

Semantičko modeliranje

**Projektovanje baza podataka
2009/10**

Model podataka

- ✓ Izraz **model podataka** (engl. *data model*) obuhvata:
- ✓ definicije *entiteta*,
- ✓ njihovih *atributa* (na primer, Radnik je entitet čiji su atributi Ime, MatičniBroj i Adresa),
- ✓ *ograničenja* koja važe za *attribute* (kao što je, na primer, pravilo da polje ImeRadnika ne može biti prazno).
- ✓ *opis veza* ili odnosa između pojedinih entiteta, kao i *ograničenja* koja važe za te *veze*. Na primer, rukovodilac grupe ne može imati više od pet podređenih koji mu podnose izveštaje sistema.
- ✓ Model podataka ništa ne govori o fizičkoj strukturi.

ER Model

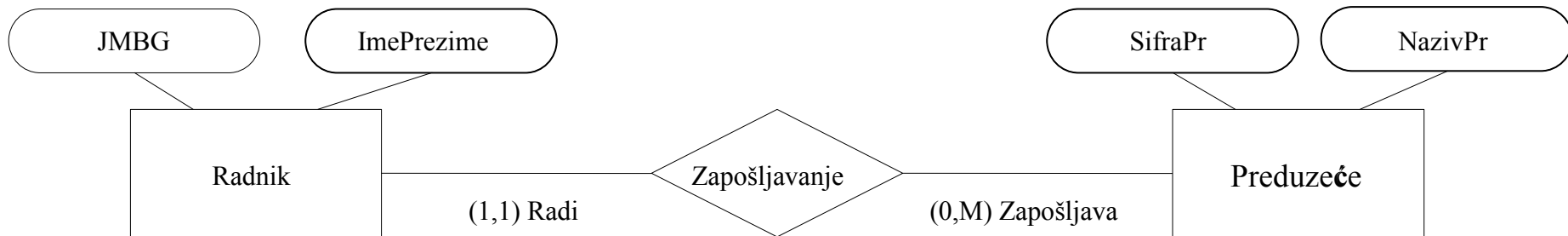
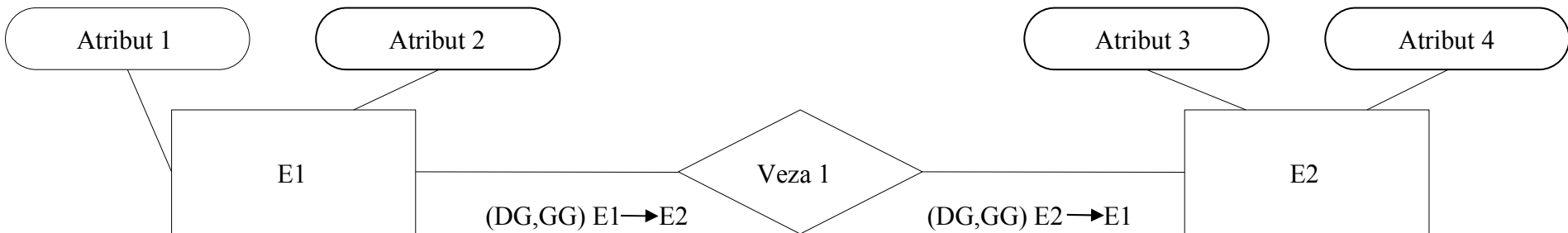
✓ Struktura modela objekti-veze (entity-relationship, MOV)

ER model definisan je ER dijagramom (DOV) kojim se definišu odgovarajući **tipovi entiteta** i uspostavljaju veze između njih, kao i definisanje detalja vezanih za **opis sadržaja entiteta (atributi i njihove karakteristike)**.

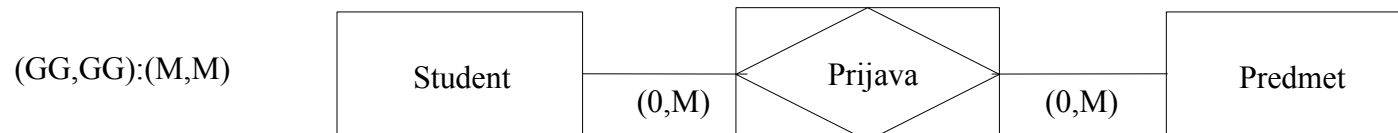
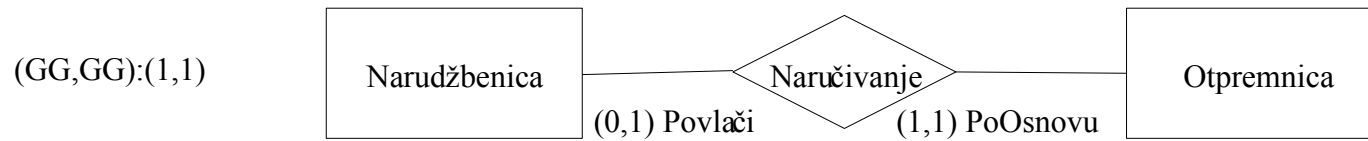
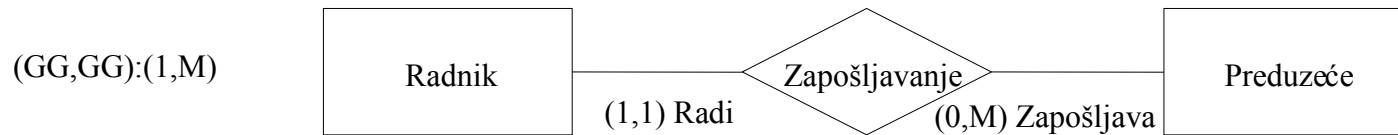
Osnovni koncepti MOV-a

- ✓ Objekat u sistemu predstavlja ili neki fizički objekat ili koncept realnog sistema.
- ✓ Veze u modelu opisuju način povezivanja dva objekta (binarna veza)
- ✓ Svaka binarna veza definiše dva preslikavanja
- ✓ Preslikavanja definišu uloge objekata u vezi.
- ✓ Kardinalnost preslikavanja ($E1 \rightarrow E2$) definiše najmanji mogući (DG) i najveći mogući (GG) broj pojavljivanja tipa objekta E2, za jedno pojavljivanje tipa objekta E1.
- ✓ $DG \in [0, 1..M]$, $GG \in [1, ..M] \rightarrow DG \leq GG$
- ✓ Objekti se opisuju atributima.
- ✓ Atribut uzima vrednost iz skupa mogućih vrednosti (domen)

Primeri



Primeri



Interpretacija podataka u modelima podataka

Sistem se najopštije može definisati kao skup entiteta i njihovih međusobnih veza.

- ✓ *Entitet* je objekat realnog sistema, koncept ili događaj u sistemu, koji se u njemu lako identifikuje. Entiteti su povezani odgovarajućim vezama.
- ✓ *Veza* je asocijacija između dva ili više entiteta.
- ✓ *Atributi* (osobine) entiteta ili veza opisuju neke njihove karakteristike.

Entiteti

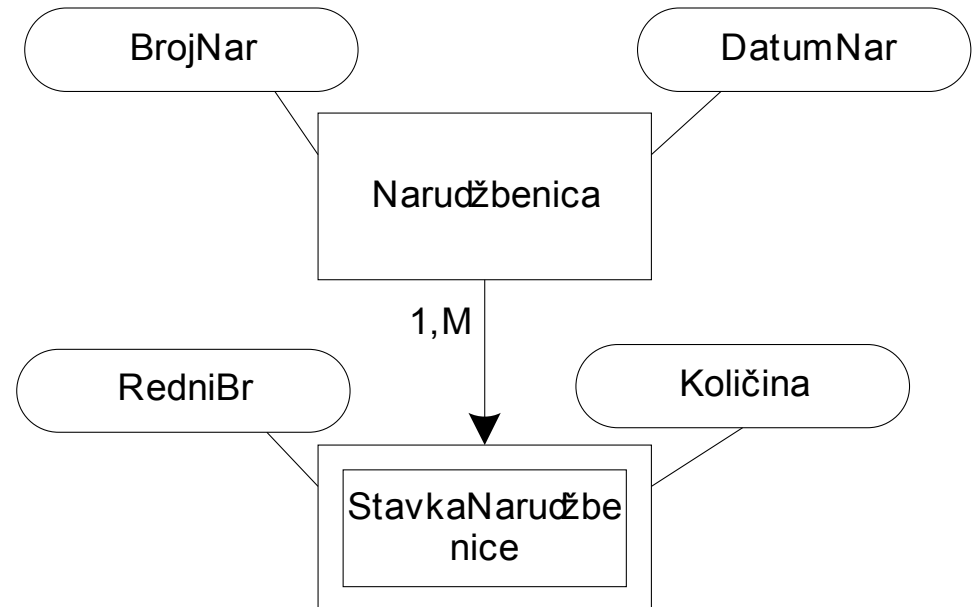
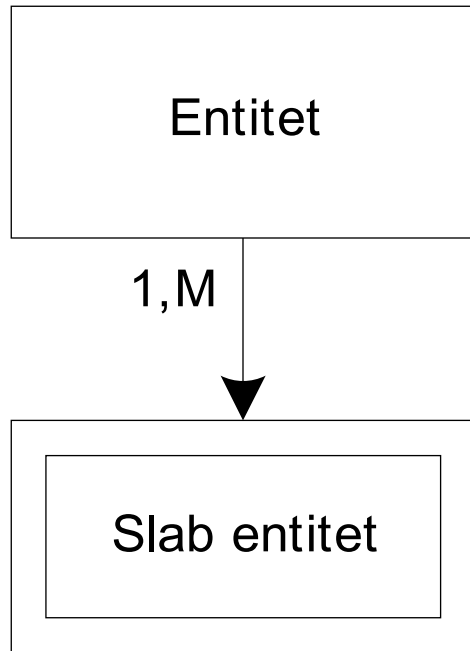
- ✓ **Nezavisni entiteti**, to su entiteti koji imaju sopstvenu identifikaciju, tj. nisu zavisni od drugog entiteta.
- ✓ **Zavisni entiteti**, to su entiteti čija je egzistencija i identifikacija zavisna od drugog ili drugih entiteta

Zavisan (slab) objekat ne može da postoji bez (egzistencijalno je zavisan od) nadređenog (jakog) objekta.

Slab objekat ne može da se identifikuje bez (identifikaciono je zavisan od) veze sa nadređenim objektom.

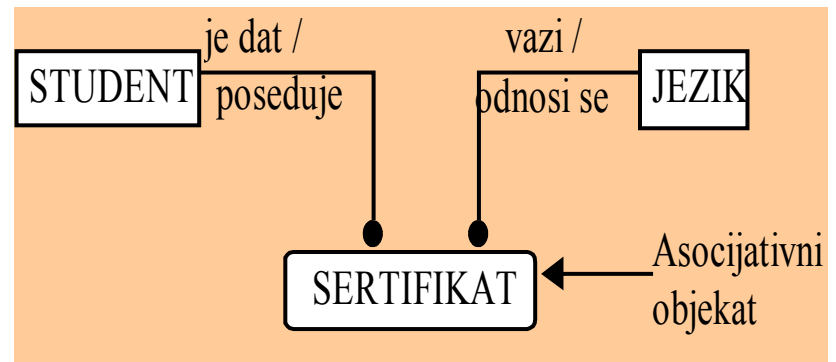
Koncept jakog i slabog objekta

- ✓ Kardinalnost preslikavanja slab→jak se podrazumeva (1,1)



Zavisni objekti

- ✓ **Karakteristični objekti**, tj. objekti koji se ponavljaju više puta za određeni nezavisni objekat
- ✓ **Asocijativni objekti**, koji predstavljaju vezu više objekata
- ✓ **Projektni objekti**, koji je sličan asocijativnom ali nema sopstvene attribute
- ✓ **Klasni objekti** koji predstavljaju podkategoriju objekata

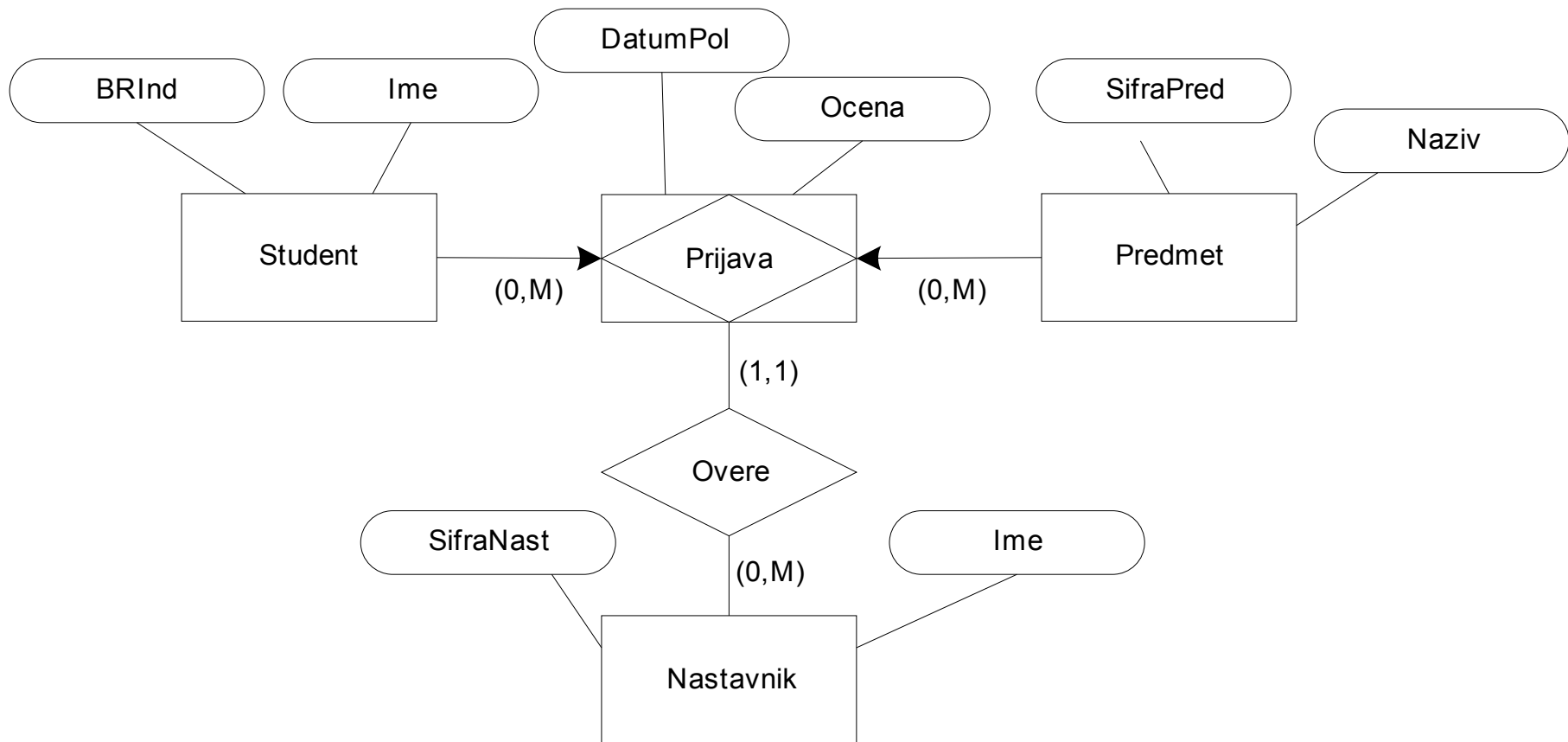


Entiteti

- ✓ Entitet je objekat, pojam, stanje i sl. o kome se prikupljaju, memorišu, obrađuju i prezentuju informacije
- ✓ Entitet po svojoj prirodi može biti različit
 - ❑ Deo okruženja (član kolektiva, aparat, zgrada, artikal, vozilo ...)
 - ❑ Apstraktni pojam (neka mera, nečije zvanje, boja, ...)
 - ❑ Događaj (udes, postupak upisa studenata,...)
 - ❑ Asocijacija (student-predmet, predmet-profesor, ..., fakultet-profesor)

Kandidati za entitete

- ✓ Osobe
- ✓ Fizički objekti (vozilo, mašina, jedinica opreme...)
- ✓ Mesta (adresa, koordinate na karti,...)
- ✓ Organizacije (preduzeća, zavod,...)
- ✓ Grupe/klase/tipovi (tip proizvoda, klasa poslova,...)
- ✓ Ugovori ...
- ✓ Primer entiteta:
STUDENT, FAKULTET, RADNIK, PROIZVOD, RAČUNAR,
ISPIT, UPIS, itd.
- ✓ Klasa entiteta: Skup sličnih entiteta:
 - Studenti određenog fakulteta,
 - Proizvodi jednog preduzeća,
 - Štedni računi građana u banci,
 - Polaganje ispita na fakultetu, ...



ER Model

✓ Objekat od koga je uspostavljena veza se zove “roditelj” (parent), a objekat prema kome je uspostavljena veza se zove “dete” (child)

✓ Identifikacija veze izvodi se u sledećim koracima:

1. Povezivanje objekata na osnovu odgovarajućih interesa.

Primer veze RADI: RADNIK -----> ODELENJE, ("radnik RADI u odeljenju")

Primer veze ZAPOŠLJAVA: ODELENJE -----> RADNIK. ("odeljenje ZAPOŠLJAVA radnike")

2. Definisanje zavisnosti objekata

3. Izborom objekta roditelj i objekta dete

Vrste veza

- ✓ **Identifikujuće veze** koje entitet dete identifikuje kroz njegovu vezu sa entitetom roditelj (lekar-pacijent).
- ✓ **Neidentifikujuće veze** koje ne identifikuju entitet dete preko identifikatora entiteta roditelja.
- ✓ **Veza prema podtipovima** uspostavlja vezu između entiteta i njegovih zavisnih, klasnih entiteta.
- ✓ **Neodređujuće veze** koje se smatraju veze više prema više.

ER koncept - analogije

TEKST	E-R KONCEPT
IMENICA	ENTITET
GLAGOL	VEZA
PRIDEV	ATRIBUT ENTITETA
PRILOG	ATRIBUT VEZE
GLAGOLSKA IMENICA	MEŠOVITI TIP ENTITETA- VEZA
REČENICA	OBJEKTI POVEZANI VEZAMA
POGLAVLJE	LOKALNI PODMODEL

Atribut i domen

Objekti se u sistemu opisuju preko svojih svojstava ili **atributa**.

Primer:

Atributi entiteta STUDENT:

BrInd, Ime, Prezime, Fakultet, Smer, Adresa

- ✓ Svaki atribut u u jednom trenutku ima neku vrednost.
- ✓ Atributi uzimaju vrednost iz skupa mogućih vrednosti. Ovaj skup se naziva **domen**.

Atribut

- ✓ Atribut je zajednička osobina koju poseduju svi entiteti jedne klase
- ✓ Broj atributa nije fiksna
- ✓ Relevantne attribute definiše kompetentna osoba
 - ☐ Od toga zavisi upotrebljivost dobijenih informacija
- ✓ Atributi svih entiteta poprimaju određene vrednosti

Atribut

- ✓ Primer: Vrednosti atributa entiteta STUDENT (jedan entitet iz klase studenata):
 - ❑ BrInd: 123/06
 - ❑ Ime: Pera
 - ❑ Prezime: Perić
 - ❑ Studijski program: master I
 - ❑ Adresa: Studentski Trg 16, 11000 Beograd

Atribut

- ✓ Premalo atributa:
 - ❑ model jednostavan za predstavljanje i analizu,
 - ❑ verodostojnost mala,
 - ❑ 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)

Atribut

✓ Primer 1:

- Sa aspekta zarada, atribut “broj cipela” nije relevantan
- Sa aspekta nabavke radničke opreme, 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

Atribut

✓ Prosti i složeni atributi

– Prosti atributi: Visina (cm), Ocena, Smer itd.

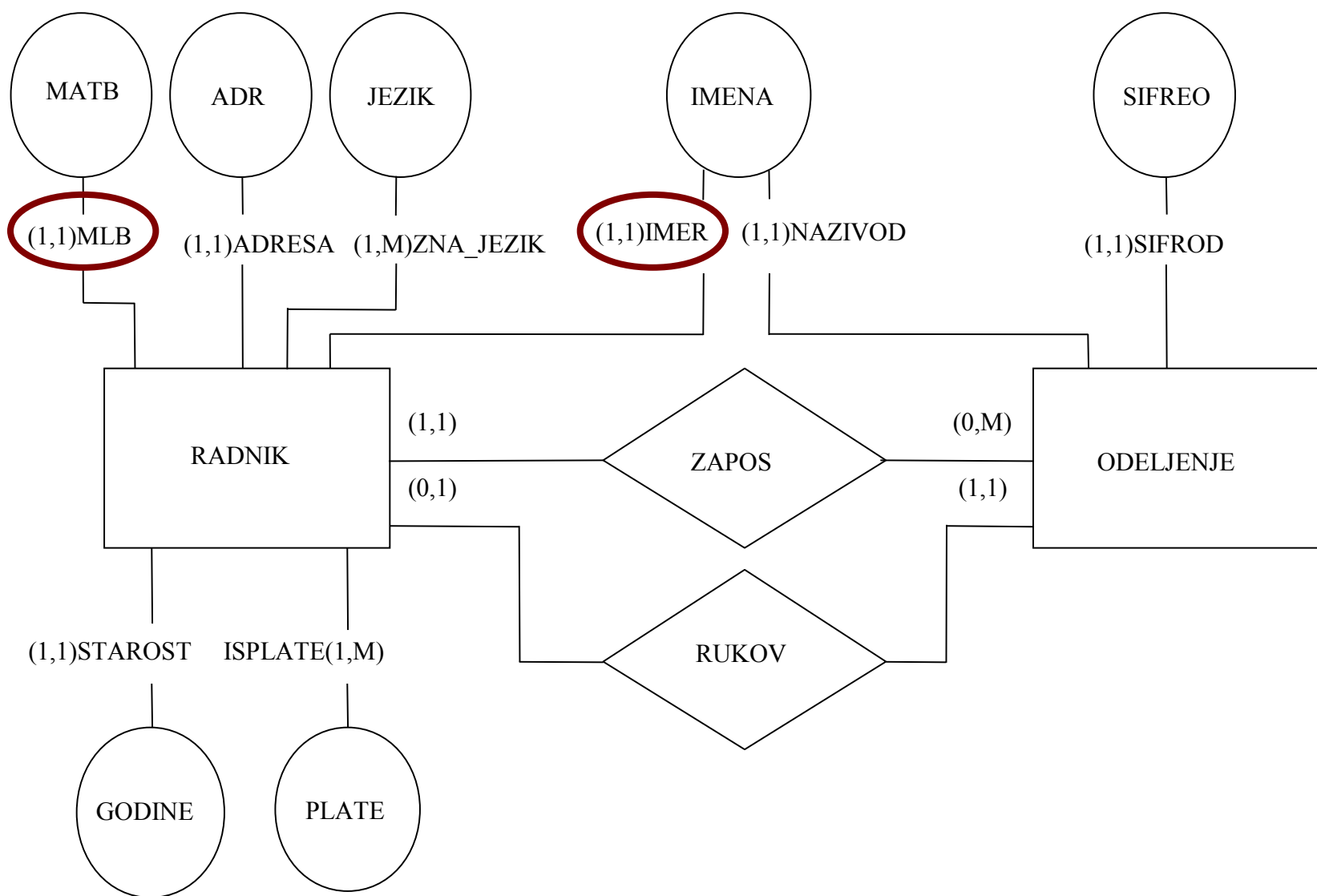
□ Složeni atributi:

Adresa (Ulica, Broj, Mesto, ...)

DatumRođenja (Dan, Mesec, Godina)

Formalno se atribut objekta može definisati kao preslikavanje iz skupa objekata datog tipa (klase objekata) u skup vrednosti (domen). Na primer,

MLB: RADNIK $\xrightarrow{(1,1)}$ **MATB**
IMER:RADNIK $\xrightarrow{(1,1)}$ **IMENA**



Definicija atributa

Jednoznačni atribut

- ✓ Ako je kardinalnost atributa ($DG = 1, GG = 1$) onda se takav atribut naziva *jednoznačni atribut objekata*.
- ✓ Ako i inverzno preslikavanje jednoznačnog atributa (preslikavanje DOMEN \rightarrow OBJEKAT) takođe ima kardinalnost ($DG = 1, GG = 1$) tada se takav atribut naziva *identifikator objekta*, jer jedno pojavljivanje takvog atributa jedinstveno određuje jedno pojavljivanje objekta u skupu pojavljivanja objekata datog tipa.
- ✓ Na primer, atribut MLB je identifikator objekata RADNIK.

MLB: RADNIK $\xrightarrow{(1,1)}$ MATB
(1,1)

Atributi identifikatori objekata posebno se označavaju (zvezdica ili podvlačenje). Vidi MLB* na slici na slajdu 27.

Višeznačni atribut

- ✓ Ako je gornja granica kardinalnost atributa $GG = M$, onda se takav atribut naziva višeznačni atribut objekta.
- ✓ Na primer, $(1, M)$
ZNA_JEZIK: RADNIK -----> JEZIK
- ✓ Zbog jednostavnije transformacije ER u implementacioni model, jednostavnijeg definisanja njegovih ograničenja i operacija, u ER se **ne koriste višeznačni atributi**.

1. Ako domen višeznačnog atributa *ima* unapred zadat, semantički značajan skup vrednosti, tad se on modelira kao klasa objekata. Na primer ZNA_JEZIK na slici na slajdu 27

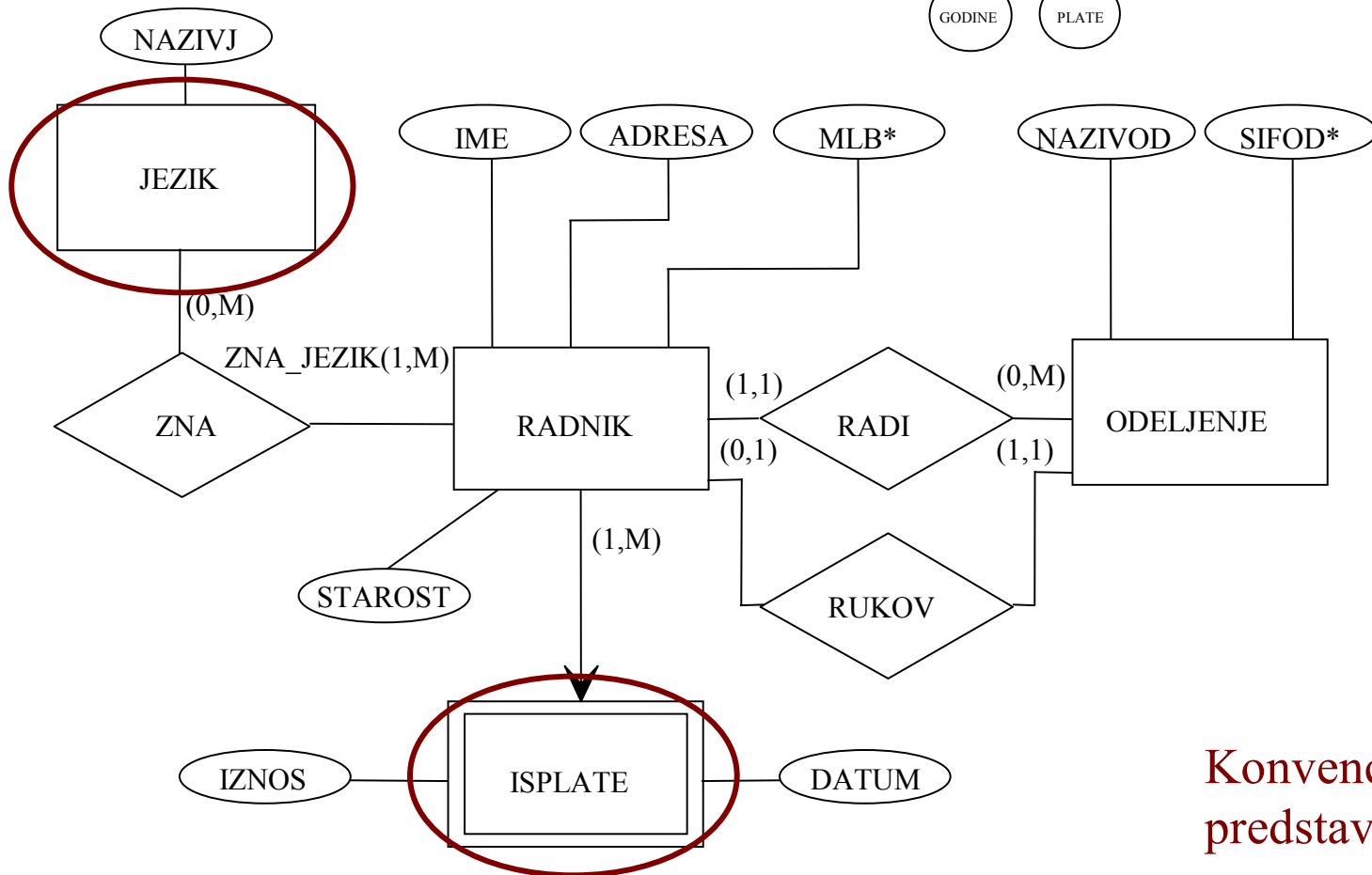
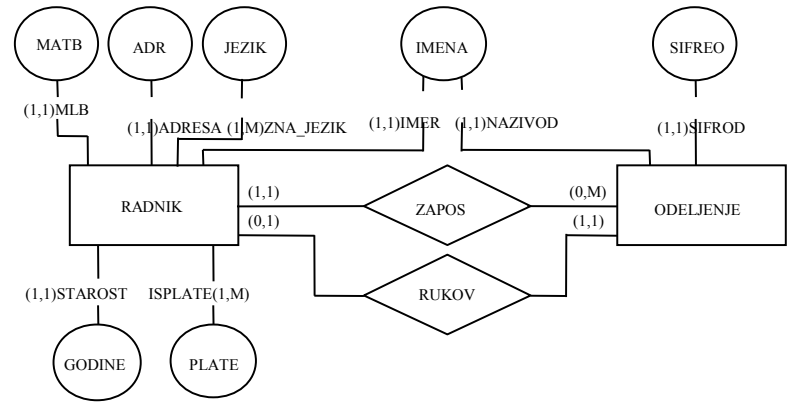
2. Ako domen višeznačnog atributa *nema* unapred zadat semantički značajan skup vrednosti, tada ga je pogodno predstaviti preko novog koncepta identifikaciono zavisnog slabog objekta. Na slici na slajdu 27 dat je primer transformacije višeznačnog atributa ISPLATE koji je bio definisan nad složenim domenom PLATE sa značenjem komponenti $\langle \text{datum}, \text{iznos} \rangle$. Sam za sebe objekat sa atributima DATUM I IZNOS nema semantički značaj (neko pojavljivanje ovog objekta ne nosi nikakvu informaciju-čija plata?), pa se zbog toga pretstavlja slabim objektom ISPLATE

MOV - pravila crtanja

Kardinalnost atributa ne mora se predstavljati
(Svi atributi moraju da budu primenljiva svojstva
na sve objekte u odgovarajućoj klasi.

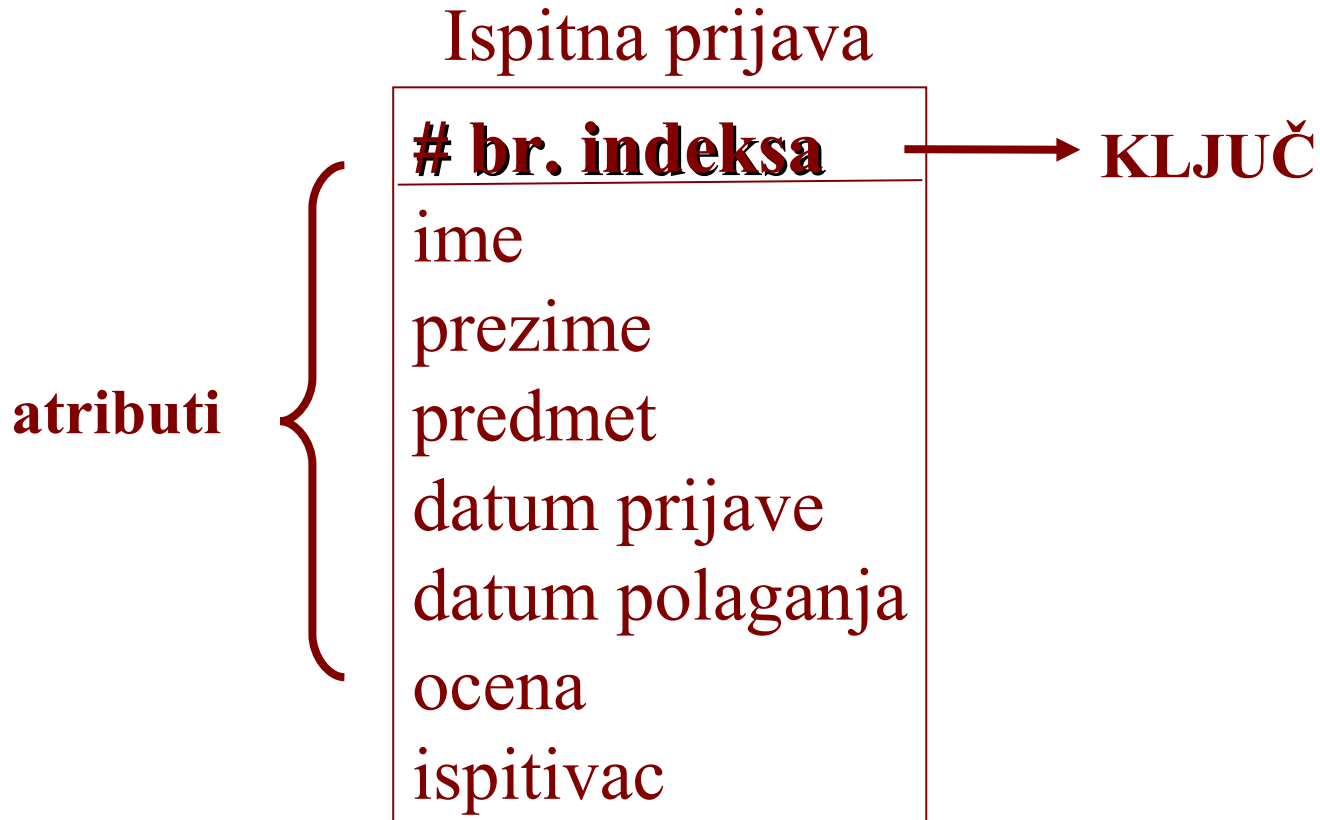
Zbog toga je donja granica preslikavanja
KLASA_OBEKATA ----> DOMEN uvek $DG = 1$.

Kako se ne koriste višeznačni atributi, to je za
ovo preslikavanje i $GG = 1$, pa se
kardinalnosti atributa ne moraju predstavljati na
ER dijagramu.).



Konvencija za predstavljanje atributa

Definisanje atributa



Neka opšta pravila

Atribut prelazi u entitet:

- ✓ ako sam atribut ima neko posebno značenje u realnom sistemu,
- ✓ atribut u osnovi identifikuje drugi tip entiteta (šifra),
- ✓ atribut je istovremeno atribut drugog entiteta.

Definisanje ključeva u modelu

- ✓ Postoje atributi (grupe atributa) čije vrednosti jednoznačno identifikuju primerke entiteta. Taj atribut (grupa atributa) naziva se *primarni ključ*.
- ✓ Ako ključ čini samo jedan atribut onda je on prost ključ, u suprotnom je složen.
- ✓ Atributi mogu biti definisani u oblasti ključeva (primarni ključ) ili u oblasti podataka.

Ključevi

- ✓ Primarni ključ
- ✓ Alternativni ključ
- ✓ Strani ključ

Ključevi

- ✓ **Alternativni ključ** predstavlja atribut ili grupu atributa koji jedinstveno identifikuju primerke entiteta, ali postoje objekti za koje taj atribut nije definisan
- ✓ **Strani (Preneseni) ključ** (Foreign Key) je atribut koji povezuje objekat 'dete' sa objektom 'roditelj'

Atributi ER modela

✓ Osnovna pravila za definisanje atributa su:

- Svaki entiteta ima proizvoljan broj atributa
- Određeni atribut pripada jednom i samo jednom entitetu
- Svako pojavljivanje entiteta ima vrednosti za sve attribute tog entiteta
- Atribut određenog pojavljivanja entiteta može imati samo jednu vrednost
- Različita pojavljivanja entiteta mogu, a ne moraju imati različite vrednosti za isti atribut
- Svaki atribut mora imati samo jedno konzistentno značenje.

Svaki atribut predstavlja jednu određenu činjenicu pa i svako značenje vrednosti atributa mora imati jedno dosledno značenje.

Definisanje ključeva u modelu

- ✓ Atributi ili grupe atributa koji mogu biti izabrani kao primarni ključevi nazivaju se 'atributi kandidati za ključ'.
- ✓ Kandidat za ključ mora jedinstveno identifikovati svaki primerak entiteta.
- ✓ Iz toga sledi da nijedan deo primarnog ključa ne može biti NULL, odnosno prazan (empty) ili nedostajući (missing).

Veze između entiteta

- ✓ U realnom sistemu objekti nisu međusobno izolovani, nego se nalaze u međusobnoj interakciji
- ✓ Odnosi između objekata posmatranja prikazuju se najčešće primenom logike skupova i preslikavanja njihovih elemenata.
- ✓ Odnosi između entitet:
 - 1:1; 1:N; N:M

EER

- ✓ Pored osnovnog, postoji i EER (prošireni model objekti veze) koji omogućava detaljnije definisanje veza između objekata.
- ✓ Pored asocijativnih veza koje oslikaju semantiku udruživanja objekata u sistemu, postoje i specifične veze kojima se izražava hijerarhija i komponovanje objekata.
- ✓ Postoje dve reprezentativne vrste ovakvih veza:
 - ❑ Specijalizacija/generalizacija
 - ❑ Agregacija

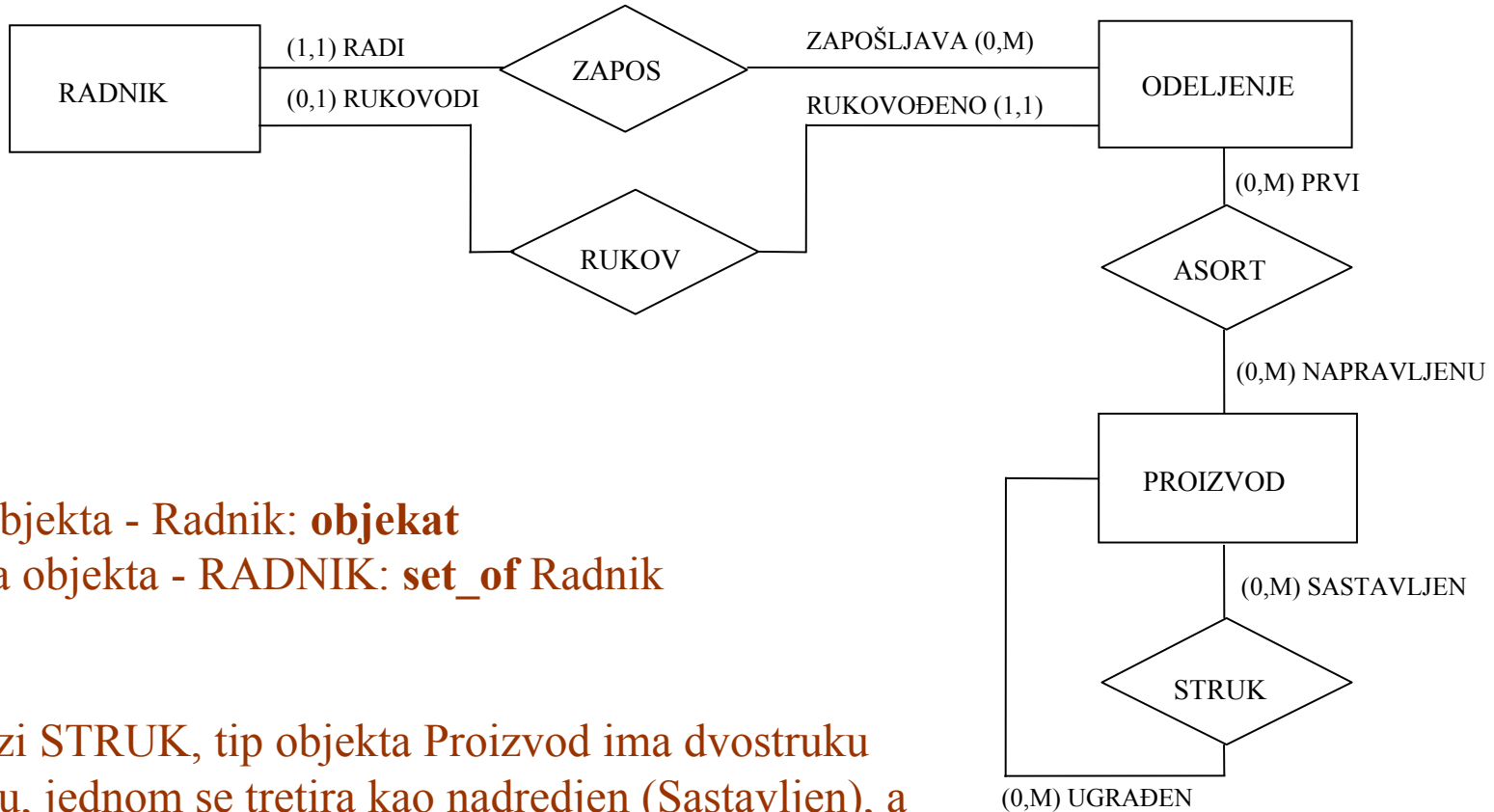
Apstrakcija podataka

- ✓ Apstrakcija je kontrolisano uključivanje detalja, "sakrivanje" detalja, odnosno "izvlačenje" opštih karakteristika u opisivanju nekog sistema.
- ✓ Postupak inverzan apstrakciji naziva se detaljisanje.
- ✓ Koristeći se različitim nivoima apstrakcije, neki složeni sistem se može istovremeno i jasno i detaljno opisati: na višim nivoima *jasno*, na nižim *detaljno*, postepenim i kontrolisanim uključivanjem detalja.

Apstrakcija podataka - Klasifikacija

- ✓ Klasifikacija ili tipizacija je apstrakcija u kojoj se *skup sličnih objekata* predstavlja jednom **klasom** objekata, odnosno svaki objekat iz posmatranog skupa odgovarajućim tipom objekta.
- ✓ Tip objekta – opšti predstavnik neke klase
- ✓ Svaki pojedinačni objekat je jedno pojavljivanje datog tipa
- ✓ Primer: *Tip objekta je Radnik*
- ✓ *Pojavljivanja tog tipa su: Jovan Jovanović, Petar Petrović, Mina Minić*
- ✓ *Skup svih pojavljivanja tipa Radnik je klasa objekata RADNIK*

Primer



tip objekta - Radnik: **objekat**

klasa objekta - RADNIK: **set_of Radnik**

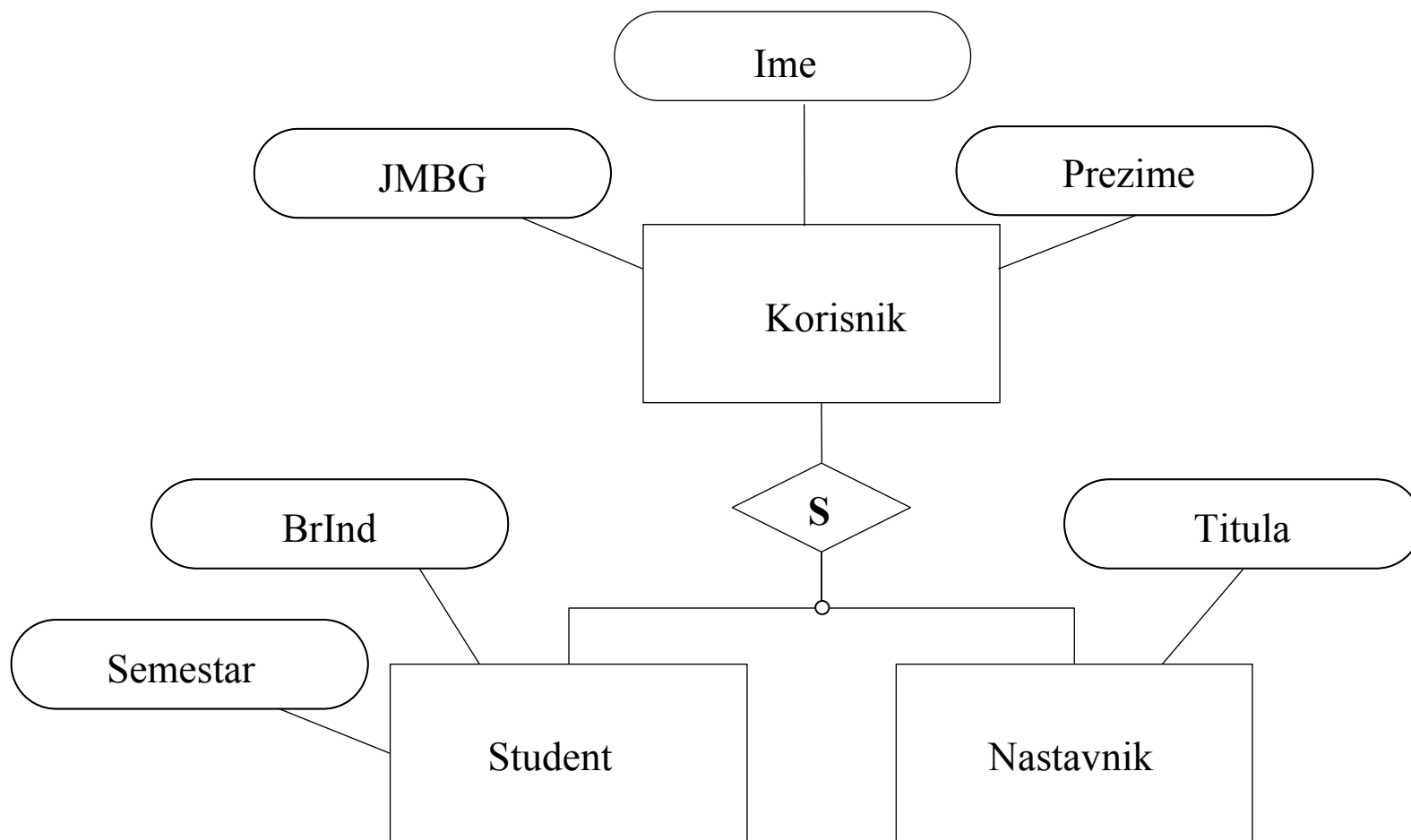
u vezi STRUK, tip objekta Proizvod ima dvostruku ulogu, jednom se tretira kao nadredjen (Sastavljen), a drugi put kao podredjen (Ugradjen) proizvod u odgovarajućoj strukturi (sastavnici).

Apstrakcija podataka – Generalizacija i specijalizacija

- ✓ Generalizacija je apstrakcija u kojoj se skup sličnih tipova objekata predstavlja opštijim generičkim tipom (nadtipom).
- ✓ Pod sličnim tipovima objekata ovde se mogu tretirati tipovi objekata koji imaju jedan broj istih (zajedničkih) atributa, tipova veza sa drugim objektima i operacija.
- ✓ *Generalizacija*
 - *Studenti, Radnici, Penzioneri, Deca su Građani*

Generalizacija i specijalizacija

Primer



Veza dva preslikavanja

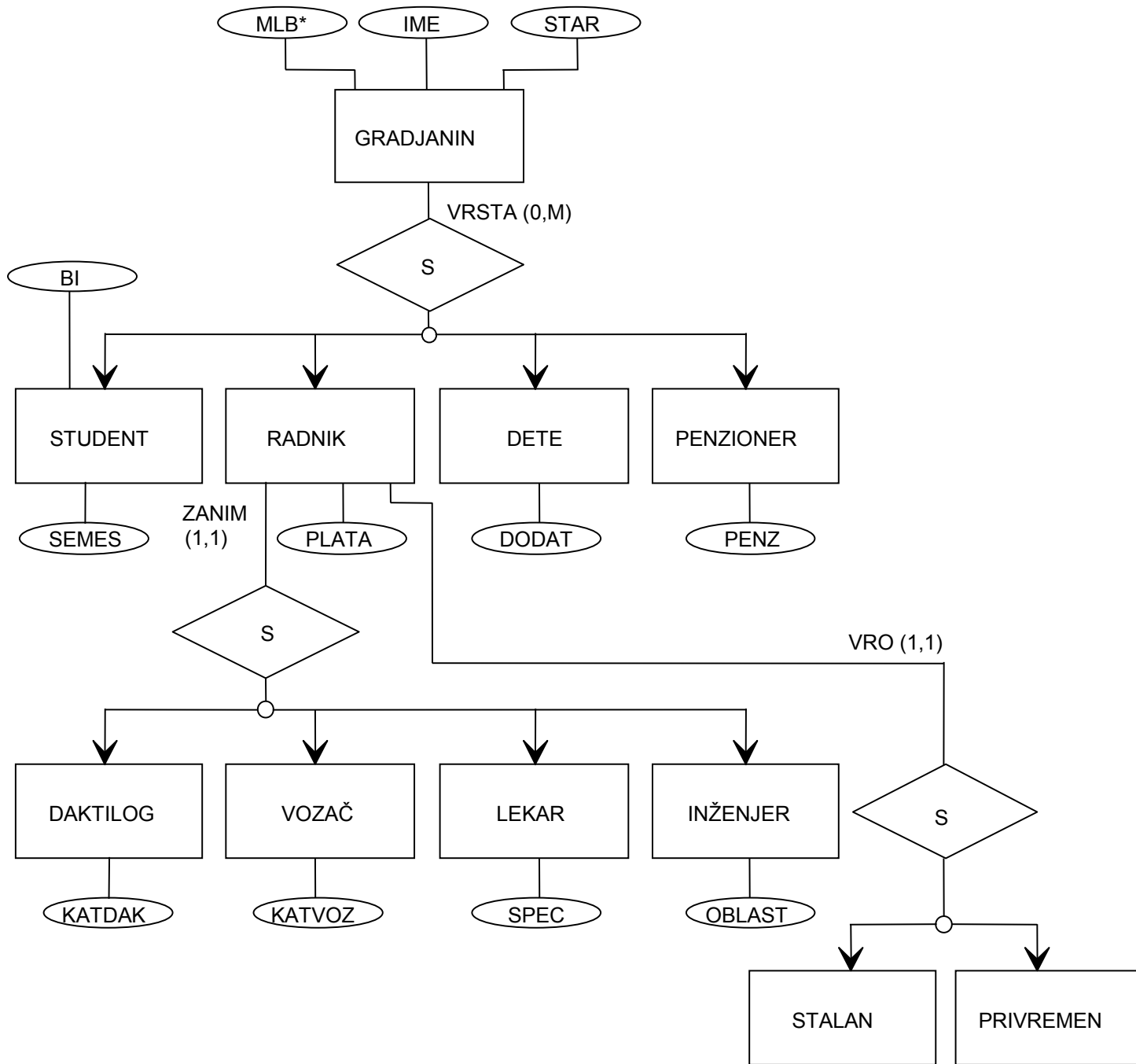
✓ Generalizacija - preslikavanje

PODTIP ----> NADTIP (trivijalno preslikavanje između podskupa i skupa za koje je uvek $DG = 1$ i $GG = 1$, kardinalnost ovoga preslikavanja se ne predstavlja na DOV) i

✓ Specijalizacija - preslikavanje

NADTIP ----> NEPOVEZANA UNIJA PODTIPOVA.

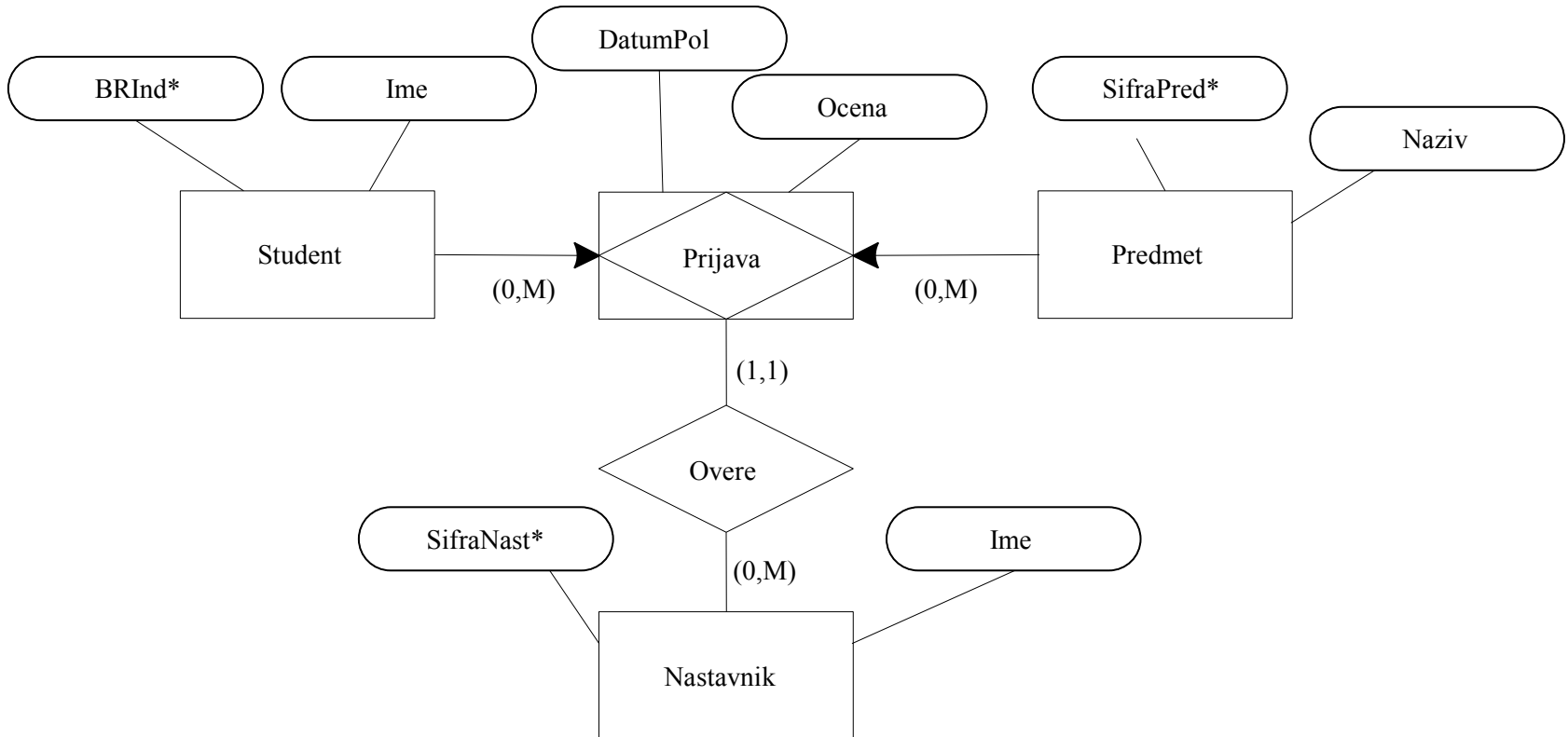
✓ Donja i gornja granica kardinalnosti specijalizacije definišu vrste specijalizacije-generalizacije:



Apstrakcija podataka – Agregacija i dekompozicija

- ✓ Agregacija je klasa veze koja se ponaša kao klasa objekata na taj način što može da učestvuje u vezama.
- ✓ **Agregacija** je apstrakcija u kojoj se *skup tipova objekata i njihovih veza* tretira kao jedinstveni agregirani tip objekta.
- ✓ Zbog toga što istovremeno predstavlja i objekat i vezu agregacija se često naziva i **mešoviti tip objekta-veza**.
- ✓ Postupak inverzan agregaciji je **dekompozicija**.

Agregacija - Primer



Agregacija

- ✓ Kardinalnost preslikavanja **KOMPONENTA** ---> **AGREGACIJA** mora biti specificirana, dok je za inverzno preslikavnaje uvek $DG = 1$ i $GG = 1$, što znači da je agregacija egzistencijalno zavisna od svojih komponenata.
- ✓ Agregirani objekat se razlikuje od ostalih objekata u sistemu po tome što nema svoj sopstveni identifikator, već ga identifikuju objekti koje on agregira.
- ✓ Jedan objekat se može predstaviti kao agregacija njegovih atributa.
- ✓ Preporuka: svaka binarna veza u kojoj oba preslikavanja imaju $GG = M$, tretira se kao agregacija.

