

Baze podataka

Predavanja

11. ER model baze podataka – 1. dio

Svibanj, 2021.



Primjer normalizacije

- Zadana je relacijska shema:

ISPIT = { matBr, prez, ime, sifPred, nazPred, datlsp, ocj, sifNas, prezNas }

i trenutna vrijednost relacije ispit(ISPIT):

ispit (ISPIT)

| matBr | prez | ime | sifPred | nazPred | datlsp | ocj | sifNas | prezNas |
|-------|-------|-------|---------|---------|----------|-----|--------|---------|
| 1111 | Novak | Ivan | 1001 | Mat-1 | 29.01.19 | 1 | 1111 | Pašić |
| 1111 | Novak | Ivan | 1001 | Mat-1 | 05.02.19 | 3 | 1111 | Pašić |
| 1111 | Novak | Ivan | 1003 | Fiz-1 | 28.06.18 | 2 | 3333 | Horvat |
| 1111 | Novak | Ivan | 1002 | Mat-2 | 27.06.18 | 4 | 2222 | Brnetić |
| 1234 | Kolar | Petar | 1001 | Mat-1 | 29.01.19 | 3 | 2222 | Brnetić |

- funkcijske zavisnosti odrediti na temelju značenja podataka
- odrediti primarni ključ relacije (tako da bude zadovoljen uvjet 1NF prema kojem neključni atributi funkcijski ovise o ključu)
- postupno normalizirati relacijsku shemu ISPIT na 2NF i 3NF

Primjer normalizacije

student (STUDENT)

| matBr | prez | ime |
|-------|-------|-------|
| 1111 | Novak | Ivan |
| 1234 | Kolar | Petar |

$K_{\text{STUDENT}} = \{ \text{matBr} \}$

predmet (PREDMET)

| sifPred | nazPred |
|---------|---------|
| 1001 | Mat-1 |
| 1003 | Fiz-1 |
| 1002 | Mat-2 |

$K_{\text{PREDMET}} = \{ \text{sifPred} \}$

nastavnik (NASTAVNIK)

| sifNas | prezNas |
|--------|---------|
| 1111 | Pašić |
| 3333 | Horvat |
| 2222 | Brnetić |

$K_{\text{NASTAVNIK}} = \{ \text{sifNas} \}$

ispit₃ (ISPIT₃)

| matBr | sifPred | datlsp | ocj | sifNas |
|-------|---------|----------|-----|--------|
| 1111 | 1001 | 29.01.19 | 1 | 1111 |
| 1111 | 1001 | 05.02.19 | 3 | 1111 |
| 1111 | 1003 | 28.06.18 | 2 | 3333 |
| 1111 | 1002 | 27.06.18 | 4 | 2222 |
| 1234 | 1001 | 29.01.19 | 3 | 2222 |

$K_{\text{ISPIT}_3} = \{ \text{matBr}, \text{sifPred}, \text{datlsp} \}$

- Shema baze podataka STUSLU:

$\text{STUSLU} = \{ \text{STUDENT}, \text{PREDMET}, \text{NASTAVNIK}, \text{ISPIT}_3 \}$

Implementacija: SQL

```
CREATE TABLE student (  
    matBr      INTEGER  
    , prez     CHAR(20)  
    , ime      CHAR(20)  
    , PRIMARY KEY (matBr));
```

```
CREATE TABLE predmet (  
    sifPred    INTEGER  
    , nazPred  CHAR(20)  
    , PRIMARY KEY (sifPred));
```

```
CREATE TABLE nastavnik (  
    sifNas     INTEGER  
    , prezNas  CHAR(20)  
    , PRIMARY KEY (sifNas));
```

```
CREATE TABLE ispit (  
    , matBr      INTEGER REFERENCES student (matBr)  
                        ON DELETE CASCADE  
    , sifPred    INTEGER REFERENCES predmet (sifPred)  
    , datIsp     DATE  
    , ocj        SMALLINT CHECK (ocj BETWEEN 1 AND 5)  
    , sifNas     INTEGER REFERENCES nastavnik(sifNas)  
                        ON DELETE SET NULL  
    , PRIMARY KEY (matBr, sifPred, datIsp));
```

ER model (*Entity-Relationship Model*) Model entiteti-veze

- postrelacijski model
- zadržava dobre karakteristike relacijskog modela
- omogućuje eksplicitni prikaz veza koje u sebi sadrže važne semantičke informacije

Literatura:

- P.P.Chen:
The Entity-Relationship Model - Toward a Unified View of Data,
ACM Transactions on Database Systems, Vol. 1, No. 1, 1976
- T. J. Teorey:
Database Modeling & Design, Morgan Kaufmann, 2011
- FER-ZPR: ERDPlus – upute, 2000
https://www.fer.unizg.hr/_download/repository/ERDplus_-_upute.pdf
Upute koje će se koristiti za treću domaću zadaću te za drugu i treću laboratorijsku vježbu.
Alat koristi modificiranu notaciju – Martinovu ili Crow's foot notation.

Entiteti, veze, uloge

Entitet

- bilo što, što ima suštinu ili bit, ima jasnoću kao činjenica ili ideja, posjeduje značajke s pomoću kojih se može razlučiti od svoje okoline

Skup entiteta E_i (*entityset*)

- Slični entiteti se grupiraju u skupove entiteta

Skup veza R_i (*relationship set*)

- matematička relacija između n entiteta:

$$R_i \subseteq E_1 \times E_2 \times E_3 \times \dots \times E_n$$

$$\text{ili } R_i = \{ (e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n \}$$

n -torka $(e_1, e_2, e_3, \dots, e_n)$, naziva se vezom.

Uloga (*role*)

- funkcija koju skup entiteta obavlja u skupu veza.

Skup vrijednosti, atribut

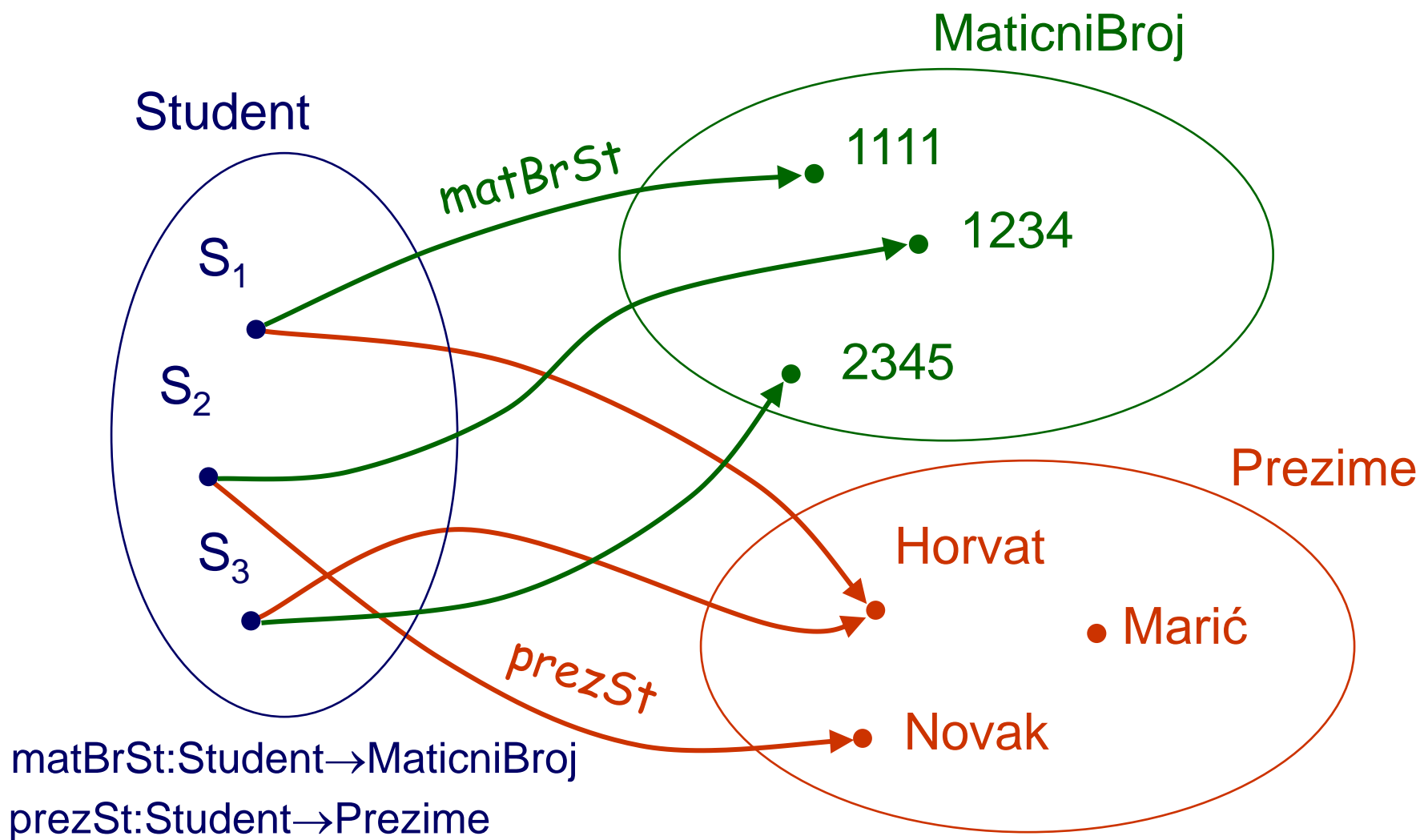
- Informacije o entitetu ili vezi izražavaju se s pomoću parova **atribut-vrijednost**
- Vrijednosti su klasificirane u skupove vrijednosti V_i .
- **Atribut** je funkcija koja preslikava iz skupa entiteta ili skupa veza u skup vrijednosti:

$$f : E_i \rightarrow V_i$$

$$f : R_i \rightarrow V_i$$

Atributi entiteta

- funkcija koja preslikava sa skupa entiteta na skup vrijednosti



Terminologija

Chen:

entitet, skup entiteta

veza, skup veza

Teorey:

instanca entiteta, entitet

(entity instance)

(entity occurrence)

instanca veze, veza

(relationship instance)

(relationship occurrence)

Grafički prikaz entiteta i veza

- entitet se grafički prikazuje pravokutnikom unutar kojeg se nalazi ime entiteta
- veza se grafički prikazuje rombom unutar kojeg se nalazi ime veze

Entiteti



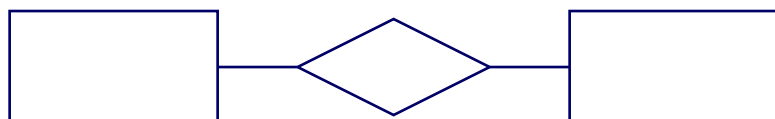
Veza



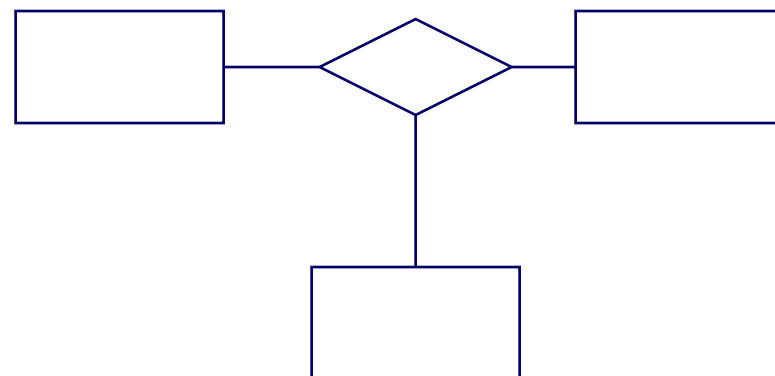
Stupanj veze

- broj entiteta koje povezuje dotična veza
- veza može biti unarna(refleksivna), binarna, ternarna, itd.
 - unarna ili refleksivna veza - veza je definirana nad jednim entitetom koji u vezi ima dvije različite uloge

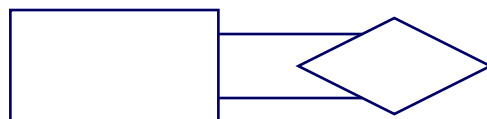
binarna



ternarna



refleksivna



Spojnost veze (*connectivity*)

- **spojnost veze** opisuje ograničenje preslikavanja pojedinačnih entiteta koje veza povezuje
- vrijednosti spojnosti: jedan (*one*), više (*many*)
- koriste se oznake 1, N ili rasponi, npr. 0..1, 1..N, 1..2, itd.



- jedan djelatnik radi na jednom projektu, na jednom projektu radi N djelatnika



- jedan djelatnik radi na nula (niti jednom) ili jednom projektu, na jednom projektu radi između nula (niti jedan) i više djelatnika

Spojnost veze (*connectivity*)

- radi pojednostavljenja
 - spojnost 0..N se često označava samo oznakom N
 - spojnost 1..1 se često označava samo oznakom 1

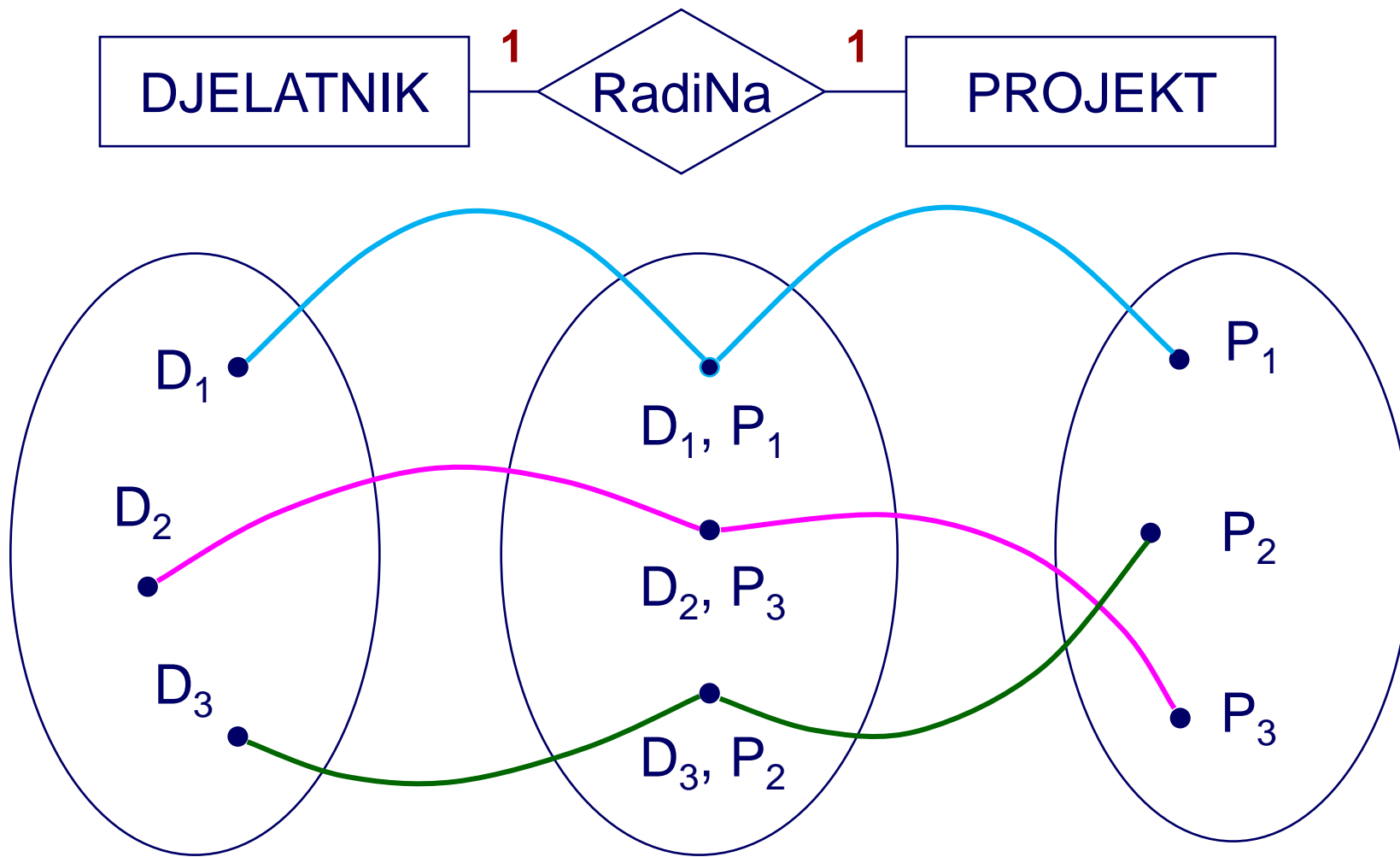


Preslikavanje (*mapping*)

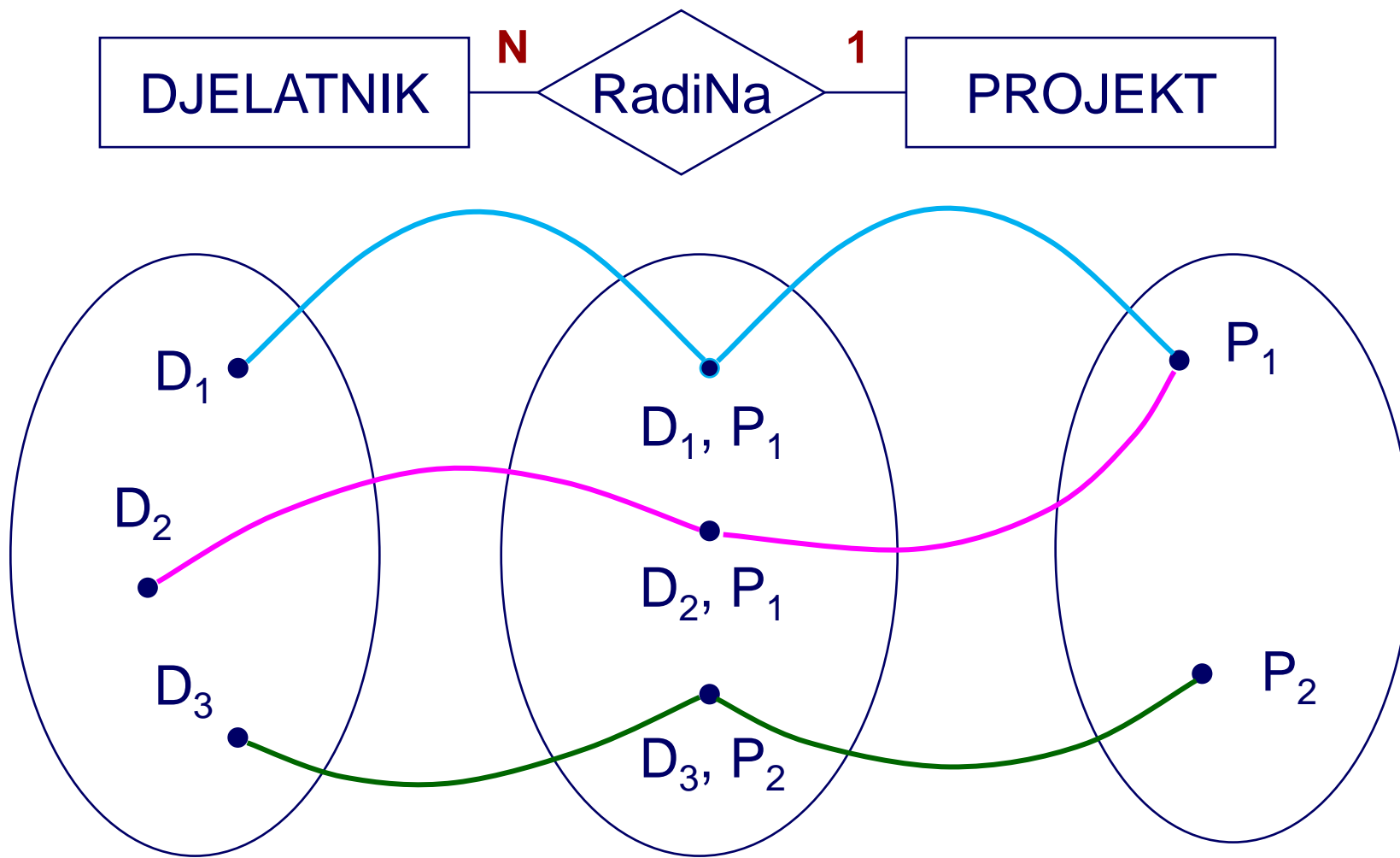
- **preslikavanje** - međusobni odnos entiteta u vezi
- kod binarnih veza moguća su **preslikavanja 1:1** (jedan-prema-jedan), **1:N** (jedan-prema-više), **N:1** (više-prema-jedan), **N:N** (više-prema-više).



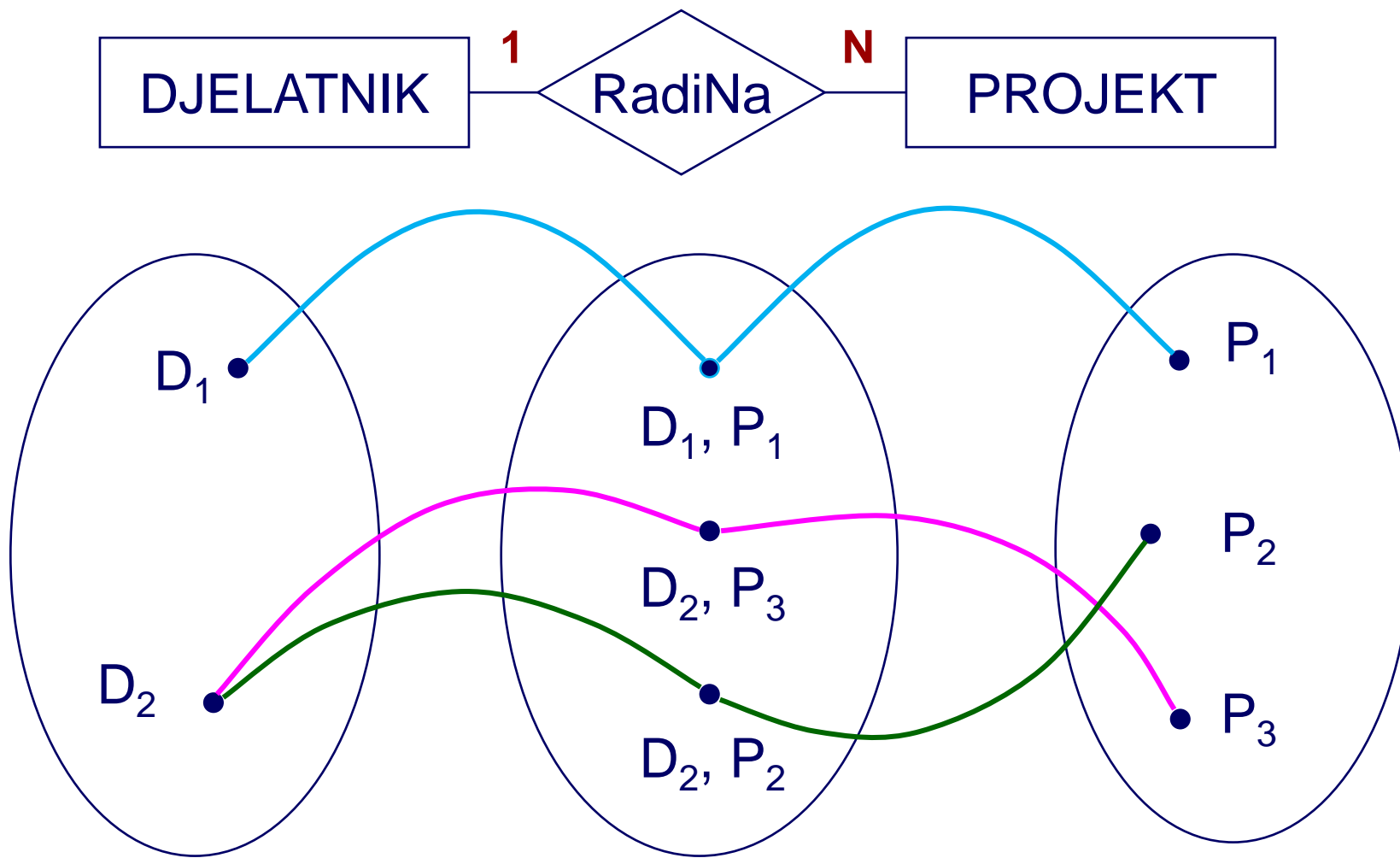
Preslikavanje 1:1 ?



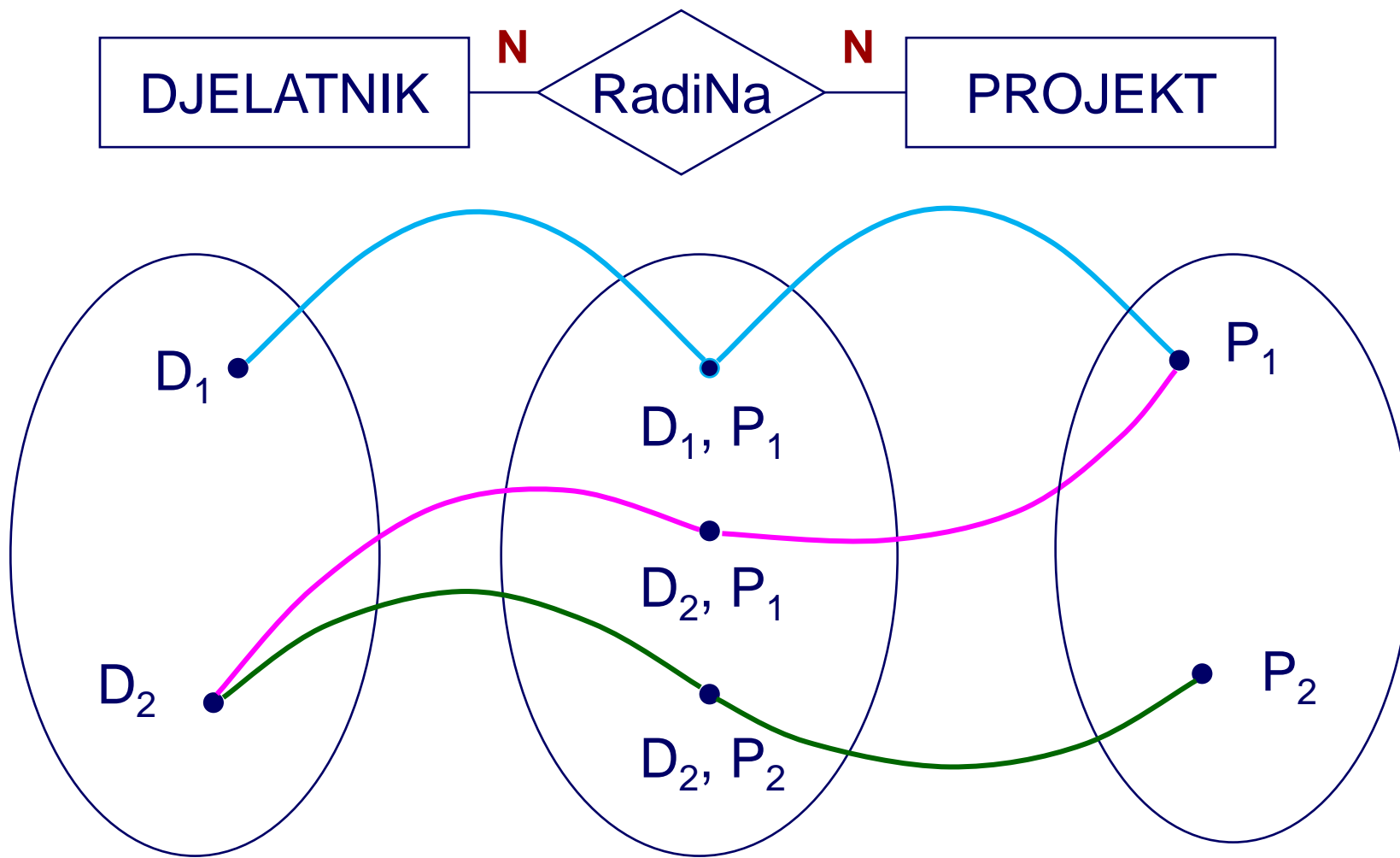
Preslikavanje N:1



Preslikavanje 1:N



Preslikavanje N:N

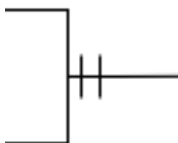


Različita notacija

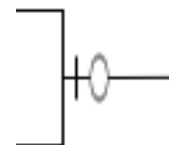
- Ova predavanja oslanjaju se na originalnu Chenova notaciju
- Alat ERDPlus koji će se koristiti za laboratorijske vježbe ima drugačiju notaciju za spojnosti, preslikavanje i slabi entitet.
- Spojnosti

Chen ERDPlus

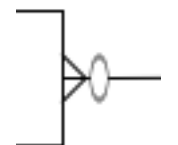
1



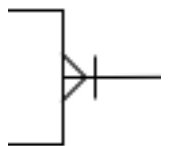
0:1



N



1:N

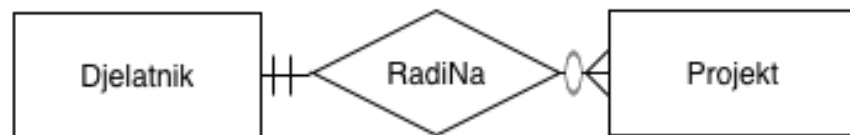


Preslikavanje

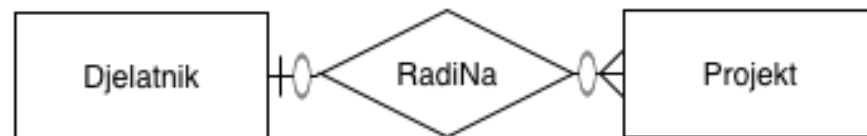
Chen

ERDPlus

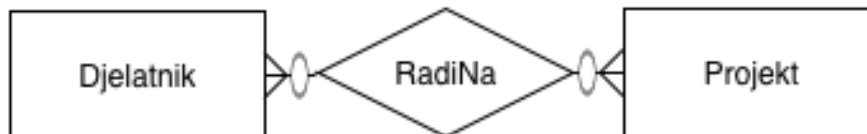
1:N



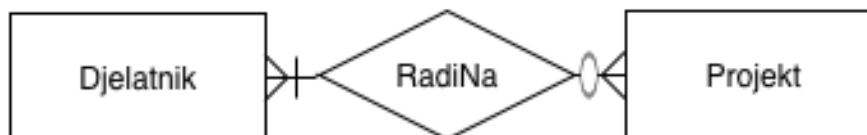
0..1:N



N:N

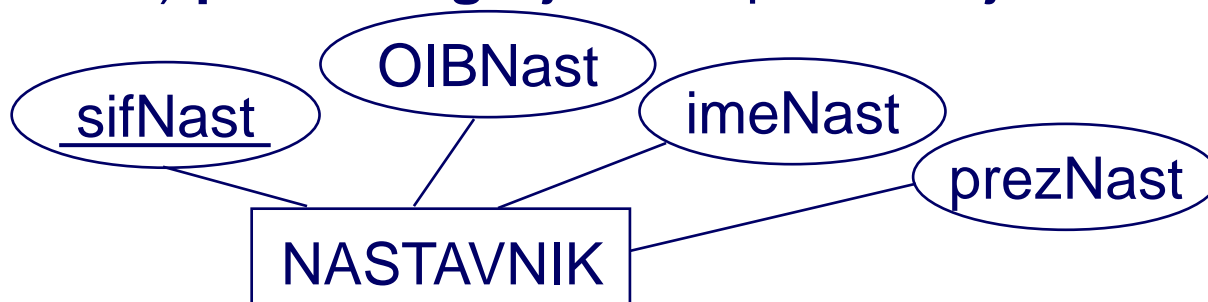


1..N:N



Atributi entiteta

- atribut entiteta se grafički prikazuju ovalom unutar kojeg se upisuje ime atributa
- atribut (ili atributi) **primarnog** ključa se potcrtavaju



- povećanjem broja atributa, dijagram postaje nepregledan
 - atributi se tada ne prikazuju grafički - umjesto toga, uz dijagram se prilažu **sheme entiteta**

Shema entiteta:

NASTAVNIK = sifNast, OIBNast, imeNast, prezNast

PK = { sifNast }

ili

NASTAVNIK

sifNast

OIBNast

imeNast

prezNast

$K_1 = \{ \text{sifNast} \}$

$K_2 = \{ \text{OIBNast} \}$

PK = K_1

Regularni i slabi entiteti

- **regularni** entitet je entitet koji može postojati sam za sebe
- **slabi** entiteti (engl. *weak entity*) ne postoje ukoliko ne postoji i neki drugi entitet (entitet vlasnik)
- Slabi entitet se grafički prikazuje dvostruko uokvirenim pravokutnikom, sa strelicom koja dolazi iz smjera veze koja ga povezuje s entitetom vlasnikom



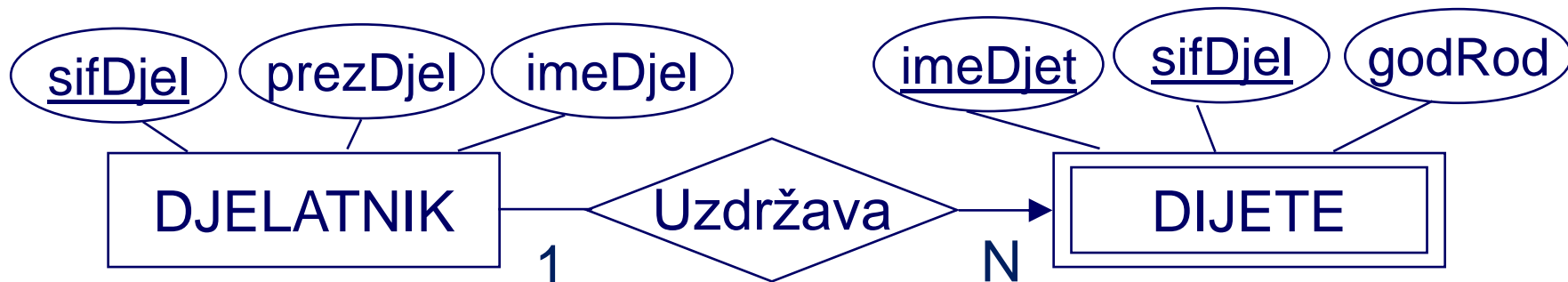
ERDPlus:



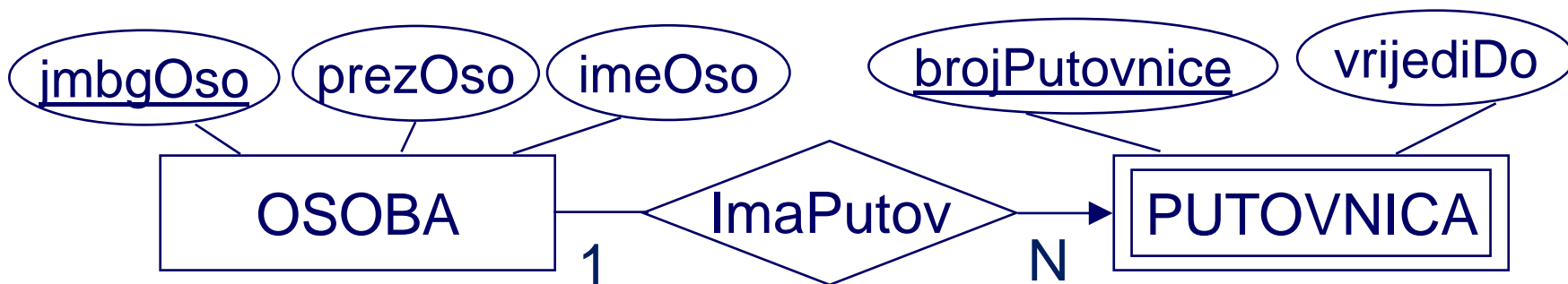
- slabi entiteti, osim što su **egzistencijalno slabi**, također mogu biti i **identifikacijski slabi**
 - kod određivanja identifikatora nisu im dovoljni vlastiti atributi
 - za identifikaciju se koriste i ključni atributi entiteta vlasnika

Identifikacijski slabi entiteti (primjer)

- entitet DIJETE, osim što je egzistencijalno slab, također je i identifikacijski slab

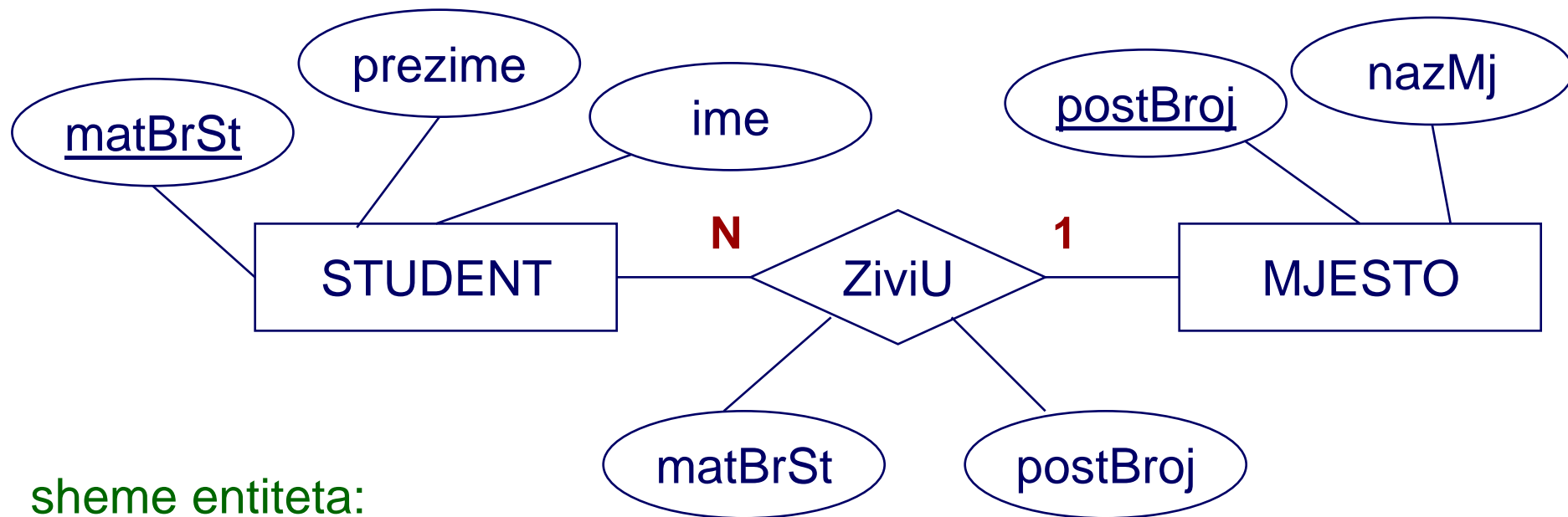


- entitet PUTOVNICA je egzistencijalno slab (nije identifikacijski slab)



Atributi veza – shema veze

- Shema veze sadrži ključeve entiteta koje povezuje, te vlastite attribute



sheme entiteta:

STUDENT = matBrSt, prezime, ime

MJESTO = postBroj, nazMj

shema veze:

ZiviU = matBrSt, postBroj

Koji atributi čine ključ veze?

Ključevi veza

- Povezanost entiteta opisuje se kao odnos među ključevima entiteta
- Ključevi veza definirani su s pomoću ključeva entiteta koje povezuju i njihovih spojnosti

Definicija 1. (Teorey)

U vezi koja povezuje entitete

$E_1, \dots, E_k, \dots, E_m$,

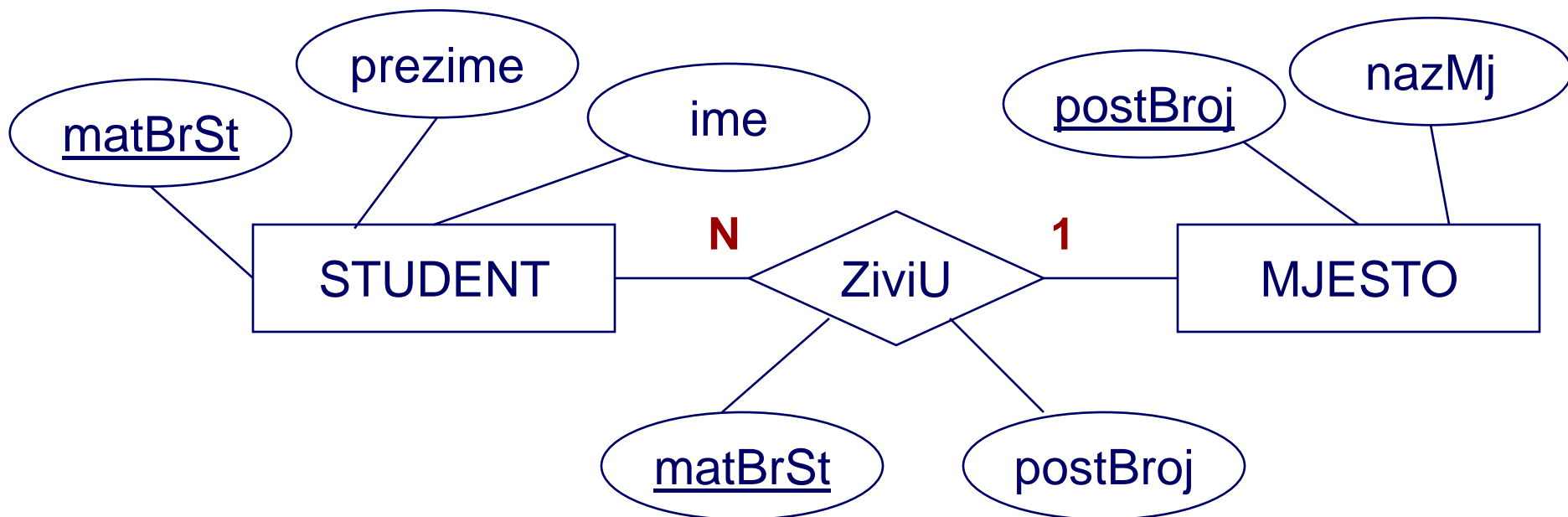
spojnost =1 entiteta E_k znači da za svaku vrijednost svih entiteta E_1, \dots, E_m , osim E_k , uvijek postoji točno jedna vrijednost od E_k .

□ može se reći da tada vrijedi funkcijska zavisnost:

$$\bigcup_{j=1}^m K_j \setminus K_k \rightarrow K_k$$

gdje su skupovi K_j , ($j = 1, \dots, m$) ključevi entiteta E_1, \dots, E_m

Ključevi veza



sheme entiteta:

STUDENT = matBrSt, prezime, ime

MJESTO = postBroj, nazMj

shema veze:

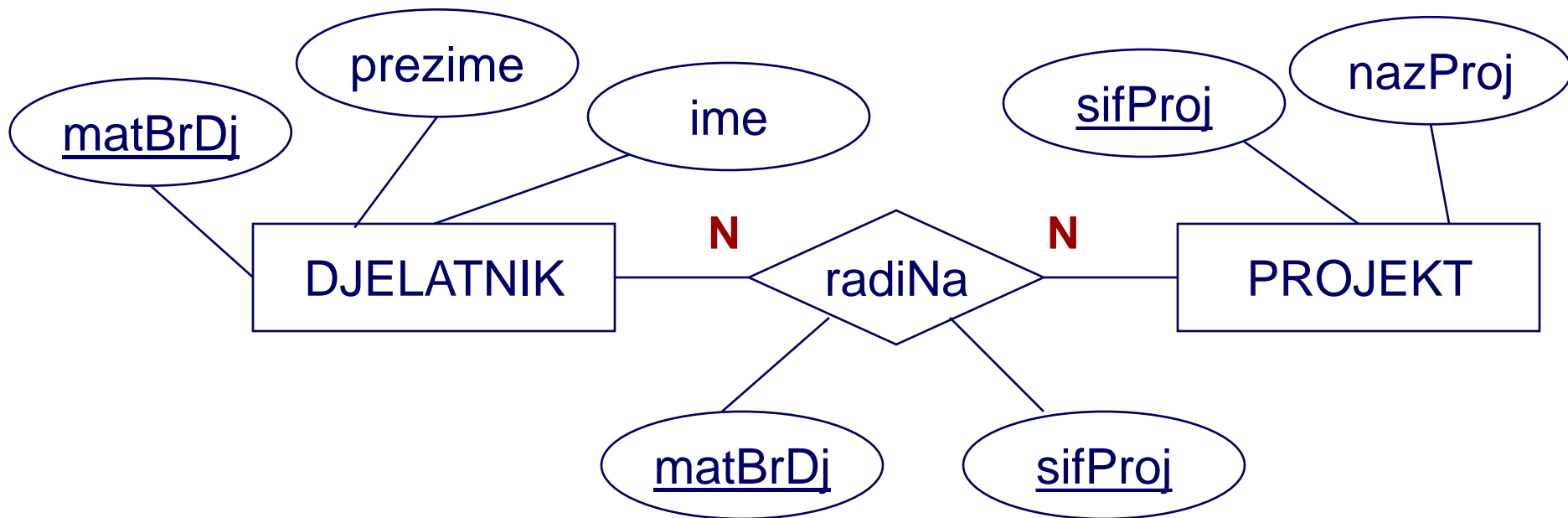
ZiviU = matBrSt, postBroj



Iz definicije 1:

$\text{matBrSt} \rightarrow \text{postBroj}$

Ključevi veza

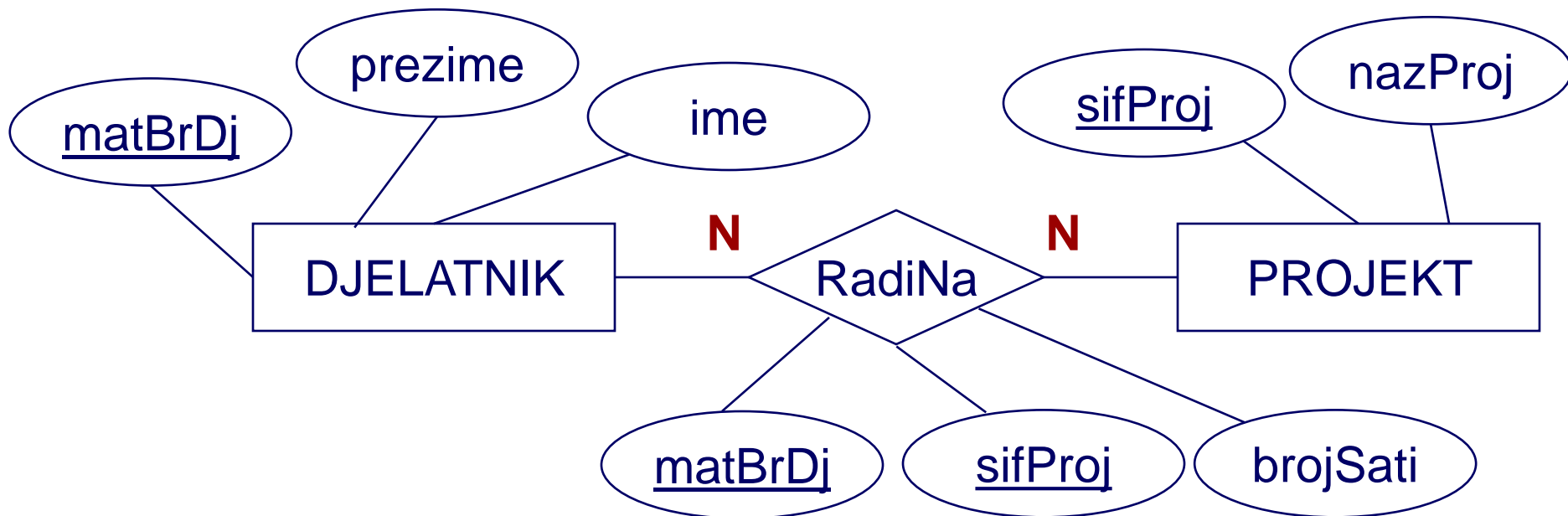


DJELATNIK = matBrDj, prezime, ime

PROJEKT = sifProj, nazProj

RadiNa = matBrDj, sifProj

Vlastiti atributi veza



DJELATNIK = matBrDj, prezime, ime

PROJEKT = sifProj, nazProj

RadiNa = matBrDj, sifProj, brojSati

ključ veze funkcijski određuje
vlastite attribute veze:
 $\text{matBrDj, sifProj} \rightarrow \text{brojSati}$

Ključ veze - dodatna razmatranje

- iz definicije 1. proizlazi da se ključ veze sastoji isključivo od ključeva entiteta koje povezuje (svih ili samo nekih, ovisno o spojnostima)

Međutim, u nekim slučajevima ključ može sadržavati i neke druge atribute.



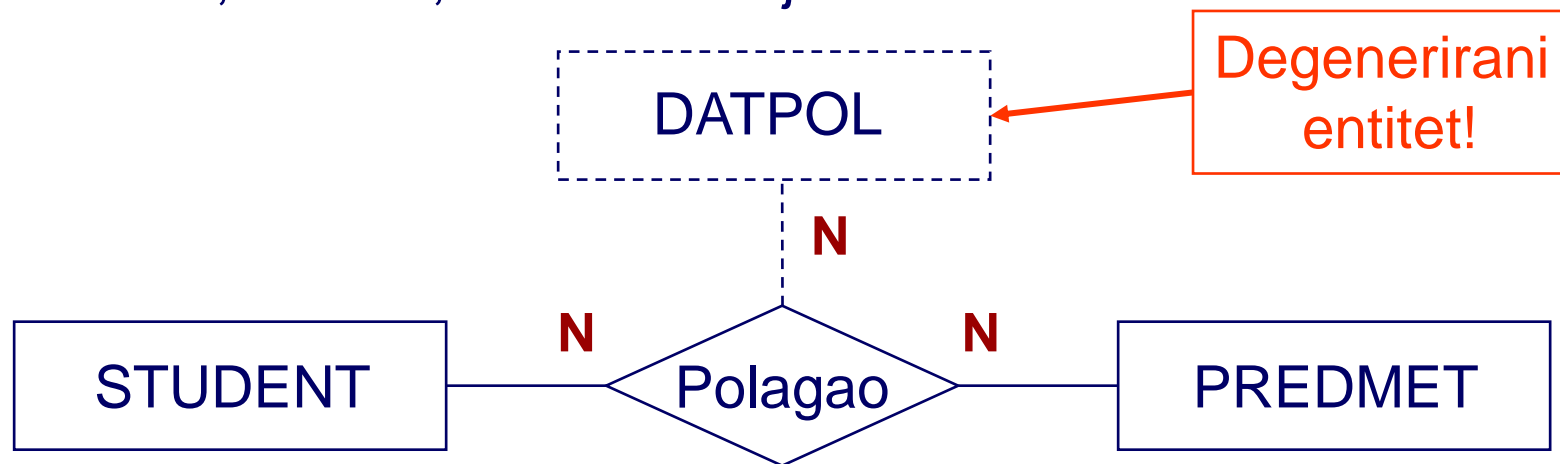
STUDENT = matBrSt, prezime, ime

PREDMET = sifPred, nazPred

Položio = matBrSt, sifPred, ocjena

Ključ veze - dodatna razmatranje

- ako se želi evidentirati sva polaganja ispita
matBrSt, sifPred \nrightarrow ocjena
- potrebno je uvesti atribut datPol (datum polaganja):
matBrSt, sifPred, datPol \rightarrow ocjena



STUDENT = matBrSt, prezime, ime

PREDMET = sifPred, nazPred

Polagao = matBrSt, sifPred, datPol, ocjena

Ključ veze - dodatna razmatranje

- druga mogućnost - veza postaje entitet:



STUDENT = matBrSt, prezime, ime

PREDMET = sifPred, nazPred

ISPIT = matBrSt, sifPred, datPol, ocjena

Studlsp = matBrSt, sifPred, datPol

Predlsp = matBrSt, sifPred, datPol

Veza 1:N → preslikavanje u relacijski model



DJELATNIK = matBrDj, prezime, ime

MJESTO = postBr, nazMjesto

Stanuje = matBrDj, postBr, adresa

Relacijske sheme opisuju entitete (veze postaju entiteti)

DJELATNIK = matBrDj, prezime, ime

MJESTO = postBr, nazMjesto

Stanuje = matBrDj, postBr, adresa

Unija relacijskih shema s jednakim ključevima

DJELATNIK = matBrDj, prezime, ime, postBr, adresa

MJESTO = postBr, nazMjesto

Veza N:N → preslikavanje u relacijski model



DJELATNIK= matBrDj, prezime, ime

PROJEKT= sifProj, nazProj

RadiNa = matBrDj, sifProj, brojSati

Relacijske sheme opisuju entitete (veze postaju entiteti)

DJELATNIK= matBrDj, prezime, ime

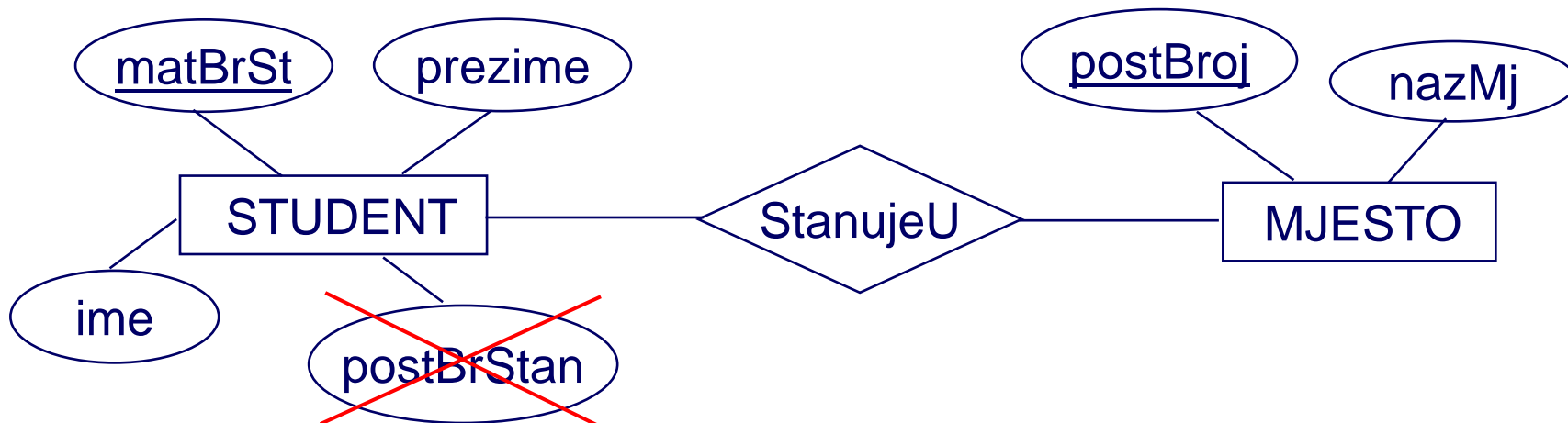
PROJEKT= sifProj, nazProj

RadiNa = matBrDj, sifProj, brojSati

Vlastiti atributi entiteta

Entiteti se opisuju samo vlastitim atributima

- **vlastiti atribut entiteta** je atribut koji opisuje znanja o entitetu koja se pripisuju isključivo samom entitetu, a nikako vezi s drugim entitetima



- isključivo **identifikacijski slabi entiteti**, osim svojih vlastitih atributa, posjeduju i attribute primarnog ključa entiteta vlasnika

Primjer: zašto je važno ispravno odrediti vlastite attribute entiteta i veza?

- Entiteti se opisuju samo vlastitim atributima: **vlastiti atribut entiteta** je atribut koji opisuje znanja o entitetu koja se pripisuju isključivo samom entitetu, a nikako vezi s drugim entitetima



DJELATNIK = matBrDj, prezime, ime, brojSati ← nije vlastiti atribut

PROJEKT = sifProj, nazProj

RadiNa = matBrDj, sifProj

- Ako se preslikavanje promijeni u N:N



matBrDj ↗ brojSati

atribut brojSati se iz entiteta DJELATNIK mora premjestiti u vezu RadiNa

Primjer: zašto je važno ispravno odrediti vlastite attribute entiteta i veza?



DJELATNIK= matBrDj, prezime, ime

PROJEKT= sifProj, nazProj

RadiNa = matBrDj, sifProj, brojSati ← **vlastiti atribut**

- Ako se preslikavanje promijeni u N:N

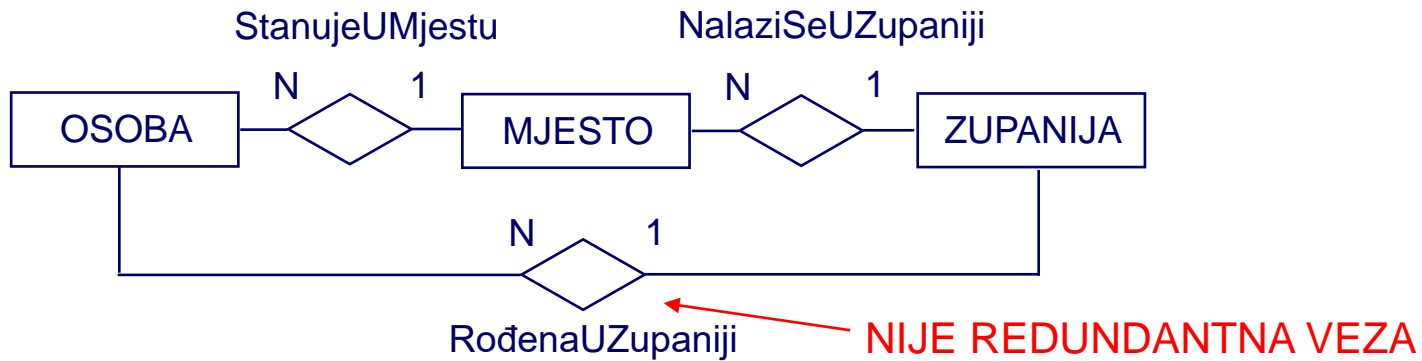
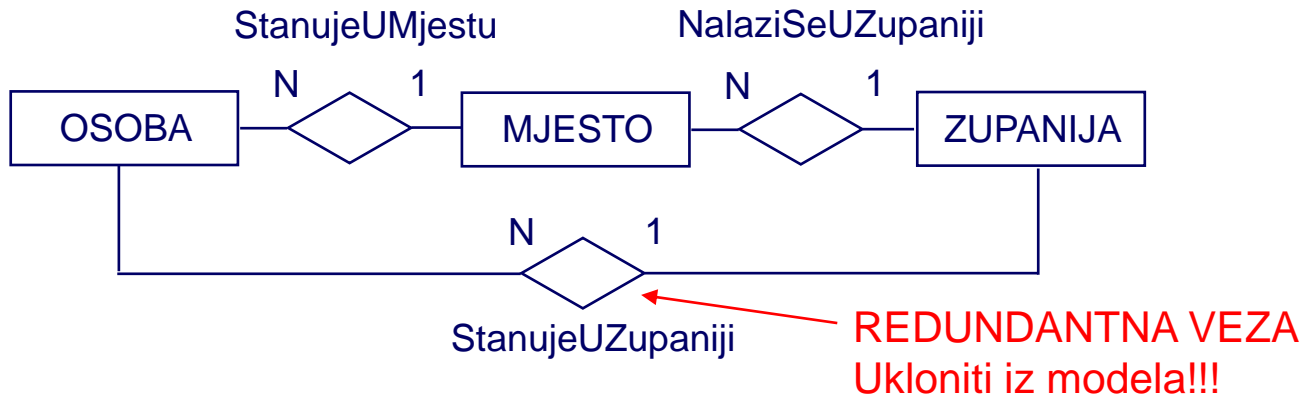


DJELATNIK= matBrDj, prezime, ime

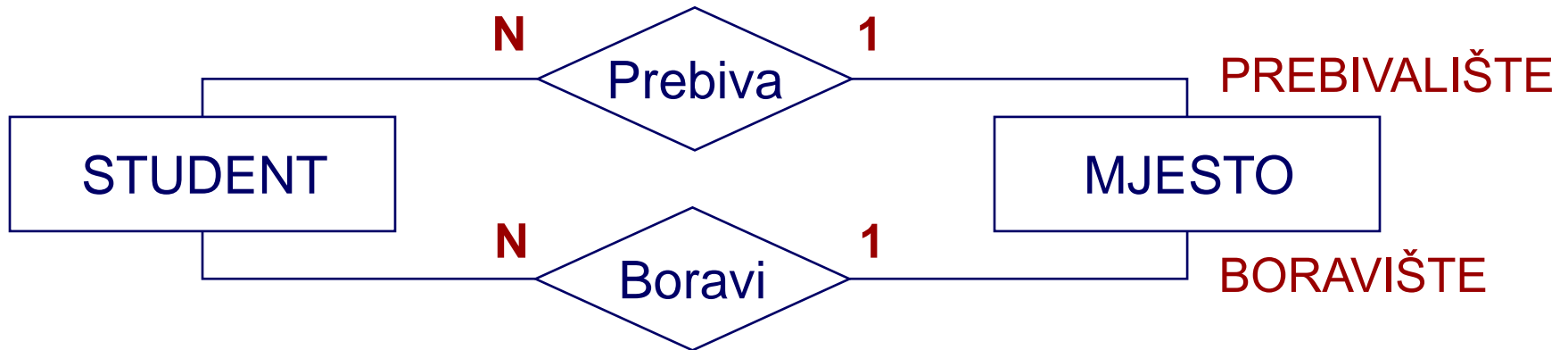
PROJEKT= sifProj, nazProj

RadiNa = matBrDj, sifProj, brojSati

Redundantne veze



Paralelne veze



STUDENT = matBrSt, prezime, ime

MJESTO = postBroj, nazMjesto

Uloge: **PREBIVALIŠTE**
BORAVIŠTE

Prebiva = matBrSt, postBroj PostBrojPreb

Boravi = matBrSt, postBroj PostBrojBor

Paralelne veze → relacijski model

- Unija shema s jednakim ključevima:

MJESTO = postBroj, nazMjesto

STUDENT = matBrSt, prezime, ime, ~~postBroj, postBroj~~

STUDENT = matBrSt, prezime, ime, postBrojBor, postBrojPreb
+ pravila integriteta

Zadatak: Ispisati prezime i ime studenta, poštanski broj i naziv mjesta boravka te poštanski broj i naziv mjesta prebivališta

```
SELECT student.*  
    , boraviste.nazMjesto AS nazMjestoBoraviste  
    , prebivaliste.nazMjesto AS nazMjestoPrebivaliste  
FROM student  
    JOIN mjesto AS boraviste  
        ON boraviste.postBroj = student.postBrojBor  
    JOIN mjesto AS prebivaliste  
        ON prebivaliste.postBroj = student.postBrojPreb
```

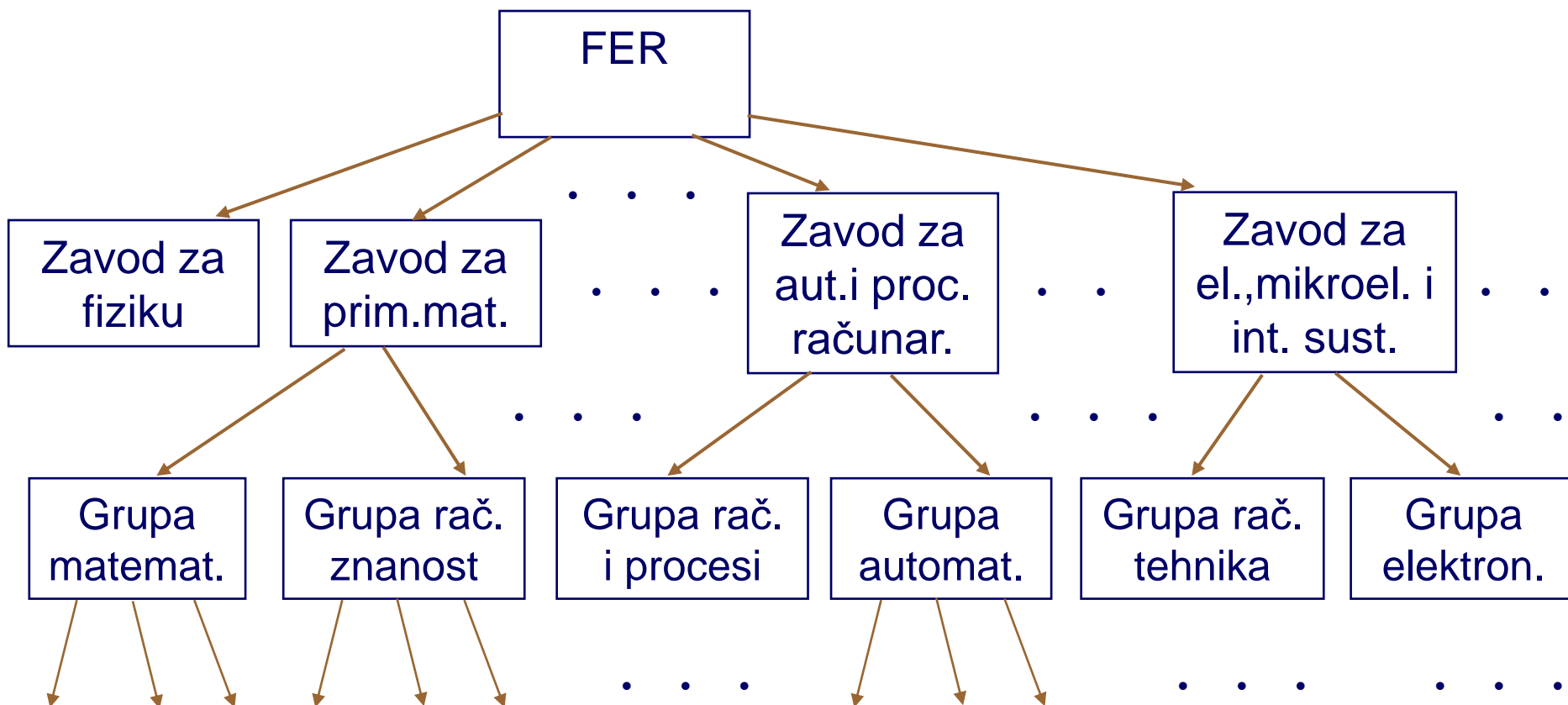

Problem

Kako opisati organizacijsku strukturu poduzeća?

- Organizacijske jedinice opisane su svojom šifrom i nazivom
- Organizacijske jedinice međusobno su povezane
 - **kako?**
 - među njima postoji hijerarhijski odnos!
 - **kolika je dubina stabla (broj razina)?**
 - promjenjiva!
- Kako opisati hijerarhiju - stablo promjenjive dubine?
- Čvorovi stabla su opisani na isti način (šifra, naziv)

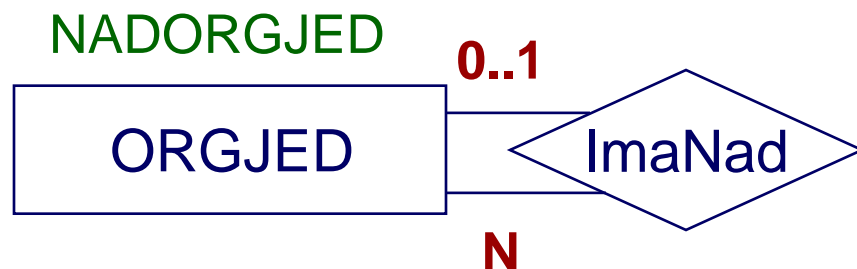
Homogeno stablo

Primjer: Organizacijska struktura



Čvorovi stabla imaju jednaku strukturu: ORGJED= sifOrgJed, nazOrgJed

Refleksivne veze - preslikavanje 1:N



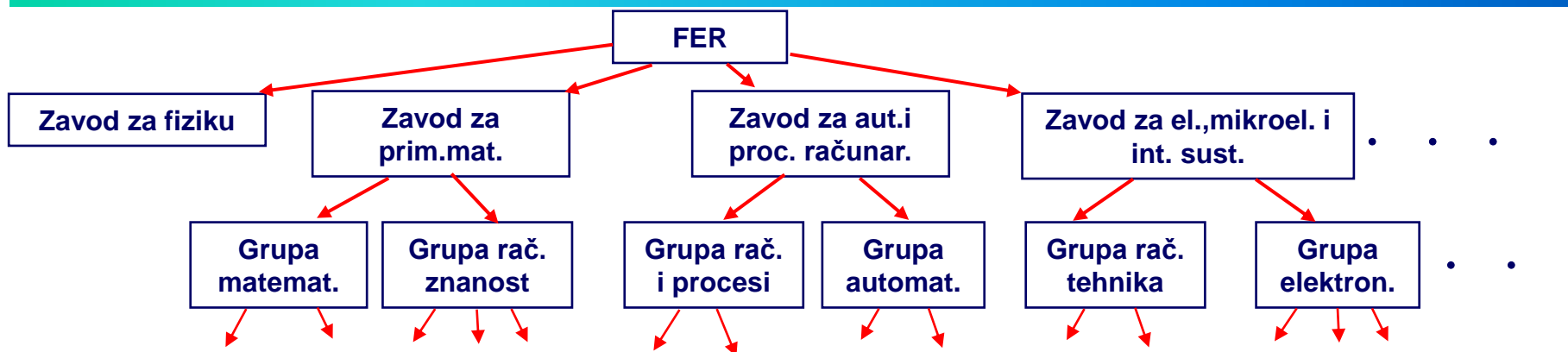
ORGJED = sifOrgJed, nazOrgJed

ImaNad = ~~sifOrgJed, sifOrgJed~~

ImaNad = sifOrgJed, sifNadOrgJed

Preimenovati
jedan od atributa !

Homogeno stablo



ORGJED

| sifOrgJed | nazOrgJed |
|-----------|-----------------------------|
| 1 | FER |
| 9 | Zavod za prim.mat. |
| 21 | Grupa Matematika |
| 33 | Grupa Rač. Znanost |
| 49 | Zavod za aut. i proc. rač. |
| 53 | Grupa Automatika |
| 67 | Grupa RASIP |
| 73 | Zavod za el.mikroel. i int. |
| 89 | Grupa Rač. tehnika |

imaNad

| sifOrgJed | sifNadOrgJed |
|-----------|--------------|
| 9 | 1 |
| 21 | 9 |
| 33 | 9 |
| 49 | 1 |
| 53 | 49 |
| 67 | 49 |
| 73 | 1 |
| 89 | 73 |

Refleksivne veze 1:N → relacijski model

- Unija shema s jednakim ključevima:

ORGJED = sifOrgJed, nazOrgJed

imaNad = sifOrgJed, sifNadOrgJed

ORGJED = sifOrgJed, nazOrgJed, sifNadOrgJed

+ pravila integriteta

Zadatak: Ispisati naziv organizacijske jedinice i naziv njezine nadređene organizacijske jedinice (ukoliko postoji)

```
SELECT orgjed.nazOrgJed, nadorgjed.nazOrgJed
FROM orgjed
      LEFT OUTER JOIN nadorgjed AS nadorgjed
      ON orgjed.sifNadOrgJed = nadorgjed.sifOrgJed
```

Što je šifra organizacijske jedinice?

- Govoreća šifra – šifra koja označava poziciju organizacijske jedinice unutar poduzeća??

npr. XXYYZZZ

XX – šifra sektora

YY – šifra odjela

ZZZ – šifra odsjeka

□ što se dešava prilikom reorganizacije?

- moraju se promijeniti šifre organizacijskih jedinica!

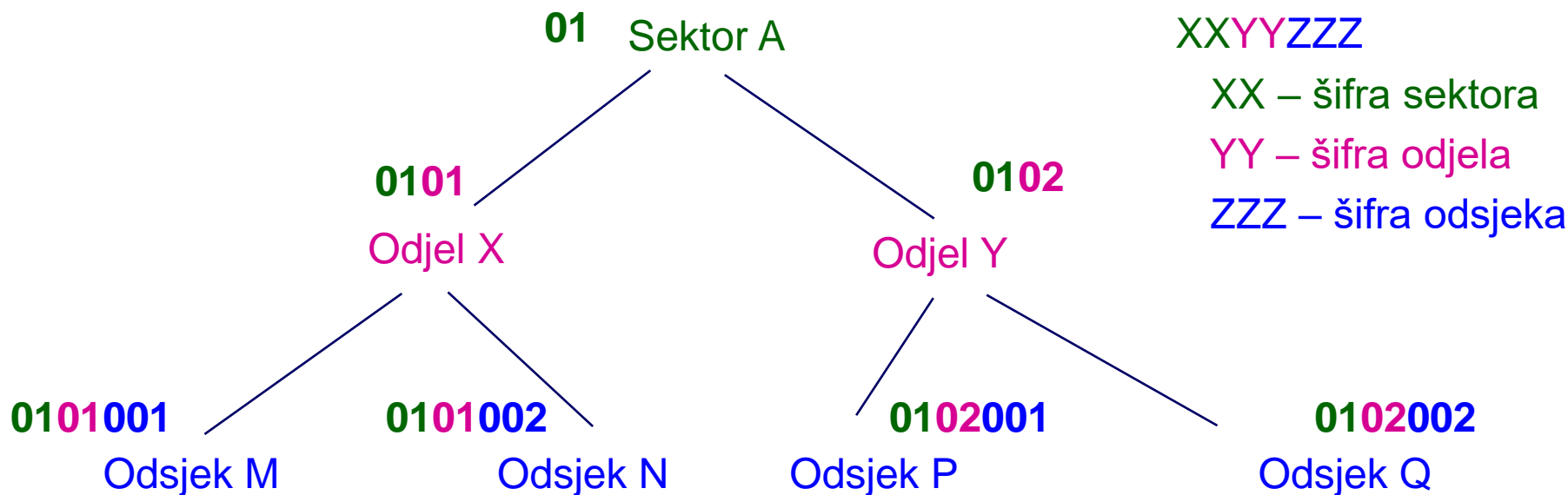
□ što se dešava kada broj odjela preraste 100??

- moraju se promijeniti šifre organizacijskih jedinica!

□ Šifra organizacijske jedinice NE SMIJE BITI GOVOREĆA!

□ To vrijedi i za sve ostale šifre i identifikatore!!!

Što je šifra organizacijske jedinice?

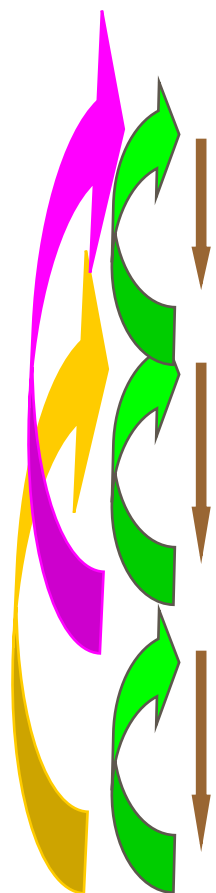


| ORGJED | sifOrgJed | nazOrgJed |
|--------|-----------|-----------|
| | 01 | Sektor A |
| | 0101 | Odjel X |
| | 0102 | Odjel Y |
| | 0101001 | Odsjek M |
| | 0101002 | Odsjek N |
| | 0102001 | Odsjek P |
| | 0102002 | Odsjek Q |

Što kada Odsjek P zbog reorganizacije iz Odjela Y preseli u Odjel X?

Što kada broj odjela preraste broj 99?

Oblikovanje ER modela



Oblikovanje ER modela

- **definiranje entiteta**
 - ime, opis, komentar
- **definiranje veza**
 - ime, opis, komentar, entiteti koje povezuje, preslikavanje
- **definiranje atributa entiteta**
 - za svaki atribut: ime, opis, komentar, domena
 - definirati ključeve, provjeriti da li zadovoljava 3NF
- **definiranje atributa veza**
 - za svaki atribut: ime, opis, komentar, domena
 - definirati ključeve, provjeriti da li zadovoljava 3NF

POSTUPAK JE ITERATIVAN!

Model baze podataka

SADRŽI OPISE

- entiteta
- veza
- atributa entiteta
- atributa veza

KARAKTERISTIKE DOBROG MODELA

- opisuje suštinu, prirodu stvari, neovisan o postojećem stanju
- sveobuhvatan
- neredundantan
- fleksibilan
- razumljiv - korisnicima i informatičarima

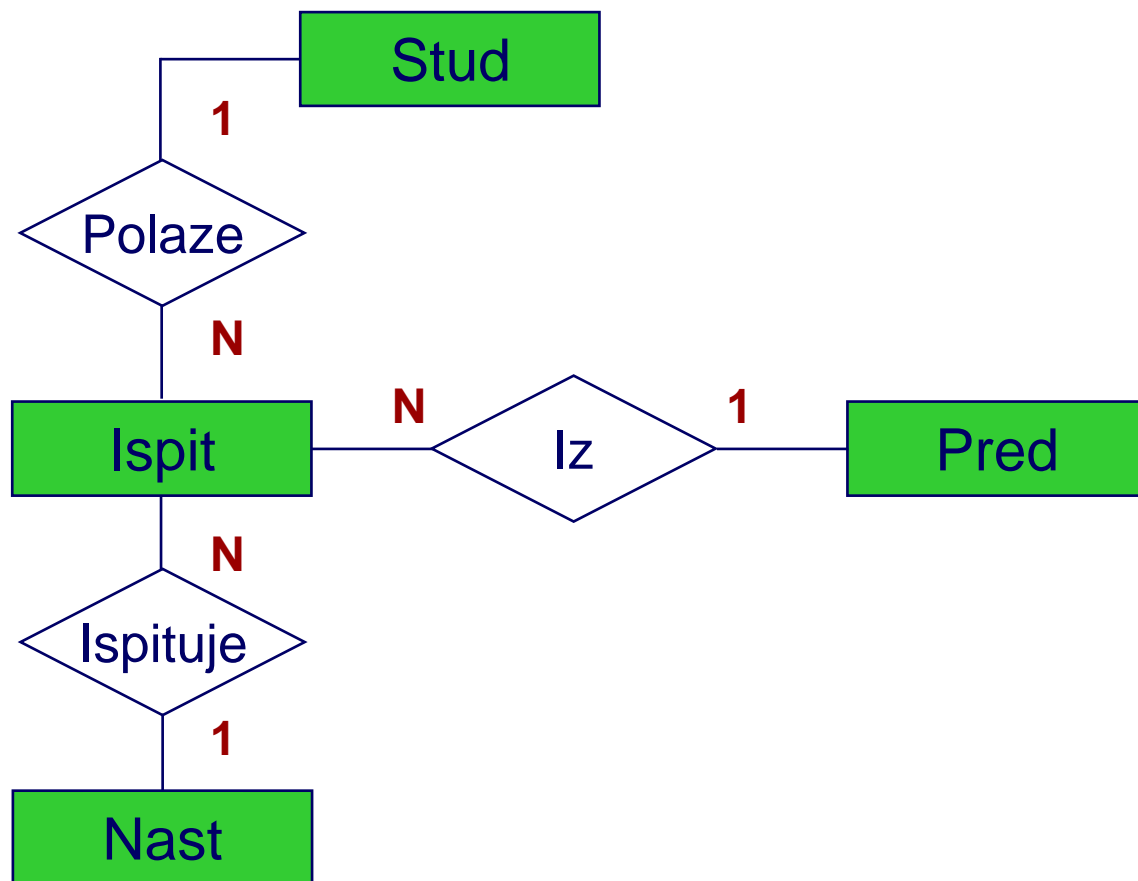
POSEBNO OBRATITI PAŽNJU NA:

- različito shvaćanje istih stvari - kupac, dobavljač ≠ poslovni partner
- praćenje promjena u vremenu - stipendist, djelatnik, umirovljenik
- jednakost - uopćavanje - različiti odjeli i pojedinci mogu iste ili slične stvari shvaćati različito

Primjer: Model baze podataka za studentsku službu

- Oblikovati model baze podataka koja će omogućiti praćenje podataka o studentima, predmetima, nastavnicima i polaganjima ispita

Primjer (nastavak)

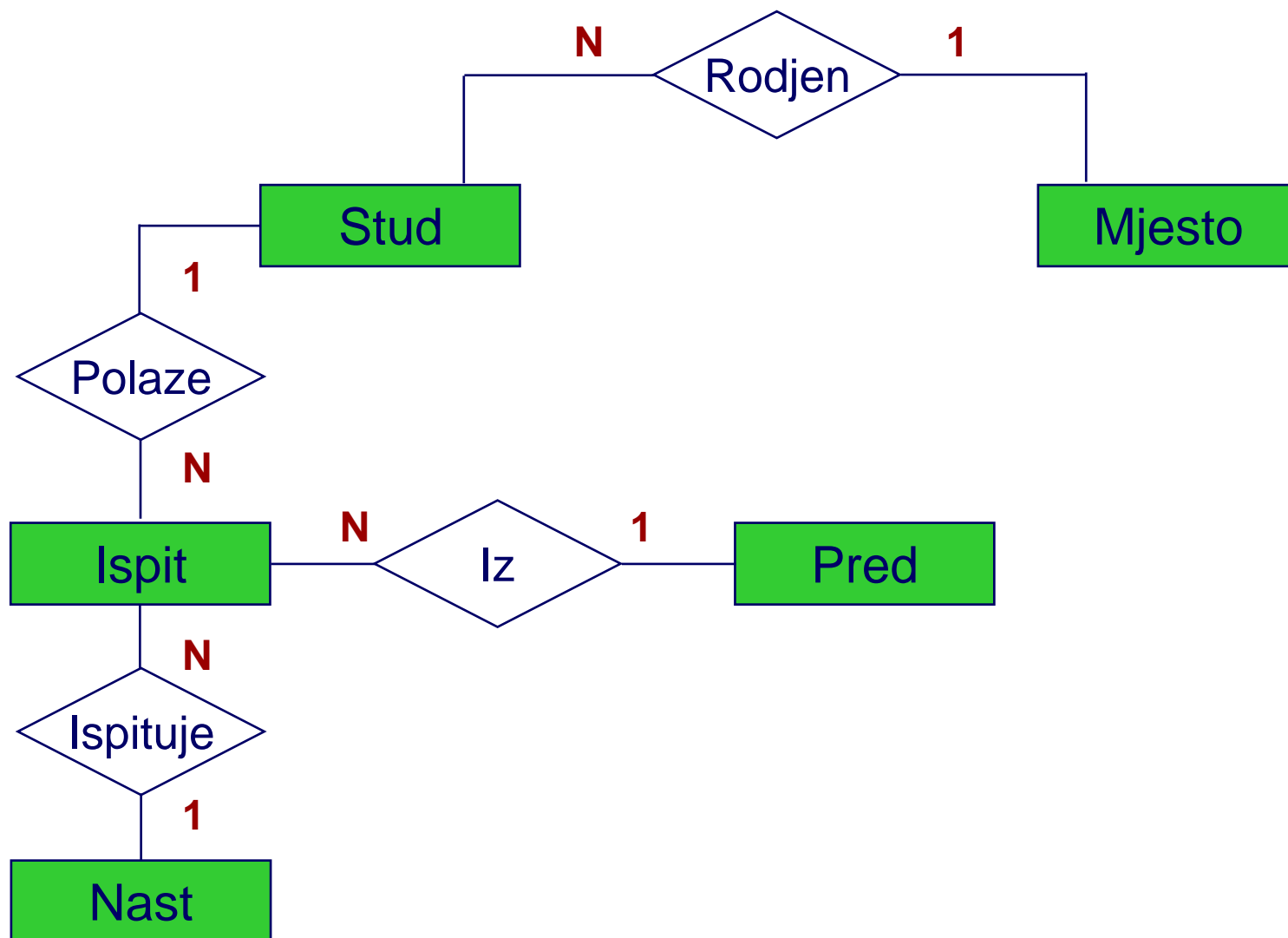


Primjer (nastavak)

Stud = matBrStud, prezStud, imeStud, datRodStud

MJESTO ROĐENJA STUDENTA ???

Primjer (nastavak)



Primjer (nastavak)

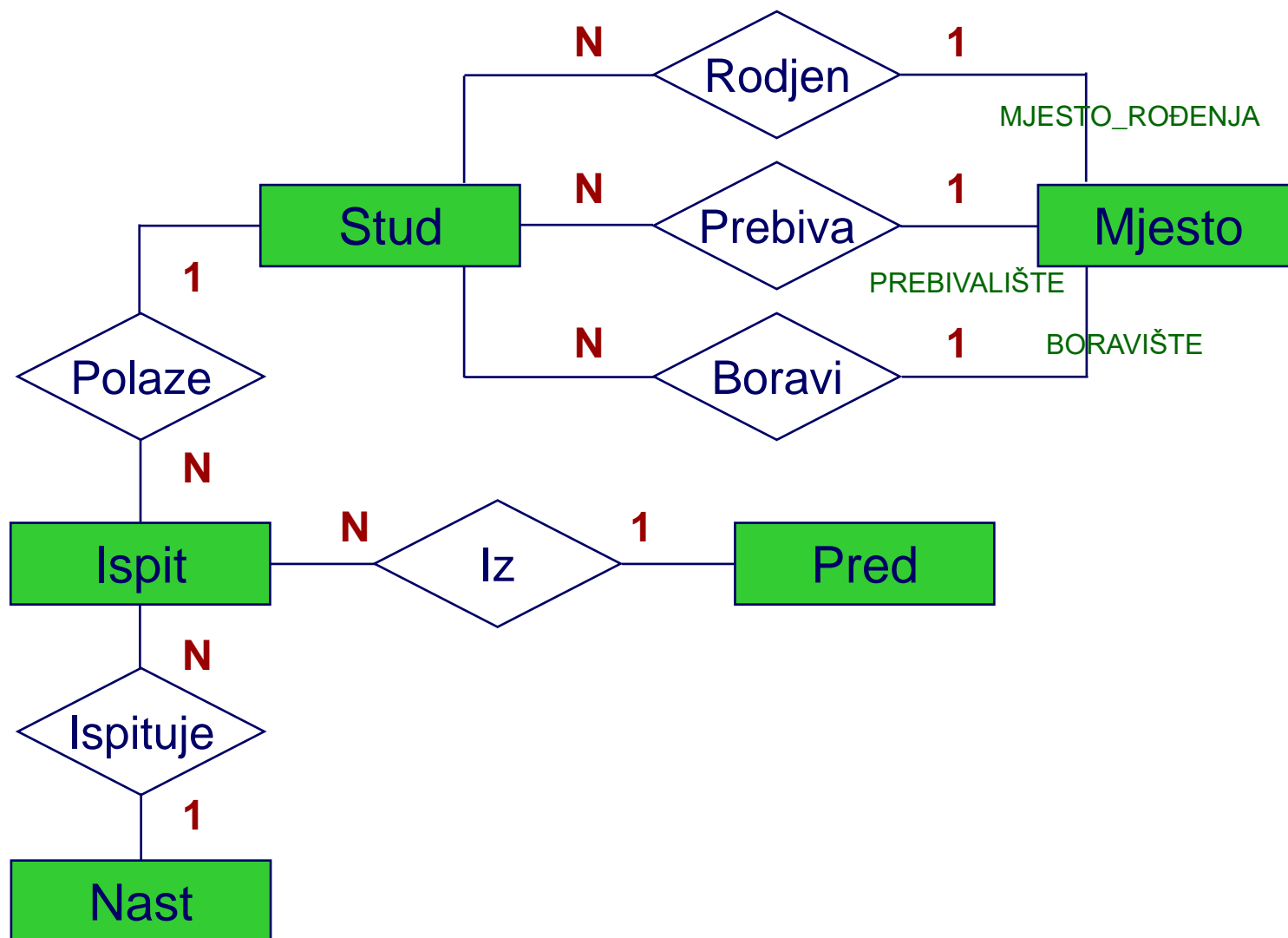
Stud = matBrStud, prezStud, imeStud, datRodStud

Mjesto = pbrMjesto, nazMjesto

PREBIVALIŠTE STUDENTA ???

BORAVIŠTE STUDENTA ???

Primjer (nastavak)



Primjer (nastavak)

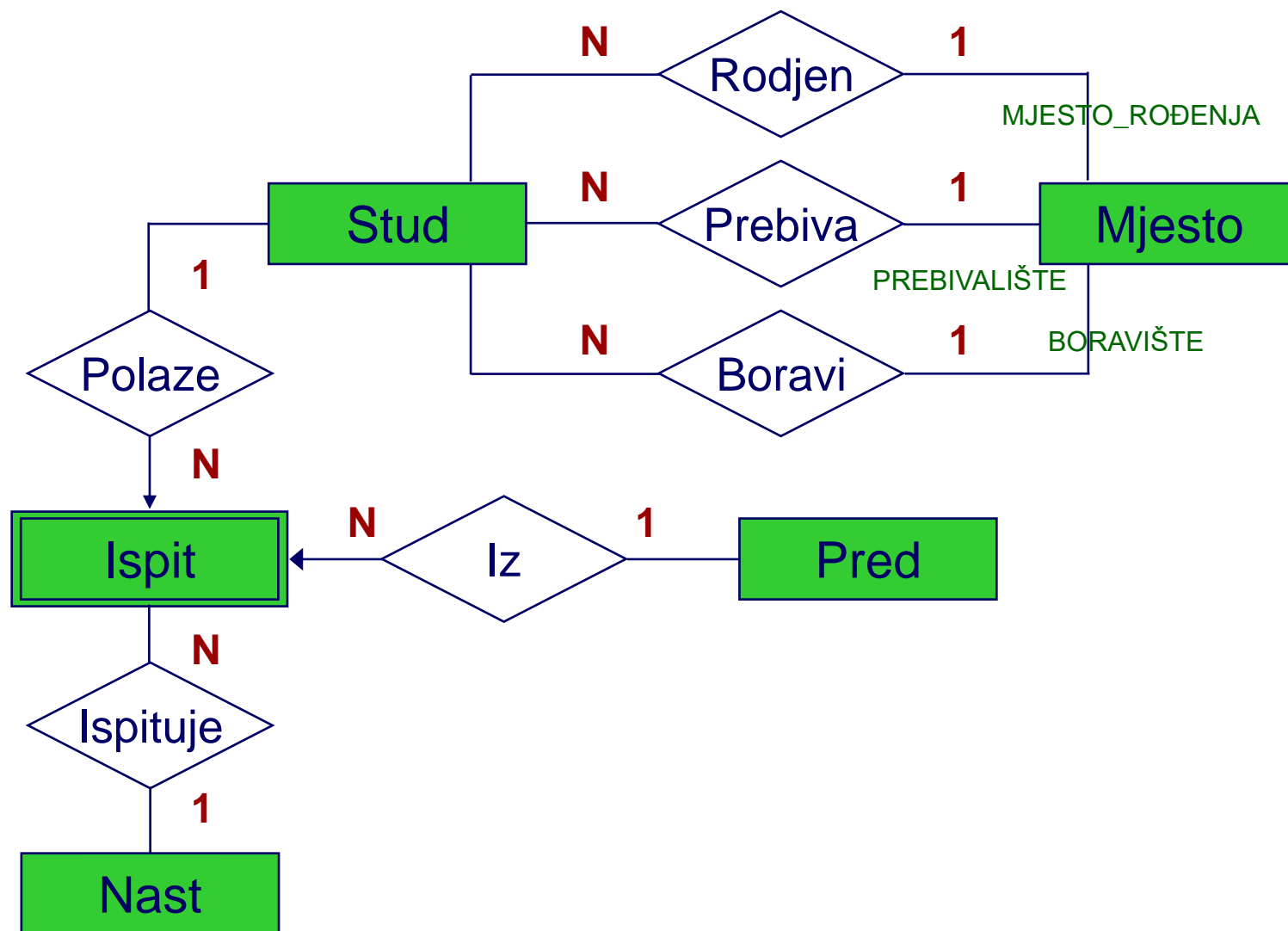
Stud = matBrStud, prezStud, imeStud, datRodStud, datUpisFERStud,
rangDMStud, eMailStud

Mjesto = pbrMjesto, nazMjesto

Ispit = matBrStud, sifraPred, datumIspit, ocjena

SLABI ENTITET !!!!

Primjer (nastavak)



Primjer (nastavak)

Stud = matBrStud, prezStud, imeStud, datRodStud, datUpisFERStud,
rangDMStud, eMailStud

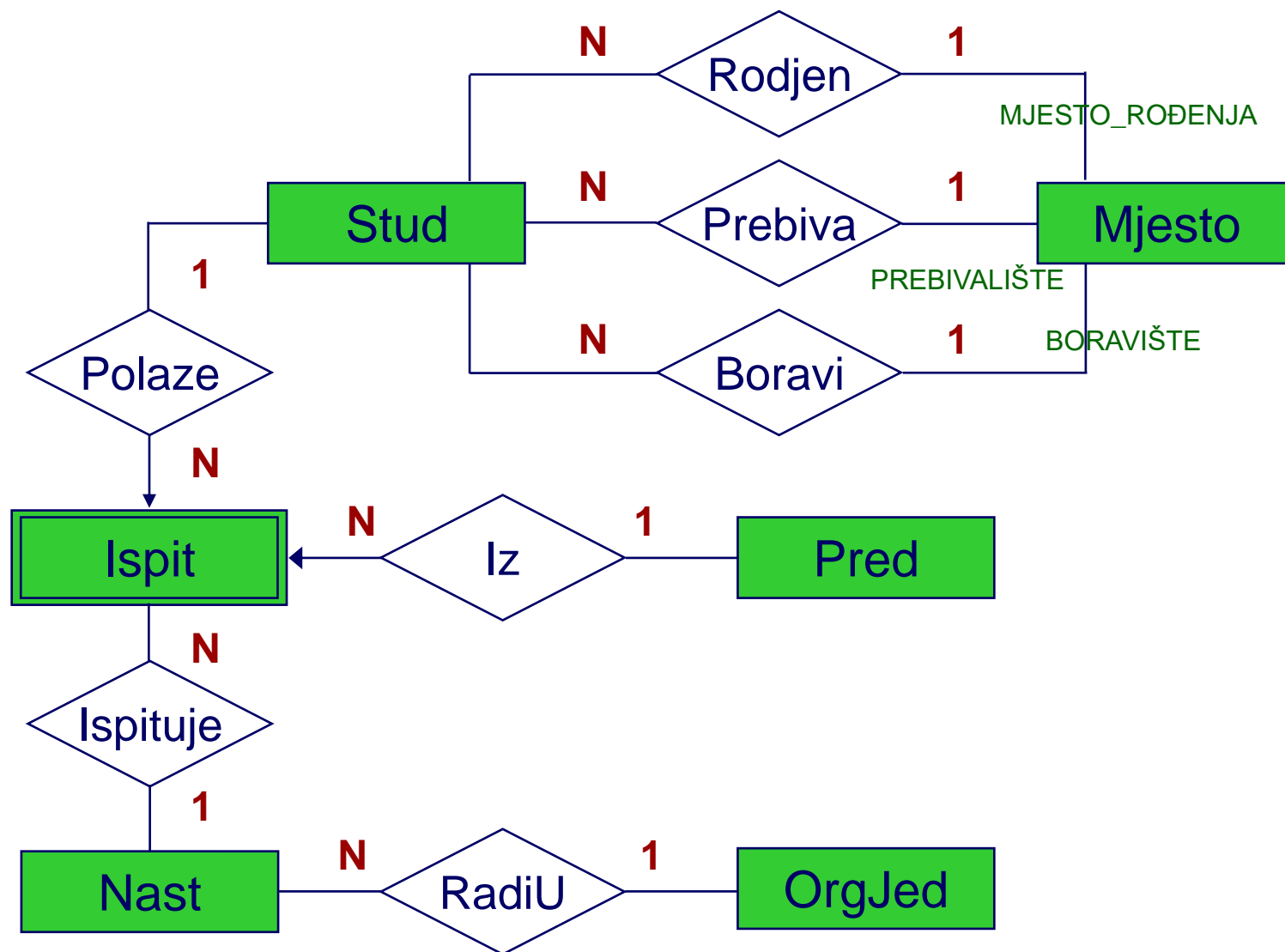
Mjesto = pbrMjesto, nazMjesto

Ispit = matBrStud, sifraPred, datumIspit, ocjena

Nast = sifraNast, prezNast, imeNast

ORGANIZACIJSKA JEDINICA ???

Primjer (nastavak)



Primjer (nastavak)

Stud = matBrStud, prezStud, imeStud, datRodStud, datUpisFERStud,
rangDMStud, eMailStud

Mjesto = pbrMjesto, nazMjesto

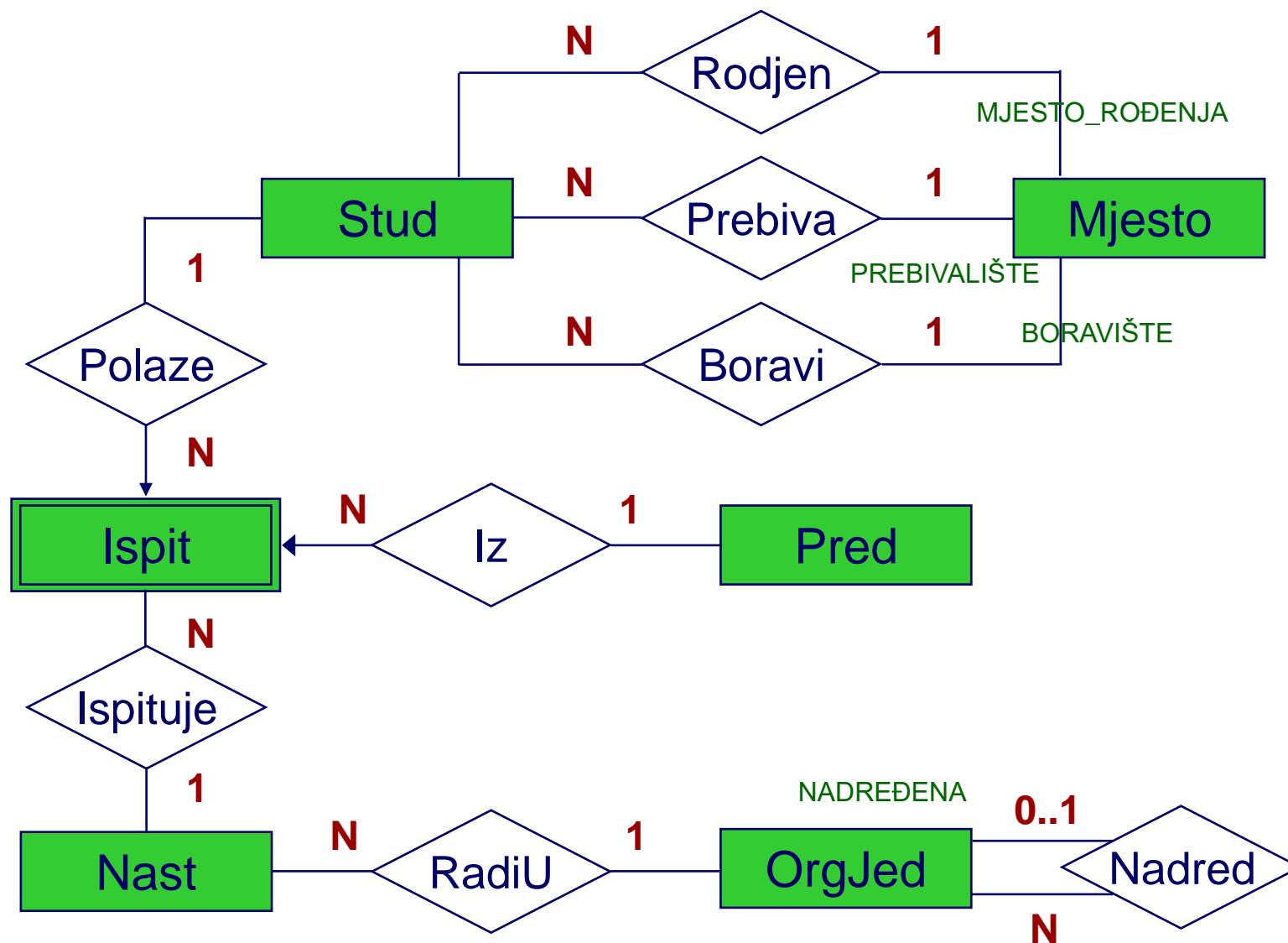
Ispit = matBrStud, sifraPred, datumIspit, ocjena

Nast = sifraNast, prezNast, imeNast

OrgJed = sifraOrgJed, nazivOrgJed

NADREĐENA ORGANIZACIJSKA JEDINICA ???

Primjer (nastavak)



Primjer (nastavak)

Stud = matBrStud, prezStud, imeStud, datRodStud, datUpisFERStud,
rangDMStud, eMailStud

Mjesto = pbrMjesto, nazMjesto

Ispit = matBrStud, sifraPred, datumIspit, ocjena

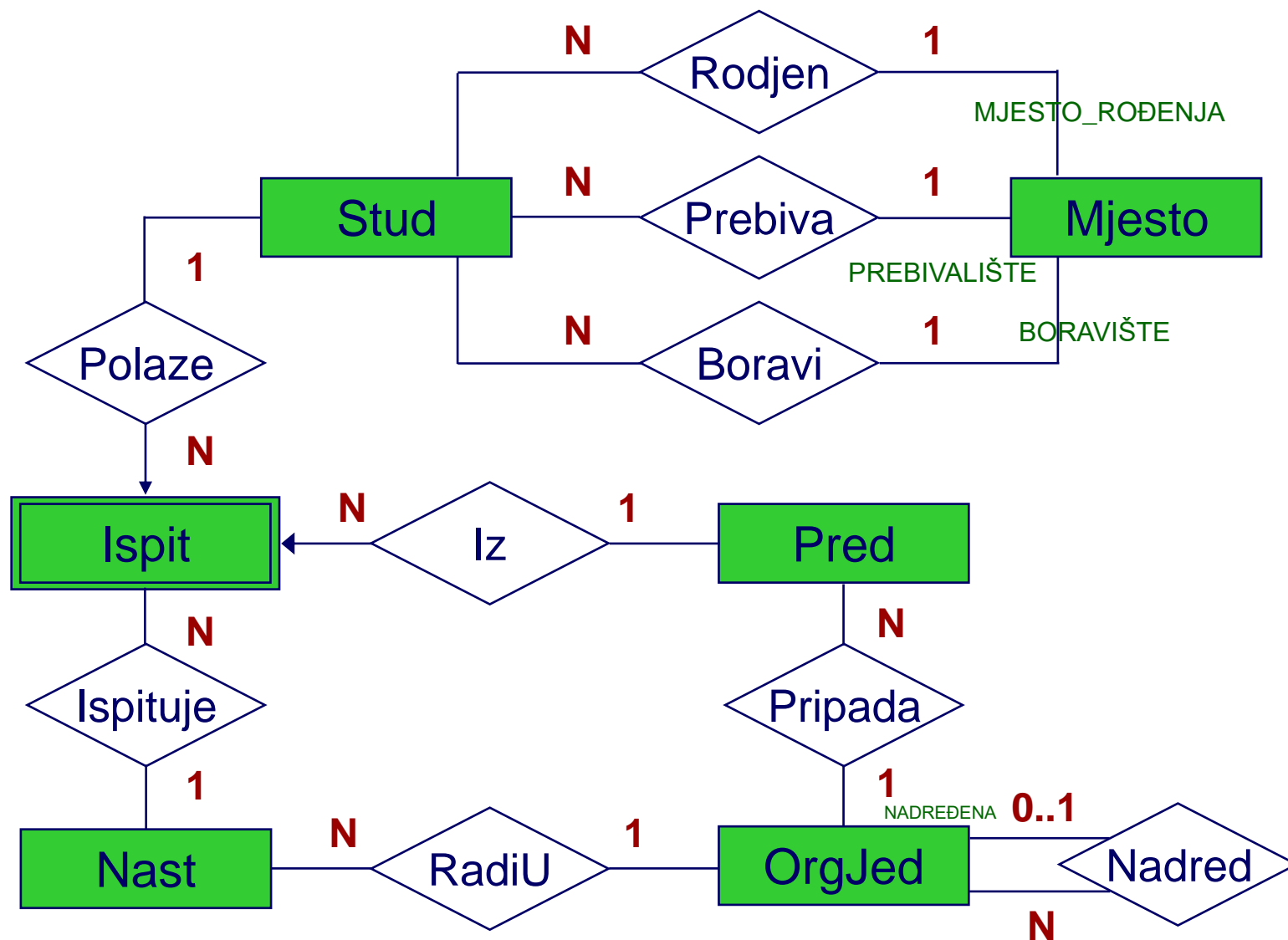
Nast = sifraNast, prezNast, imeNast, eMailNast, URLNast

OrgJed = sifraOrgJed, nazivOrgJed

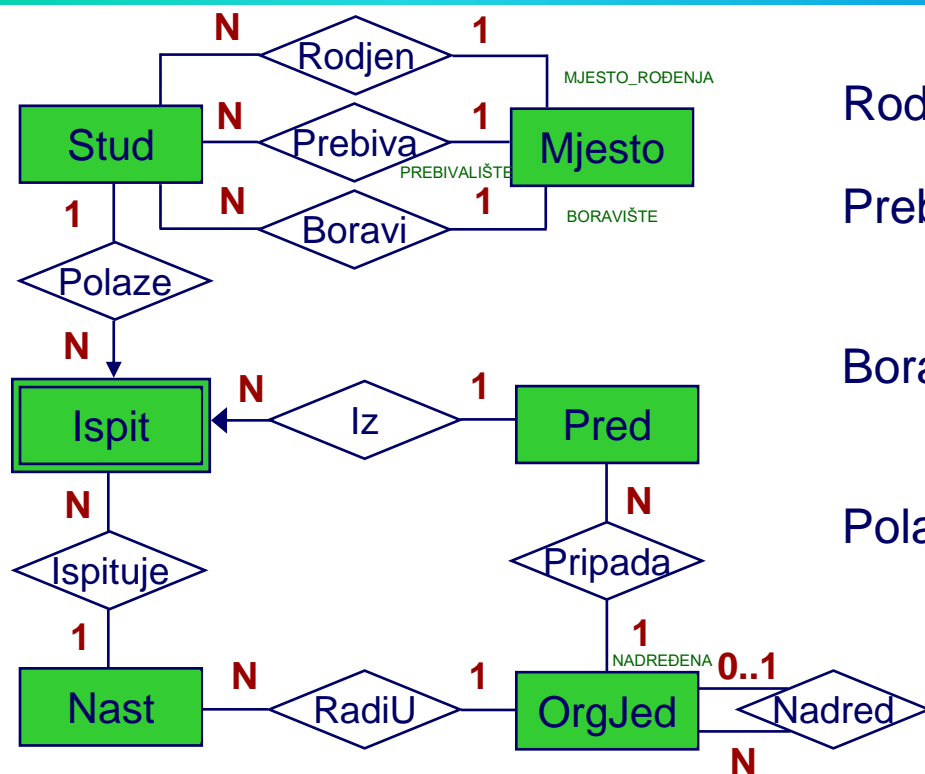
Pred = sifraPred, kraticaPred, nazivPred, URLPred

PREDMET PRIPADA ORGANIZACIJSKOJ JEDINICI ???

Primjer (nastavak)



Primjer (nastavak) - Opis veza



Rodjen = matBrStud, postBrMjRodStud

Prebiva = matBrStud, postBrMjPrebStud,
adresaMjPrebStud

Boravi = matBrStud, postBrMjBorStud,
adresaMjBorStud

Polaze = matBrStud, sifraPred, datumIspit

Iz = matBrStud, sifraPred, datumIspit

Ispituje = matBrStud, sifraPred, datumIspit, sifraNast

RadiU = sifraNast, sifraOrgJed

Pripada = sifraPred, sifraOrgJed

Nadred = sifraOrgJed, sifraNadOrgJed

→ Relacijski model

Stud = matBrStud, prezStud, imeStud, datRodStud, datUpisFERStud, rangDMStud, eMailStud

Mjesto = pbrMjesto, nazMjesto

Ispit = matBrStud, sifraPred, datumIspit, ocjena

Nast = sifraNast, prezNast, imeNast, eMailNast, URLNast

OrgJed = sifraOrgJed, nazivOrgJed

Pred = sifraPred, kraticaPred, nazivPred, URLPred

Rodjen = matBrStud, postBrMjRodStud

Prebiva = matBrStud, postBrMjPrebStud, adresaMjPrebStud

Boravi = matBrStud, postBrMjBorStud, adresaMjBorStud

Polaze = matBrStud, sifraPred, datumIspit

Iz = matBrStud, sifraPred, datumIspit

Ispituje = matBrStud, sifraPred, datumIspit ,sifraNast

RadiU = sifraNast ,sifraOrgJed

Pripada = sifraPred ,sifraOrgJed

Nadred = sifraOrgJed ,sifraNadOrgJed

→ Relacijski model

Unija shema s jednakim ključevima

Stud = matBrStud, prezStud, imeStud, datRodStud, datUpisFERStud,
rangDMStud, eMailStud, postBrMjRodStud, **postBrMjPrebStud**,
adresaMjPrebStud, postBrMjBorStud, **adresaMjBorStud**

Mjesto = pbrMjesto, nazMjesto

Ispit = matBrStud, sifraPred, datumIspit, ocjena, **sifraNast**

Nast = sifraNast, prezNast, imeNast, eMailNast, URLNast, **sifraOrgJed**

OrgJed = sifraOrgJed, nazivOrgJed, **sifraNadOrgJed**

Pred = sifraPred, kraticaPred, nazivPred, URLPred, **sifraOrgJed**