



*Računarstvo i informatika*

*Katedra za računarstvo*

*Elektronski fakultet u Nišu*

# Baze podataka (Računske vežbe) ER dijagrami

Letnji semestar 2017/2018



# Sadržaj

- Baze podataka – osnovni koncepti
- Faze projektovanja baze podataka
- Model entiteta i veza (ER model)
- ER dijagram - notacija
- ER dijagram – Crow's foot notacija
- ER dijagram – UML notacija

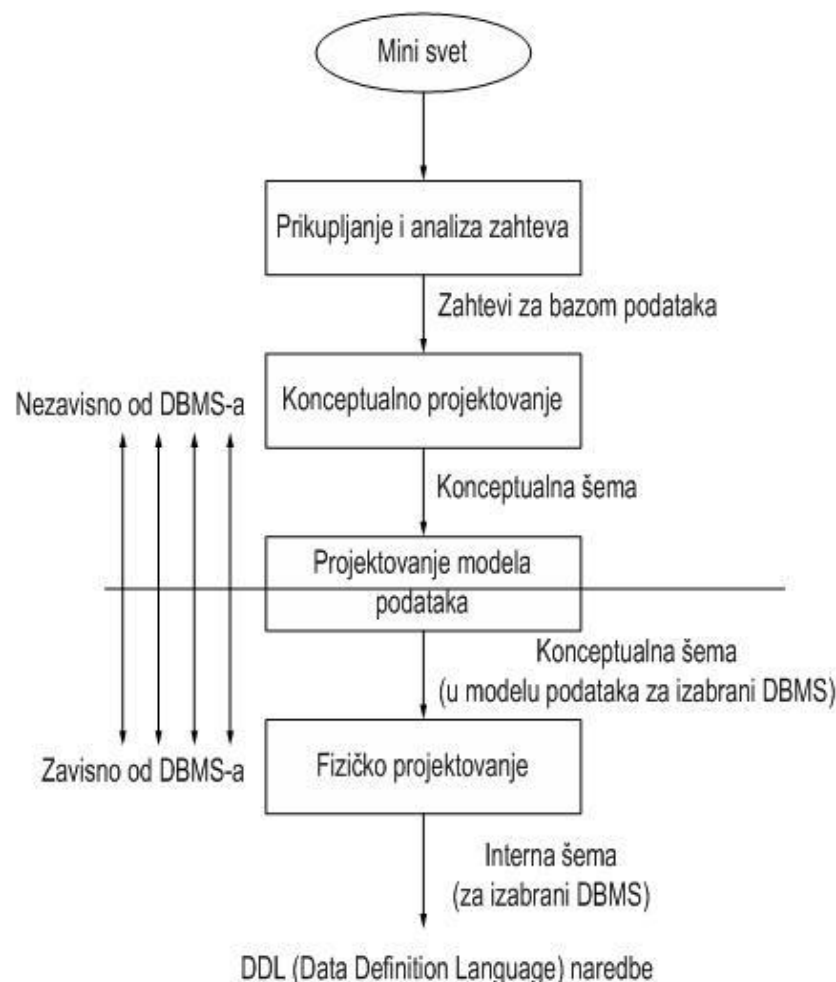


# Baze podataka – osnovni koncepti

- **Baza podataka** je kolekcija povezanih podataka.
- Model podataka je centralni deo tehnologije baza podataka.
- **Model podataka** predstavlja alat za opis strukture baze podataka odnosno apstrakciju konkretnog domena.
- Model podataka se sastoji od:
  - skupa tipova objekata
  - skupa operatora
  - skupa ograničenja
- Tipovi modela podataka:
  - konceptualni (EER, OO,...)
  - implementacioni (relacioni, hijerarhijski, ...)
  - fizički modeli



# Faze projektovanja baze podataka





# Model entiteta i veza

- **Tvorac modela je Peter Chen (1976)**
- **Osnovna ideja**
  - Realan savet (ili njegov deo) može se opisati pomoću dva primitivna koncepta: entitet i veza.
  - **Entitet** je bilo koji objekat koji se može jednoznačno identifikovati.
  - **Veza** je relacija između dva ili više entiteta.
- **Namena modela**
  - Za formiranje konceptualnog (logičkog) modela podataka.
  - Pogodno sredstvo za komunikaciju između korisnika i analitičara i projektanta softvera.



# Model entiteta i veza

- **Osnovni koncepti ER model su:**
  - entiteti i tip entiteta; jaki i slabi tip entiteta
  - atributi i vrednost atributa; ključni atribut; domen atributa
  - veze i tip veze; specijalni tipovi veza
  - ER dijagram (grafička reprezentacija ER modela)
- **Entitet je subjekat, objekat, pojam, događaj ili stanje o kojima se prikupljaju, obrađuju i prezentiraju podaci u automatizovanim informacionim sistemima a koji se može jednoznačno identifikovati.**



# ER dijagram - notacija

- Entitet se može identifikovati imenom i listom svojstava.
- U ER dijagramu tip entiteta se crta kao pravougaonik sa upisanim imenom. Ime se upisuje velikim slovima.

**RADNIK**

**UKOVODILAC**

**SEKTOR**

**PROJEKAT**

**FAKTURA**





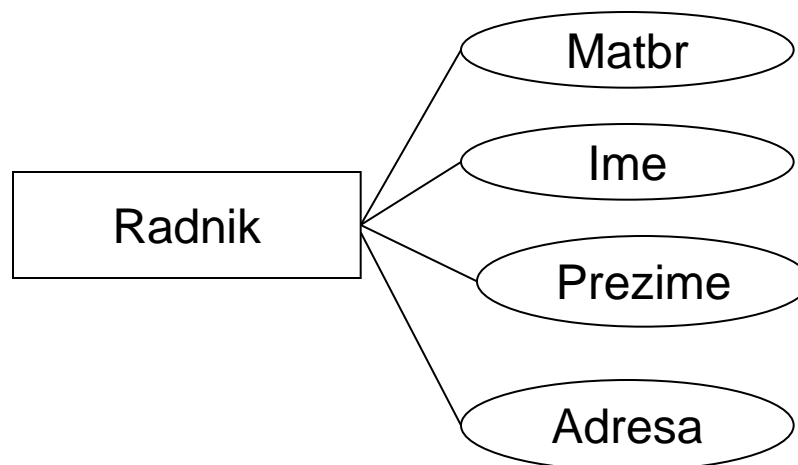
# ER dijagram – notacija

- Atributi predstavljaju zajedničke osobine koje poseduju svi entiteti jednog skupa entiteta.
- Domen atributa predstavlja skup vrednosti koje neki atribut može da ima.
- Domen atributa se definiše tipom podataka, dužinom podataka i opsegom vrednosti.



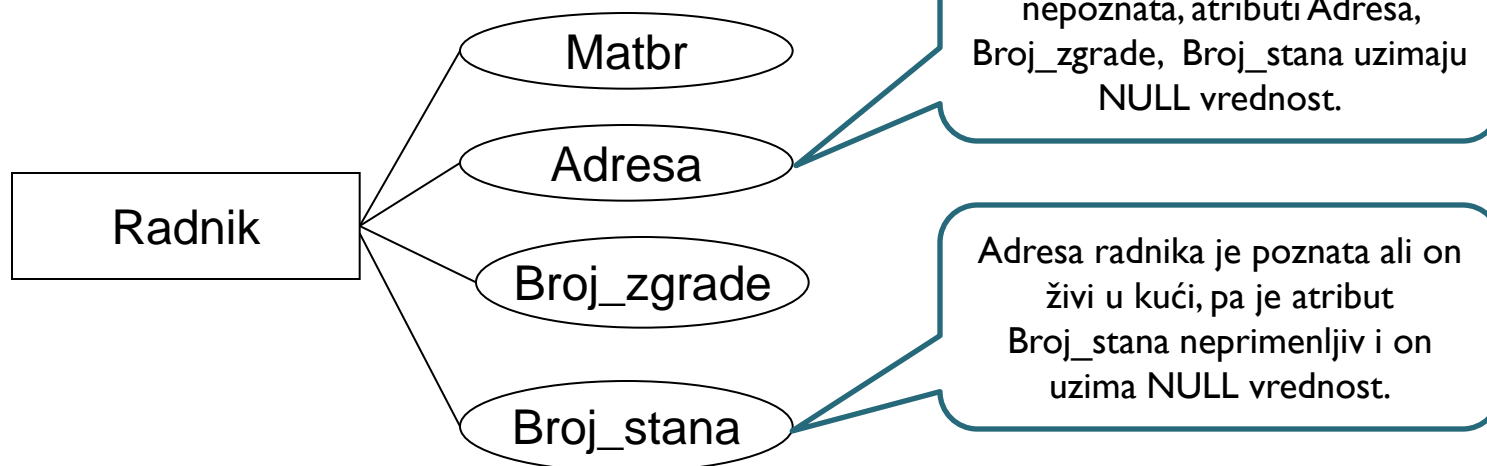
# ER dijagram – notacija

- U ER dijagramu atributi se prikazuju kao elipse sa upisanim nazivom i povezuju se neusmernim potegom sa tipom entiteta ili tipom veze na koji se odnose.



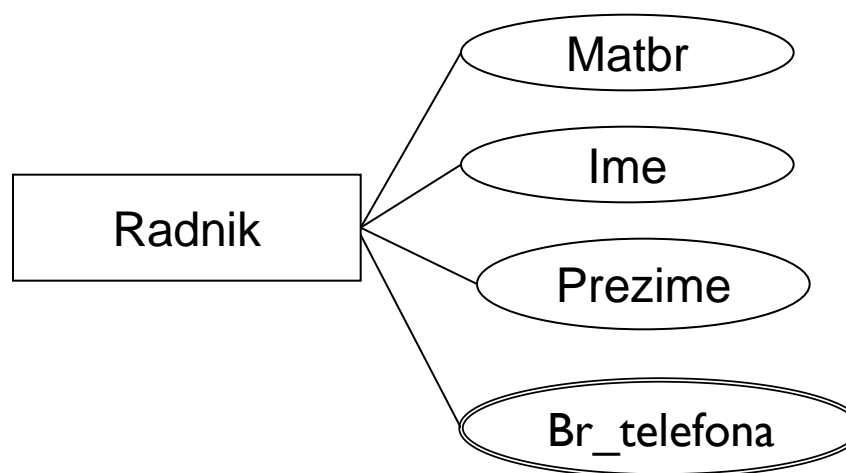
# ER dijagram - notacija

- Neki atributi mogu biti bez vrednosti za neki entitet ili mogu biti neprimenjivi za konkretne entitete. (**Primer:** Adresa).
- Tada se za taj entitet kreira **NULL vrednost** tog atributa.



# ER dijagram - notacija

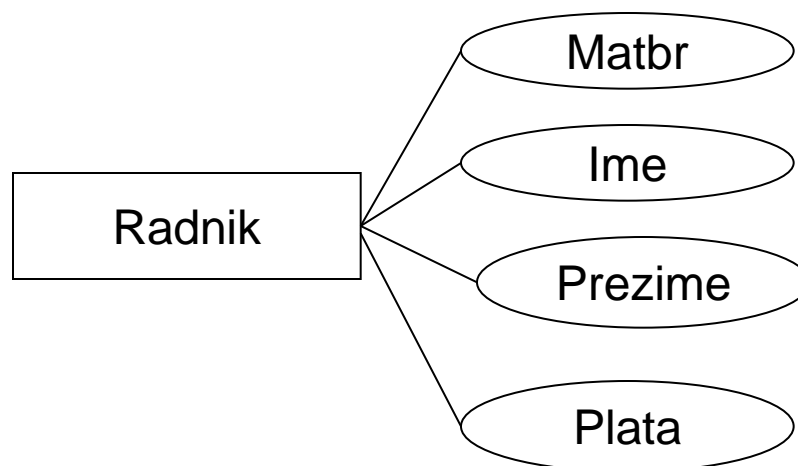
- **Jednovrednosni atribut** je atribut koji za pojavu određenog entiteta može uzeti samo jednu vrednost (**Primer:** Matbr – samo jedan matični broj, ime – osoba ima samo jedno ime)
- **Viševrednosni atribut** je atribut za pojavu određenog entiteta može uzeti više vrednosti (**Primer:** Br\_telefona – radnik može imati više brojeva).





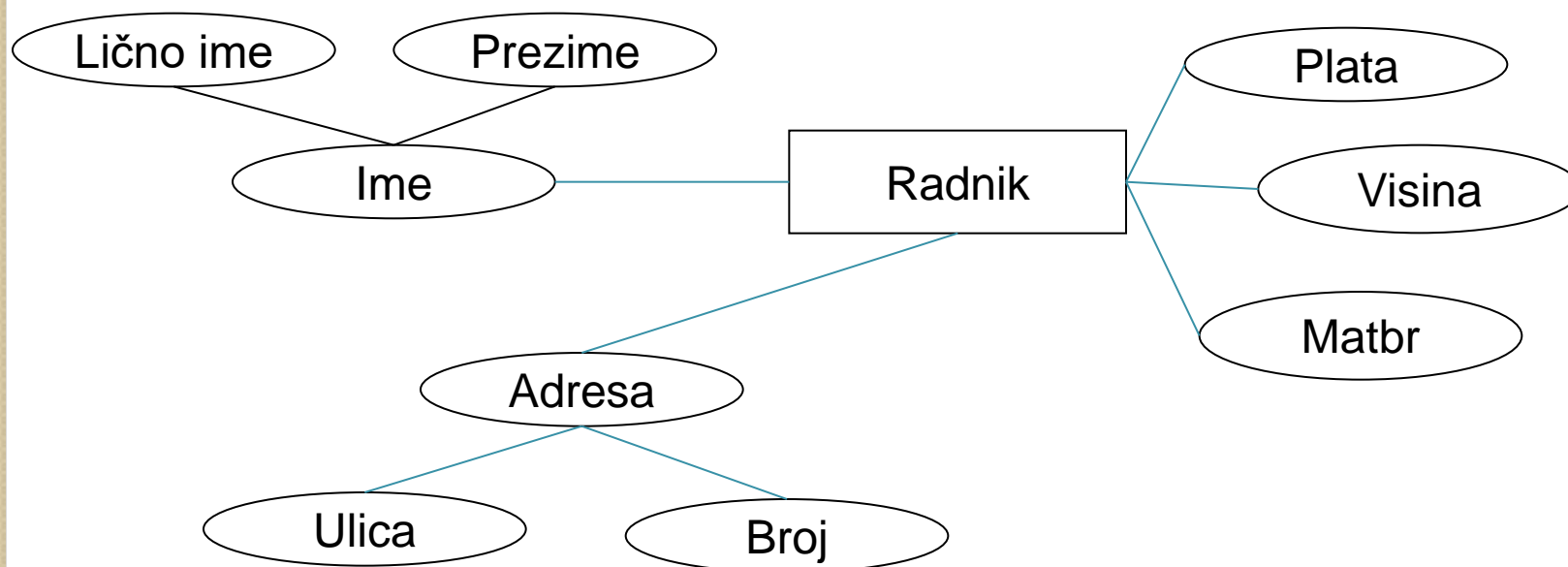
# ER dijagram - notacija

- **Prost atribut** je atribut koji se dalje ne može dekomponovati odnosno ne može doći do razdvojene primene komponenti atributa. (**Primer:** Plata, Visina, MBR.)
- Vrednost prostog atributa je prost podatak.



# ER dijagram - notacija

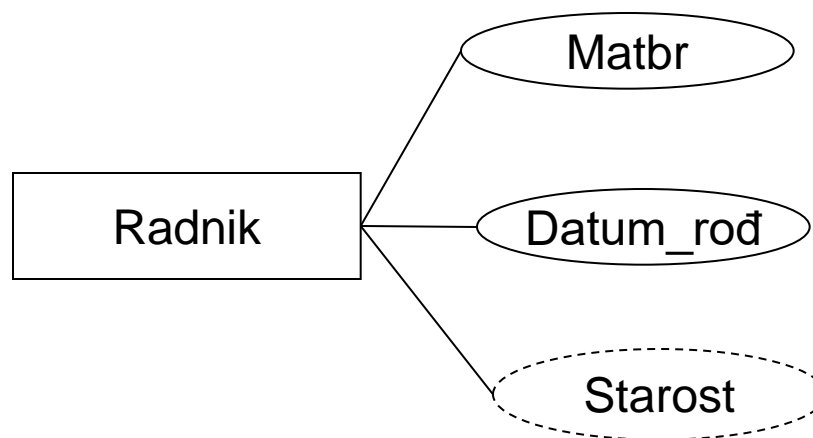
- **Složeni atribut** je atribut koji se sastoji od niza prostih atributa (**Primer:** Ime, Adresa).
- Vrednost složenog atributa je strukturni podatak.





# ER dijagram - notacija

- **Izvedeni atributi** su atributi čija se vrednost može dobiti iz vrednosti drugih atributa. (**Primer:**  $\text{Starost} = \text{Tekući\_datum} - \text{Datum\_rođ}$ )
- Vrednost izvedenog atributa je izvedeni podatak.
- Obično se ne čuvaju u bazi podataka.





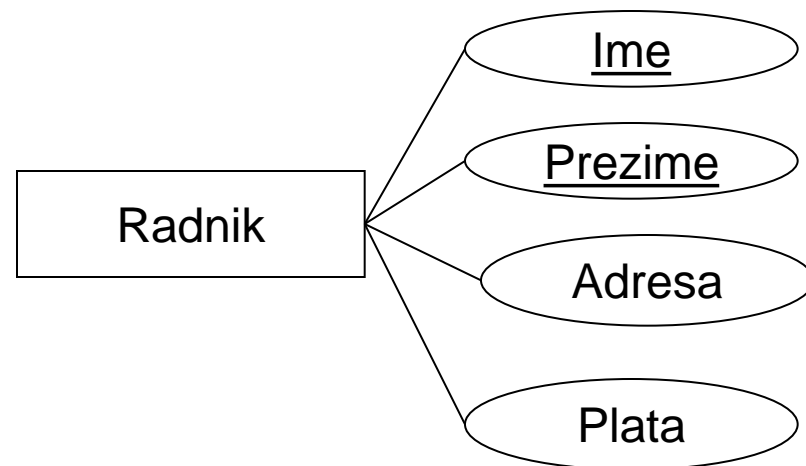
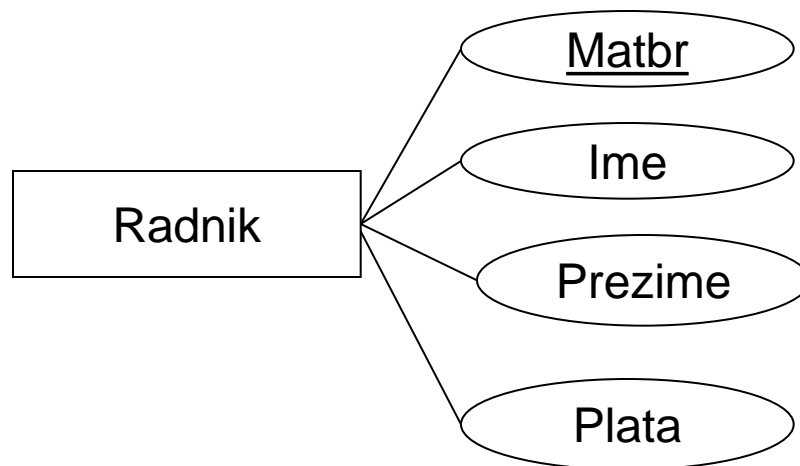
# ER dijagram - notacija

- **Ključ entiteta** predstavlja atribut ili skup atributa čije vrednosti jednoznačno identifikuju svaku pojavu entiteta (**Primer:** Matbr).
- Uklanjanjem bilo koje komponente (atributa) ključa, on gubi svoje svojstvo identifikacije.
- **Surogat ključ** - Uvodi se tamo gde ne možemo odrediti prirodan podskup atributa koji bi činili ključ (Id, Redni broj, Broj indeksa i sl.).



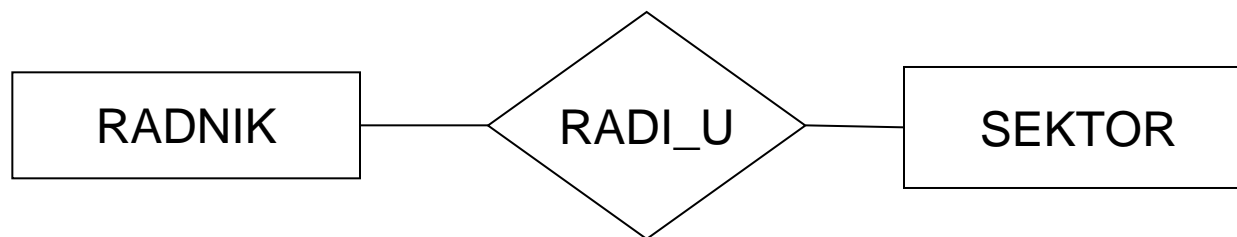
# ER dijagram - notacija

- Kod ključnih atributa naziv atributa je podvučen (**Primer:** MBR, Ime i Prezime).



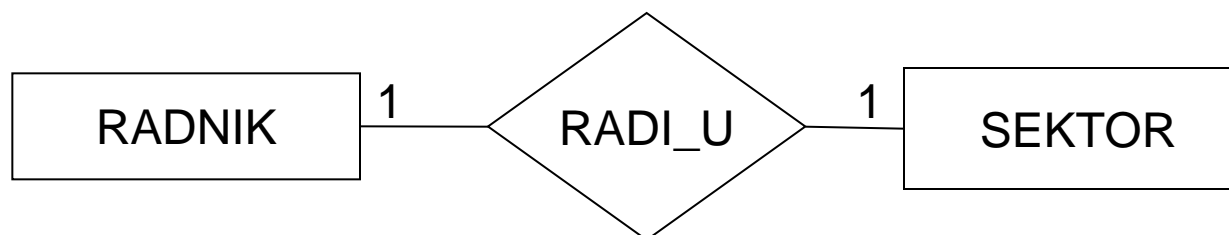
# ER dijagram notacija

- **Tip veze** modelira relacije između entiteta u istom ili različitim skupovima.
- **Veza uvek funkcioniše u oba smera** (Ako je RADNIK u vezi sa SEKTOROM važi i obratno).
- U ER dijagrami veza se predstavlja kao romb u koji se upisuje ime te veze.



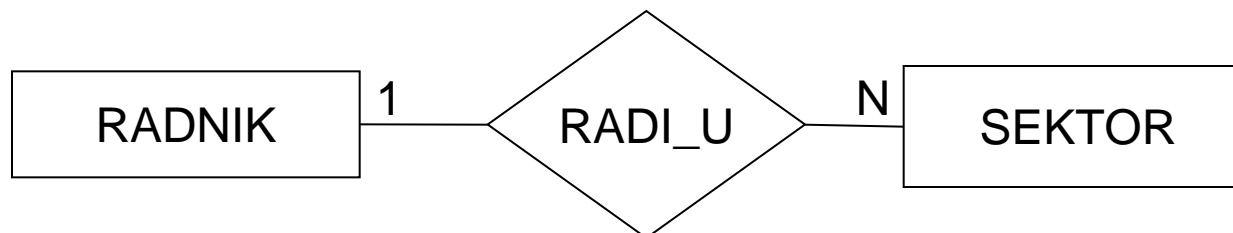
# ER dijagram - notacija

- **Kardinalnost veze** definiše broj entiteta jedne vrste koji su u vezi sa određenim brojem entiteta druge vrste.
  - Jedan-prema-jedan (1:1)
  - Jedan-prema-više (1:N) i Više-prema-jedan (N:1)
  - Više-prema-više (M:N)

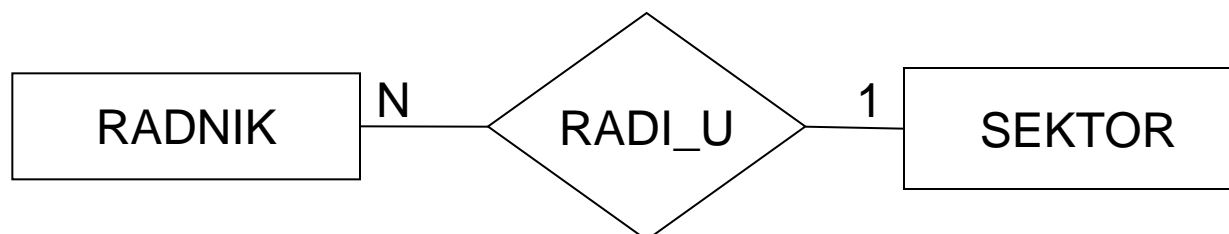


Radnik radi u **jednom** sektoru. Sektor ima **jednog** radnika.

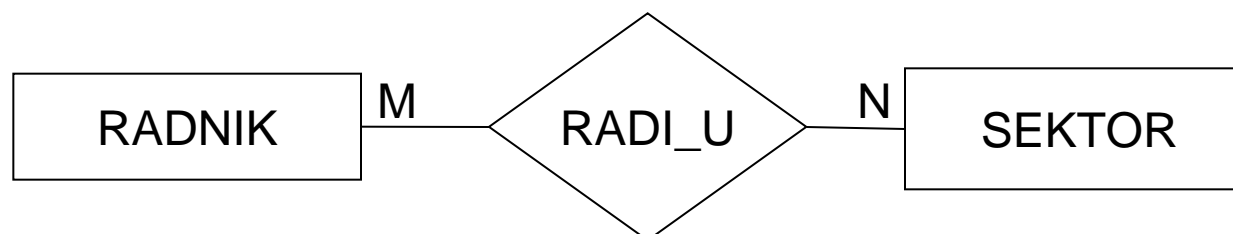
# ER dijagram - notacija



Radnik radi u **više** sektora. Sektor ima **jednog** radnika.



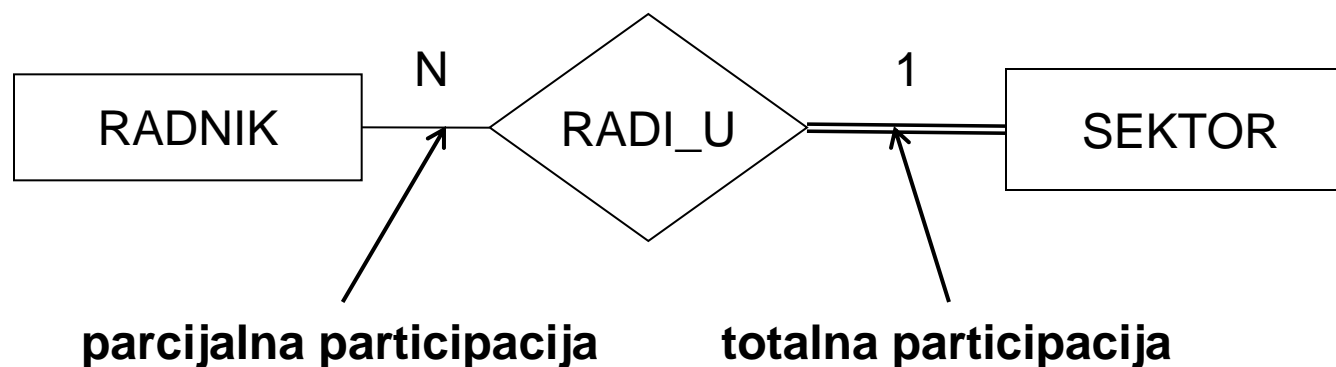
Radnik radi u **jednom** sektoru. Sektor ima **više** radnika.



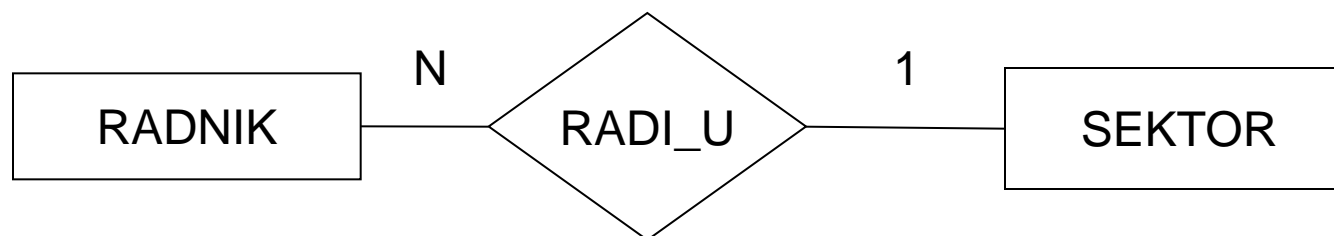
Radnik radi u **više** sektora. Sektor ima **više** radnika.

# ER dijagram - notacija

- **Participacija** entiteta u vezi definiše da li svi entiteti određenog tipa učestvuju u vezi ili ne.
  - Totalna (egzistencijalna) participacija
  - Parcijalna participacija

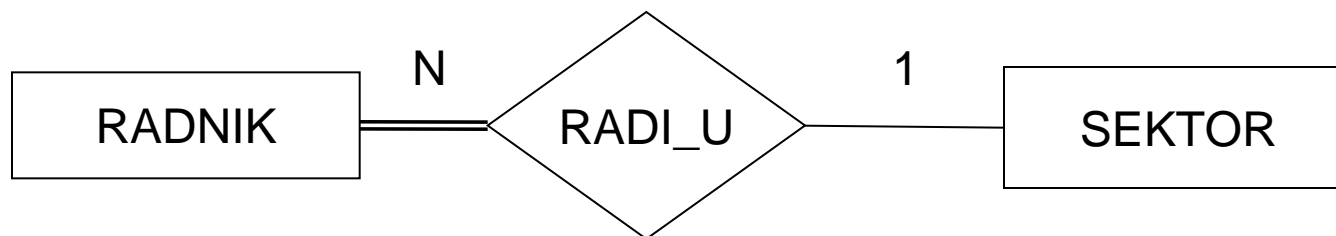


# ER dijagram - notacija



Radnik **ne mora** da radi ni u jednom sektoru a **može** da radi najviše u jednom sektoru.

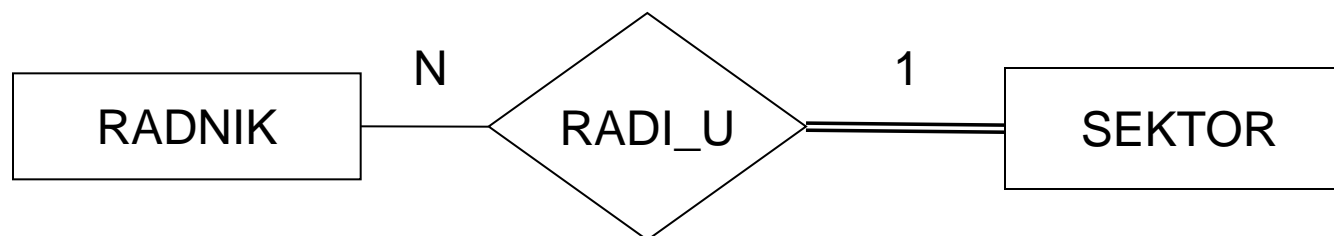
Sektor **ne mora** da ima ni jednog radnik a **može** da ima više radnika.



Radnik **mora** da radi u jednom sektoru i **može** da radi najviše u jednom sektoru.

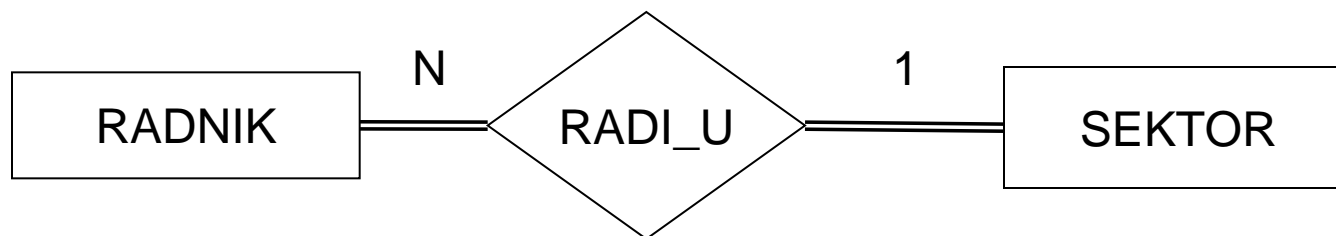
Sektor **ne mora** da ima ni jednog radnik a **može** da ima više radnika.

# ER dijagram - notacija



Radnik **ne mora** da radi ni u jednom sektoru a **može** da radi najviše u jednom sektoru.

Sektor **mora** da ima bar jednog radnik a **može** da ima više radnika.



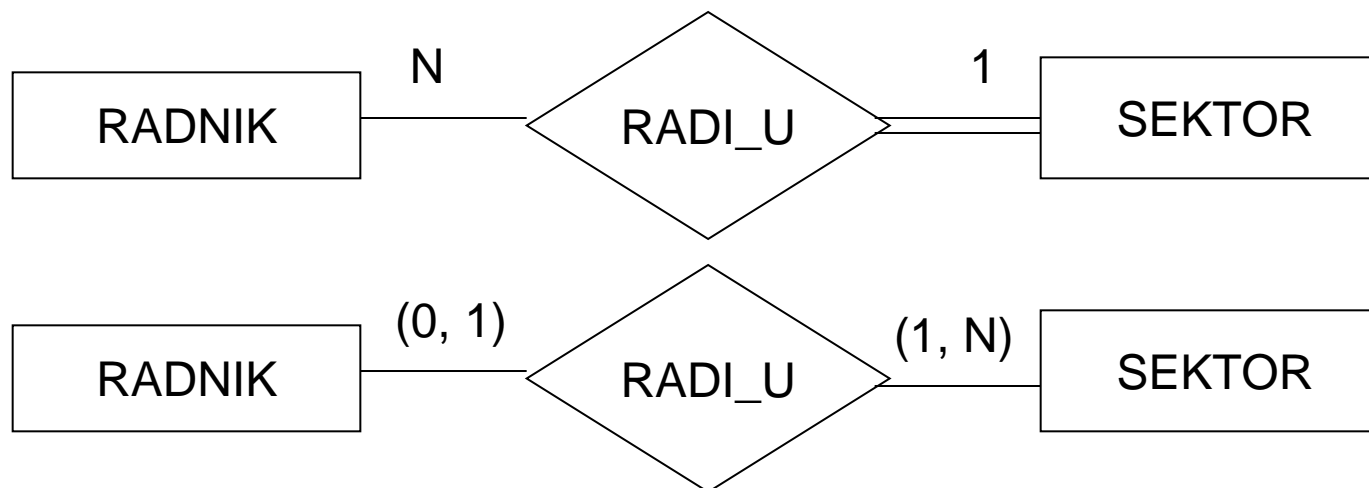
Radnik **mora** da radi u jednom sektoru i **može** da radi najviše u jednom sektoru.

Sektor **mora** da ima bar jednog radnik a **može** da ima više radnika.



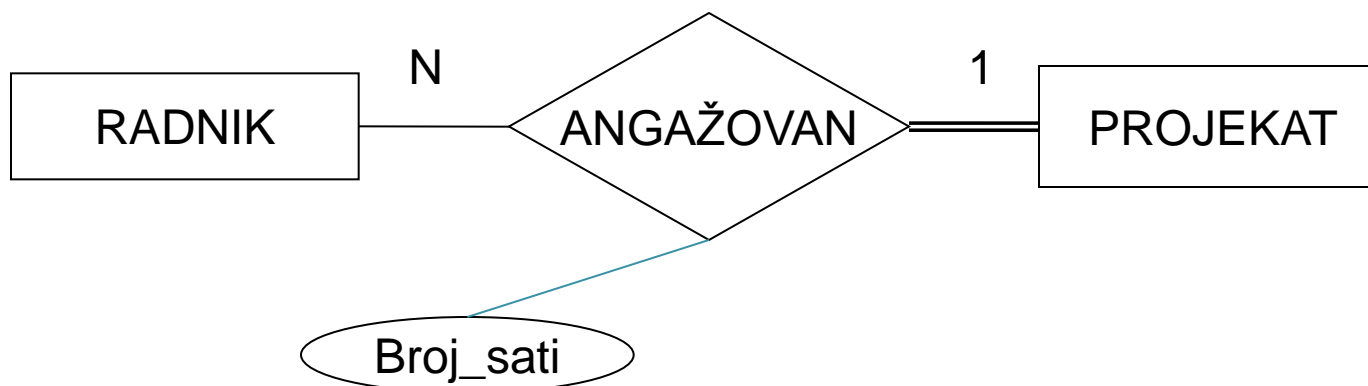
# ER dijagram - notacija

- Strukturna ograničenja – kardinalnost i participacija se predstavljaju uređenim parom (min, max) što ima sledeće značenje:
  - (min, max) – minimalan odnosno maksimalan broj entiteta određenog tipa koji učestvuje u vezi
  - min = 0 – parcijalna participacija
  - min > 0 – totalna participacija



# ER dijagram - notacija

- **Atributi veze** predstavljaju zajedničku osobinu koju imaju sve veze određenog tipa.

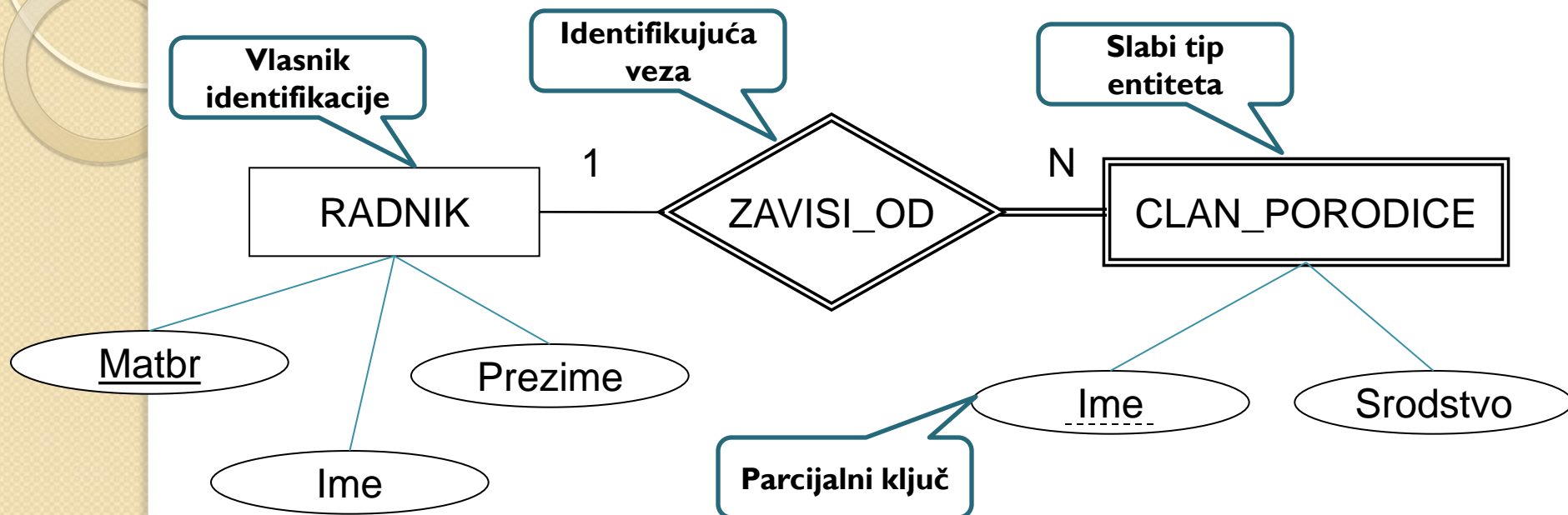




# ER dijagram - notacija

- **Slabi tip entiteta** je tip entiteta za koji je nemoguće odrediti ključ. (**Primer:** CLAN\_PORODICE)
- Mora biti povezan sa tipom entiteta koji ima sposobnost identifikacije – **vlasnik identifikacije**. (**Primer:** RADNIK)
- **Identifikujuća veza** je veza između slabog tipa entiteta i vlasnika identifikacije. (**Primer:** ZAVISI\_OD)
- **Parcijalni ključ** je skup atributa slabog tipa entiteta koji zajedno sa primarnim ključem vlasnika identifikacije omogućava jednoznačnu identifikaciju entiteta slabog tipa. (**Primer:** Ime)

# ER dijagram - notacija



**Identifikujuća veza sa strane slabog tipa entiteta mora biti totalna** (svaki instanca slabog tipa entiteta mora biti u vezi sa instancom vlasnika identifikacije)

**Kardinalost na vlasnika identifikacije mora biti 1** (slabi tip entiteta može imati samo jednog vlasnika identifikacione)

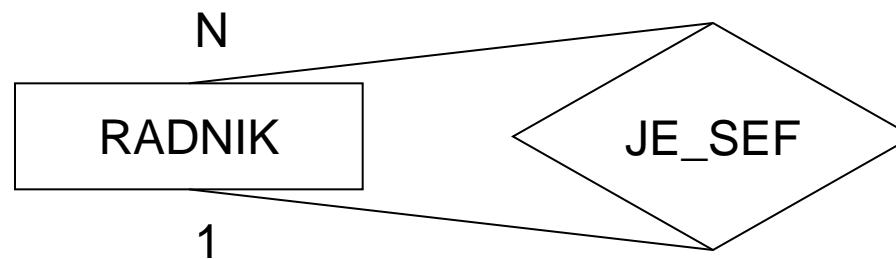


# ER dijagram - notacija

- **Stepen tipa veze** je broj entiteta koji učestvuju u vezi:
  - unarna (rekurzivna veza)
  - binarna (povezuje dva tipa entiteta)
  - ternarna (povezuje tri tipa entiteta)
  - n-arna (povezuje više od tri tipa entiteta)

# ER dijagram - notacija

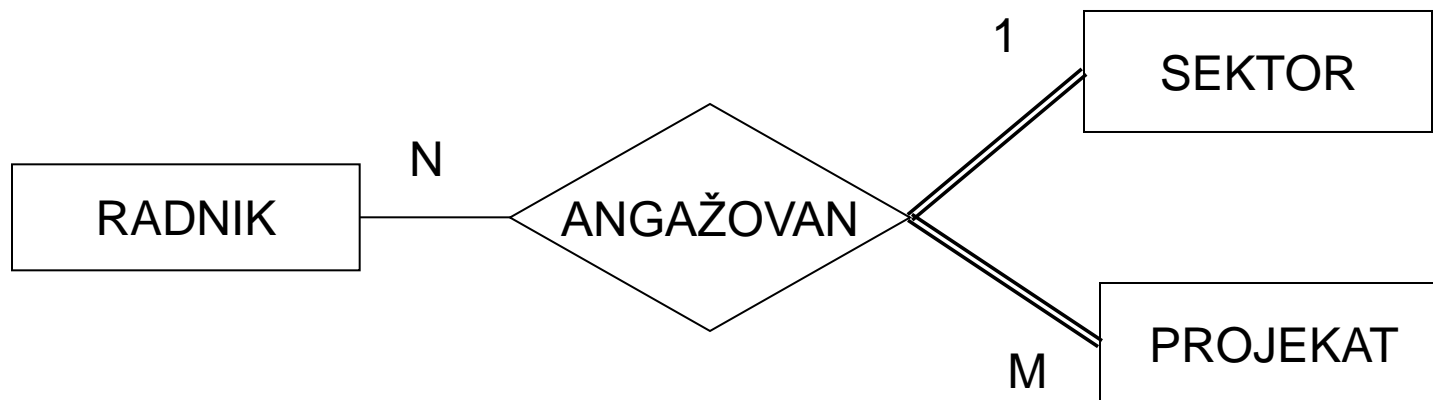
Unarna veza



Binarna veza

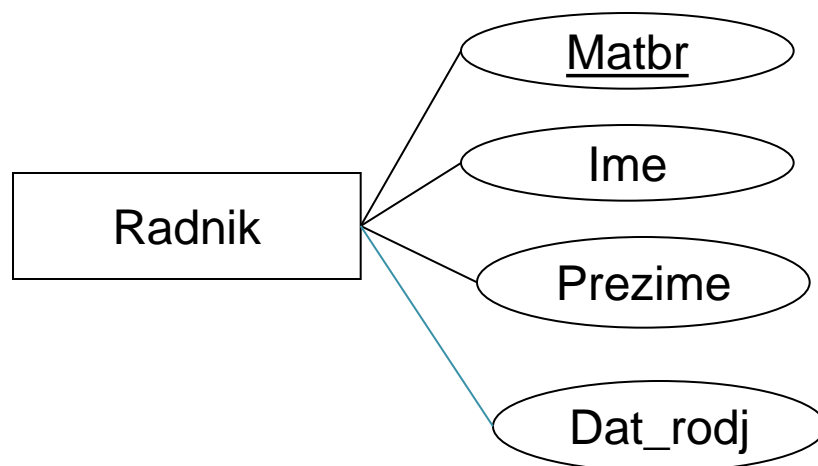


Ternarna veza



# Crow's foot notacija

- Entitet



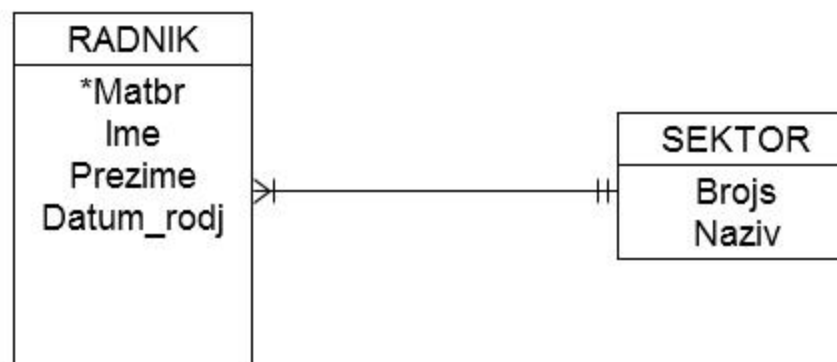
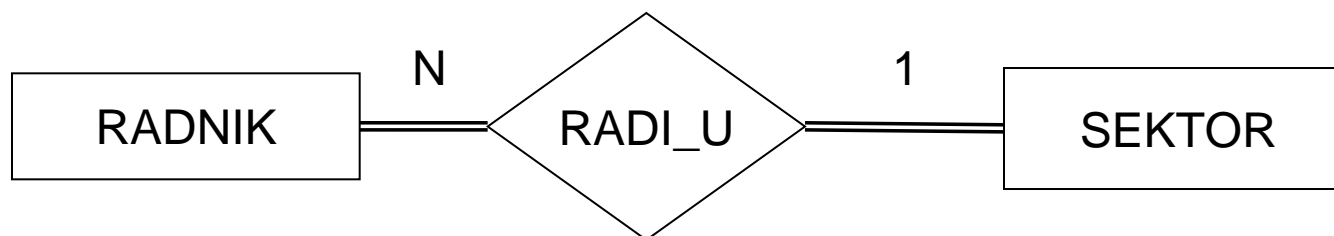
RADNIK
*Matbr
Ime
Prezime
Datum_rodj

Nema posebnih oznaka za izvedene, viševrednosne i složene attribute.



# Crow's foot notacija

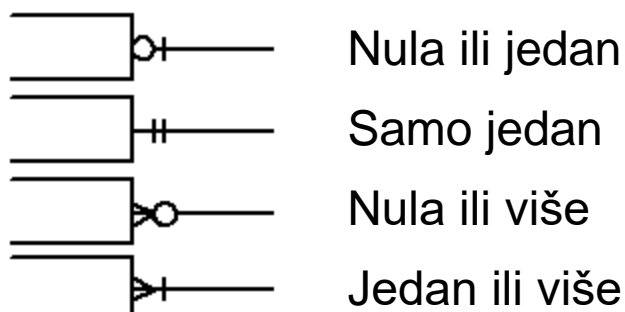
- Veza





# Crow's foot notacija

- Kardinalnost i participacija su određene (min, max) parom.



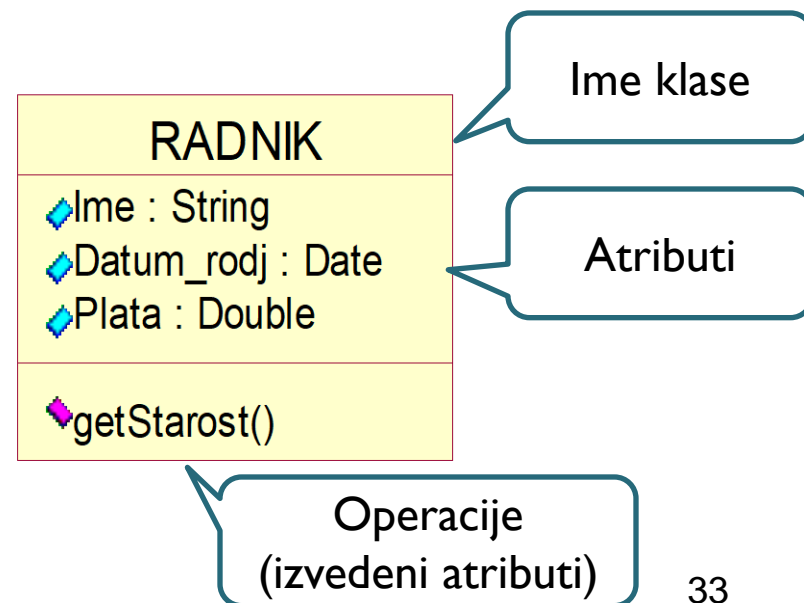
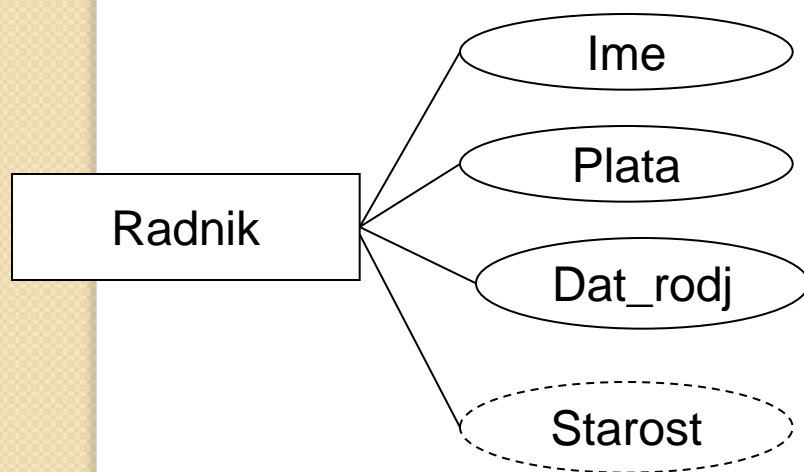


# UML notacija

- Unified Modeling Language
- Tvorci: **James Rumbaugh, Grady Booch i Ivar Jacobson** (“*Three Amigos*”)
- UML je jezik za projektovanje opšte namene koji uključuje standardizovanu grafičku notaciju koji se koristi za kreiranje abstraktnih modela sistema.
- Osnovna namena je objektno-orijentisano projektovanje softvera. Može se uspešno koristiti i za projektovanje baza podataka.
- **Dijagram klasa** definiše statičku strukturu sistema. Dijagram klasa opisuje kako je sistem definisan a ne kako se ponaša.
- Dijagram klase sadrži klase koje čine jedan sistem, njihove attribute i međusobne relacije.

# UML notacija

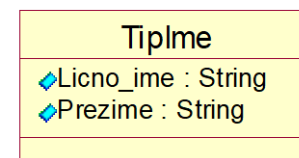
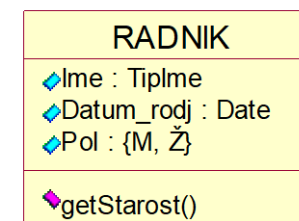
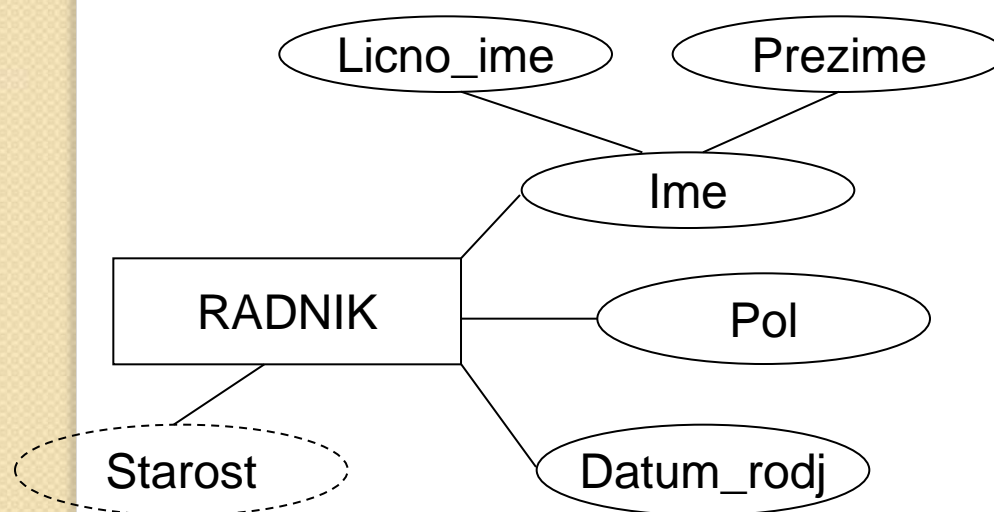
- Klasa = Tip entiteta
- Klasa je predstavljena pravougaonikom koji je podeljena na tri regiona koji sadrže:
  - Ime klase
  - Atribute
  - Operacije





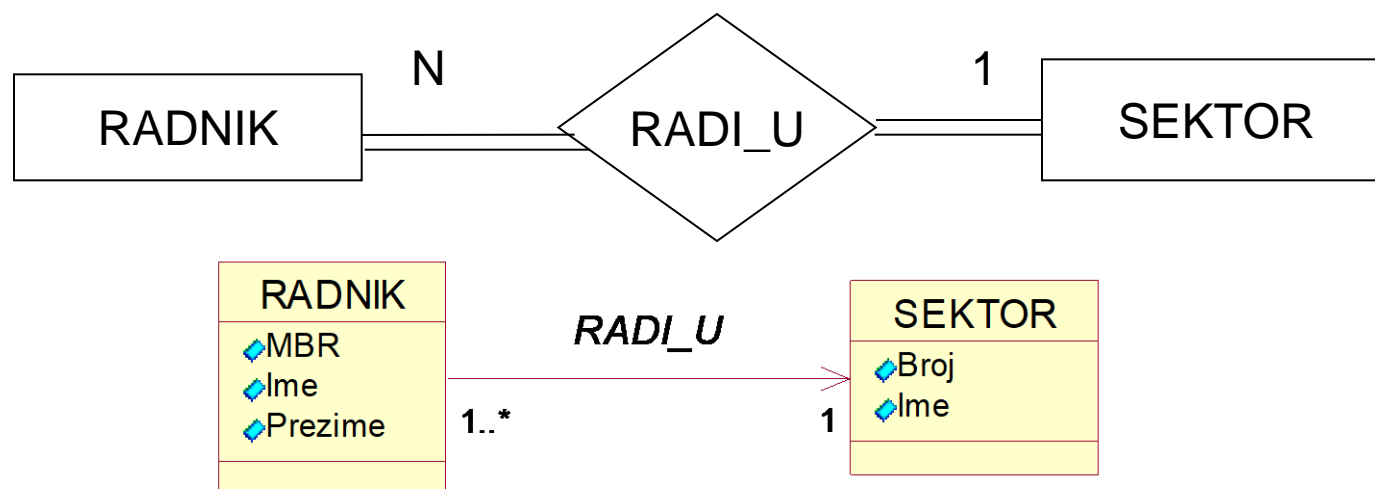
# UML notacija

- Atribut klase = Atribut entiteta
- Atributi klase:
  - Domen atributa (ime domena ili opis – **Primer:** datum i pol)
  - Strukturni domen (za složene i viševrednosne attribute, **Primer:** složeni atribut Ime)
  - Operacije (izvedeni atributi, **Primer:** getStarost)

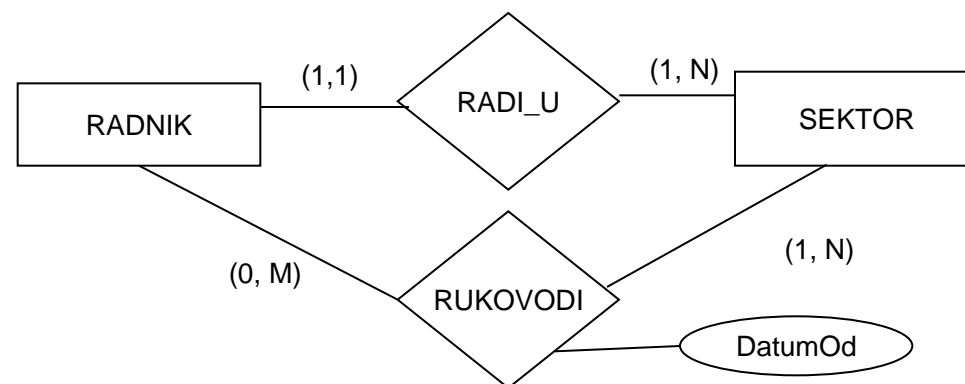
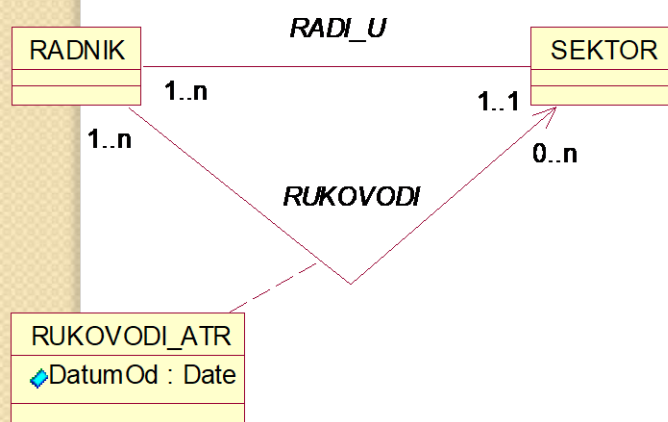
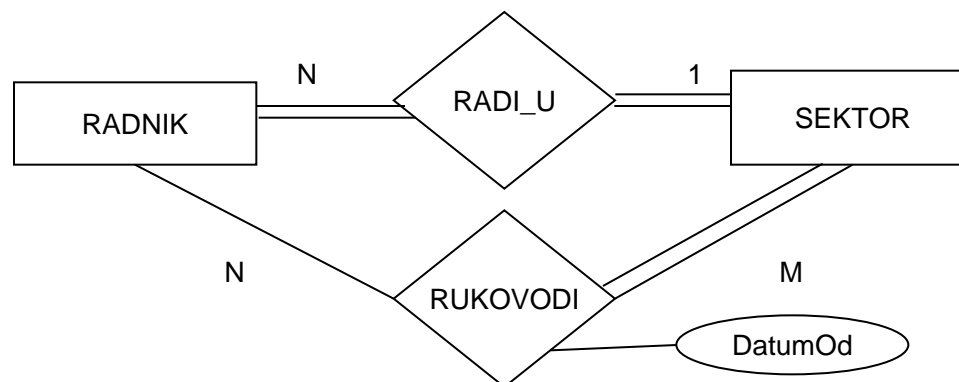


# UML notacija

- Asocijacija = Tip veze
- Asocijacija se predstavlja linijom koja povezuje dve klase. Kardinalnost i participacija se predstavljaju (min, max) parom.



# UML notacija





# UML notacija

- Agregacija nema odgovarajući ekvivalent u EER dijagramu.
- Koristi se za predstavljanje viševrednosnih atributa.

