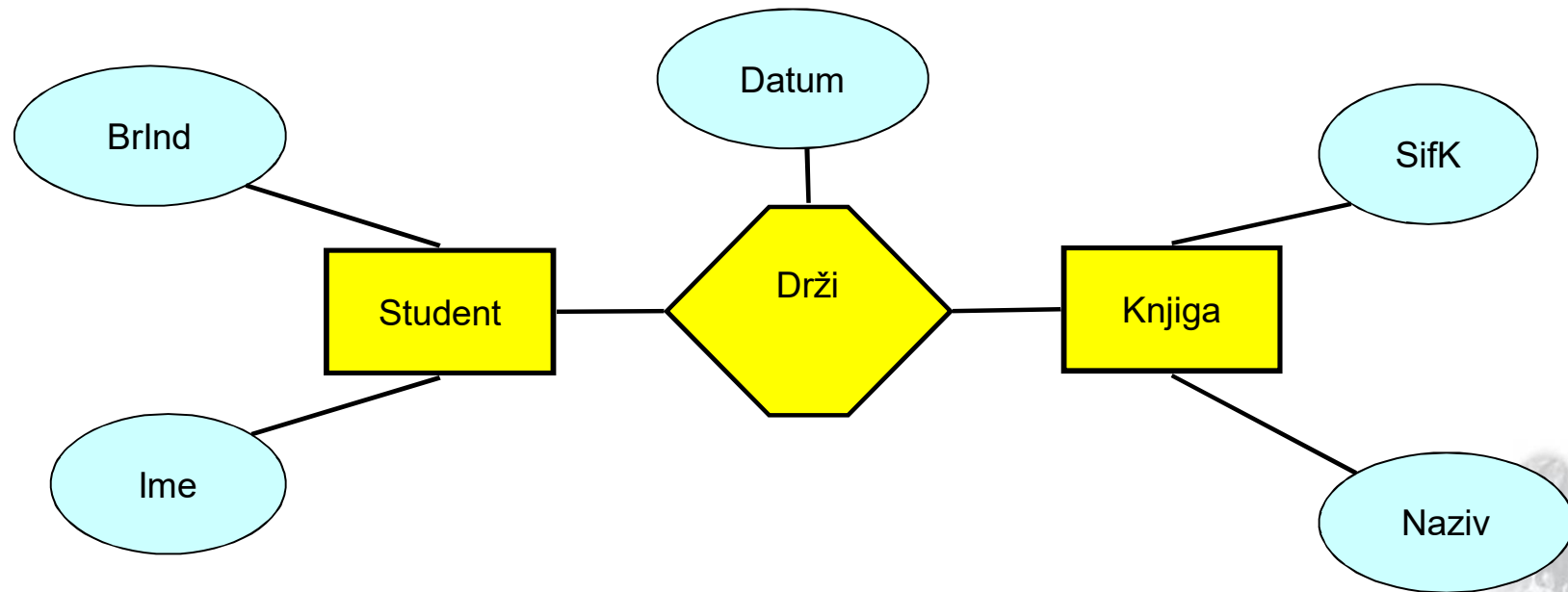
**Knjiga**

SifK	Naziv
001	Računovodstvo
002	Baze podataka
003	Osnove finansija
004	Poslovna informatika
005	Marketing

Veze između objekata realnog sveta – formira se klasa veza

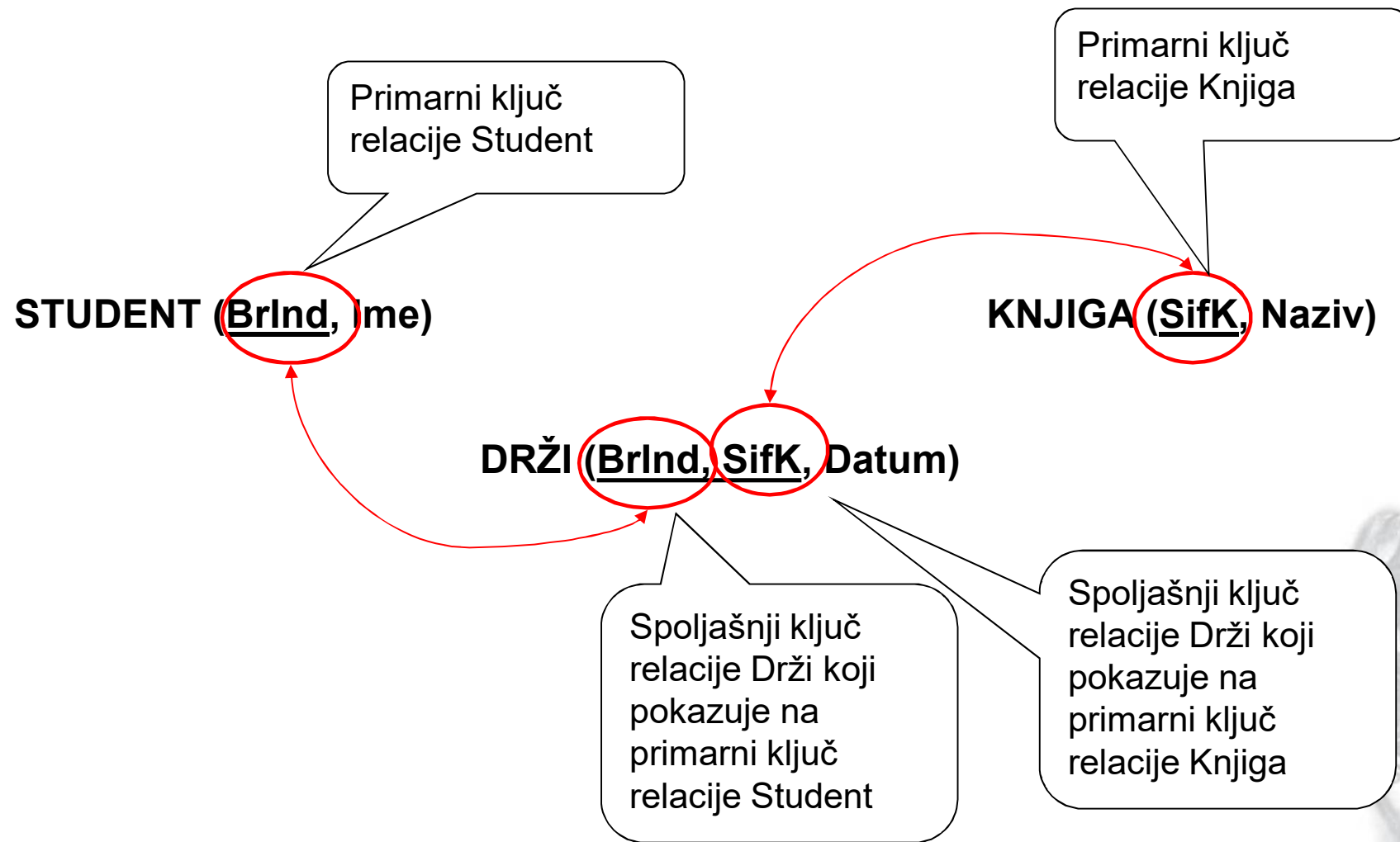


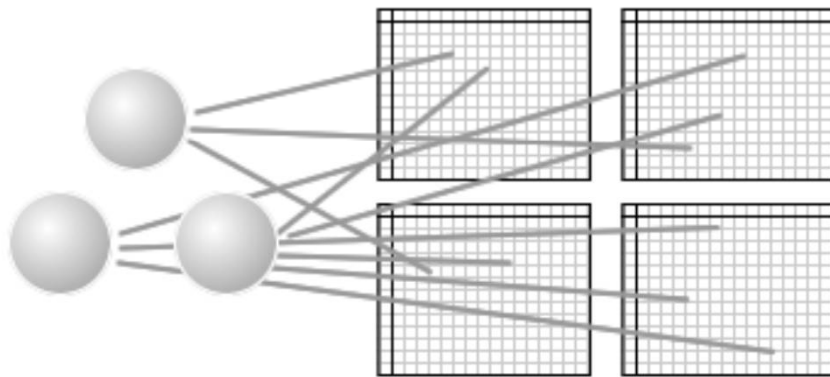
**Klasa veza može da ima svoje atribute**



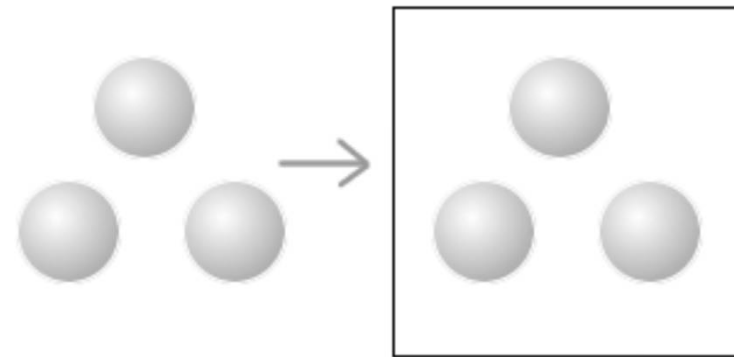
ER dijagram relacionog modela

- Suština relacionog modela je da se i klase objekata i klase veza između objekata predstavljaju na jedinstven način, tj. preko tabela.
- Nije od značaja gde i kako su smeštene tabele
- RBP se sastoji iz više tabela. Tabele su povezane ključevima
- Informacija iz RBP se dobija postavljanjem upita





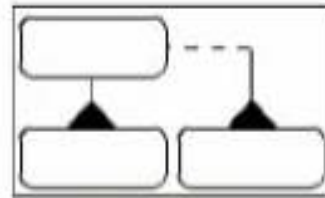
Relaciona baza



Objektna baza

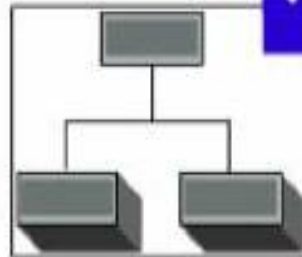


Konceptualno modelovanje

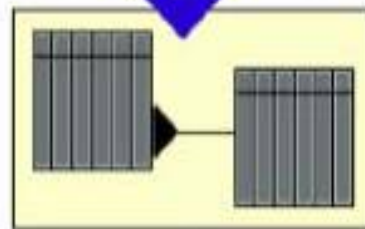
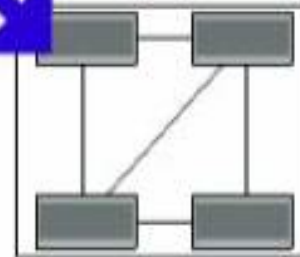


ER model

Hijerarhijska BP



Mrežna BP



Relaciona BP



# Pitanja ?

