



Principi softverskog inženjerstva

Vežbe - IV nedelja Modeli baze podataka

Dražen Drašković, asistent
Elektrotehnički fakultet
Univerziteta u Beogradu



Potrebno je da:

- Razumete koncepte modela i njegovu svrhu
- Naučite kako se odnosi između definišu i kako su odnosi uključeni u proces dizajniranja baze podataka
- Naučite kako ERD komponente utiču na dizajn baze podataka i implementaciju
- Naučite simbole za modelovanje

Osnovni koncepti modelovanja

- **Model je reprezentacija stvarnosti.**
 - većina modela informacionih sistema predstavlja se grafičkom reprezentacijom;
- Modeli mogu biti izgrađeni za postojeće sisteme kao način da se bolje razumeju sistemi.
- Ili za predloženi sistem kao način da se zahtevi dokumentuju.

Konceptualno modelovanje podataka

- **Konceptualni model podataka** teži da otkrije i analizira podatke korisničkih zahteva i organizaciju tih podataka:
 - Koji podaci su važni?
 - Koje podatke treba održavati?
- Glavna aktivnost u ovoj fazi je identifikovanje entiteta, atributa i njihovih odnosa, kako bi se konstruisao ER dijagram (Entity Relationship Diagram).

Zašto model baze?

- Predstavlja “realnost” postojeće baze podataka
- ***** Dokumentacija *****
- Alat za efikasnu komunikaciju
- Uključivanje korisnika u sistem
- Predstavlja apstrakciju zahteva
- Određuje poslovna pravila koja čuvamo u bazi podataka
- Nezavisnost od određenog DBMS

ER model

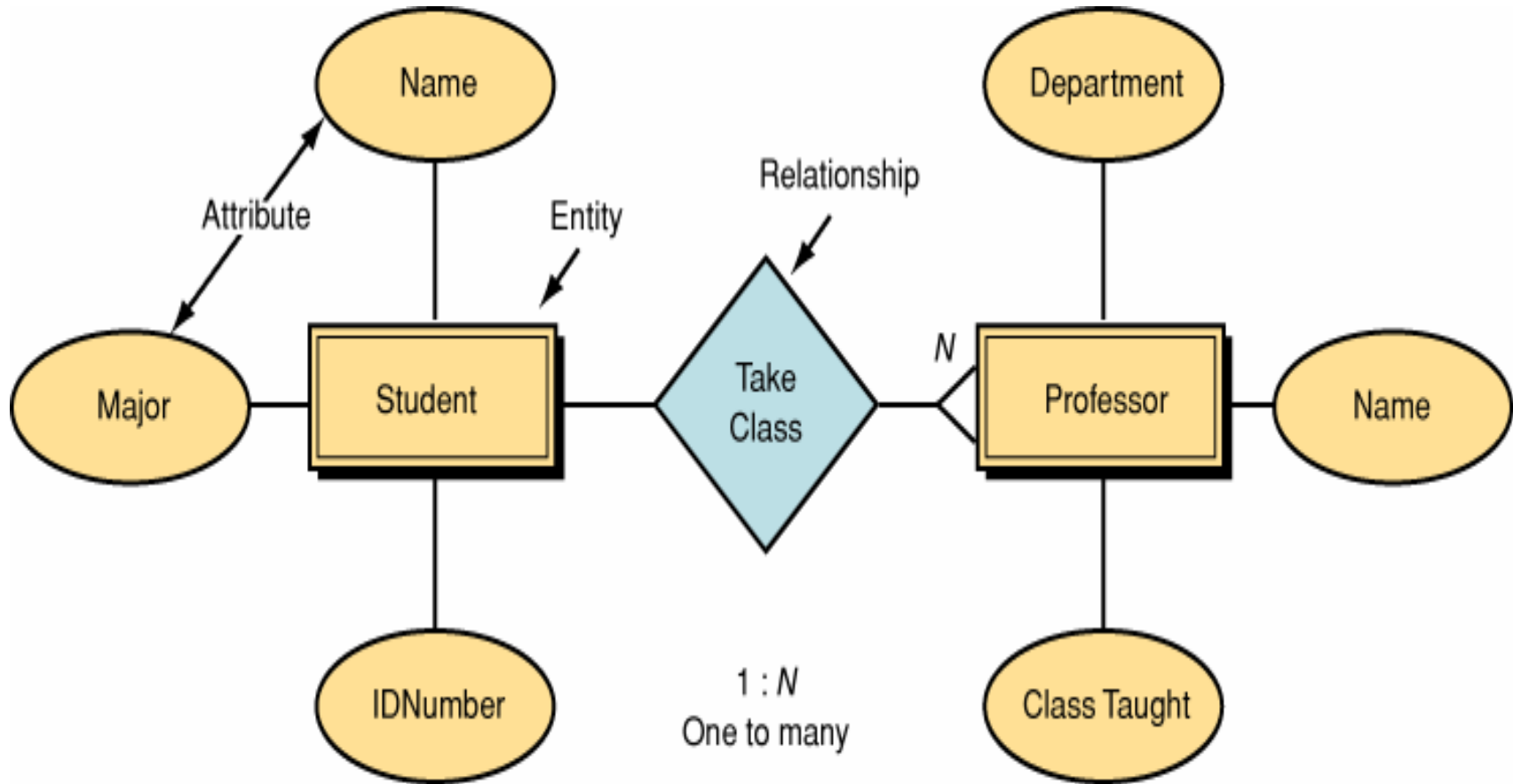


- **Entity-relationship model (ERM)**
- Modelovanje entiteta i odnosa je metoda koja se koristi za prikazivanje konceptualne šeme relacione baze podataka.
- Dijagrami nastali ovim procesom nazivaju se ER dijagrami.
- Koriste se u prvoj fazi projektovanja informacionih sistema u toku analize zahteva da opiše potrebne informacije ili tip informacija koje se čuvaju u bazi podataka.

Osnovni elementi ER dijagrama

- Entitet: skup ljudi, mesta, objekata, događaja, koncepata od interesa (tabela)
 - Instanca entiteta - član entiteta: osoba, mesto, objekat,... (red u tabeli)
- Atribut: karakteristika entiteta (polje u tabeli)
- Odnos/Veza: asocijacija entiteta (odgovara odnosu primarni ključ-strani ključ u vezi tabela)

ER diagram - Chenova notacija



ER dijagram - Notacija



- Entitet
 - **pravougaonik** koji sadrži ime entiteta.
- Atribut
 - **elipsa** koja sadrži ime atributa.
- Odnos
 - **romb** koji sadrži naziv odnosa.

Koraci u kreiranju baze/ER dijagrama

Proces dizajna baze podataka:

1. Odrediti svrhu baze
2. Pronaći i organizovati zahtevane informacije (identifikovati entitete, identifikovati attribute)
3. Podeliti informacije u tabele
4. Dodeliti stavke tabelama (instance entiteta)
5. Odrediti primarne ključeve
6. Identifikovati odnose (relacije) između tabela
7. Analizirati druge detalje dizajna baze podataka
8. Primeniti pravila normalizacije

ER dijagram - Notacija



Entity Name

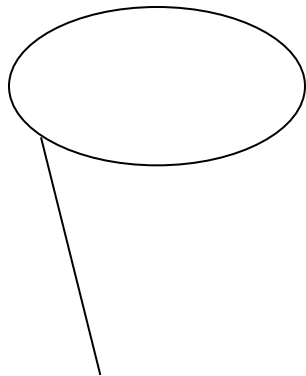
Jak entitet

Entity Name

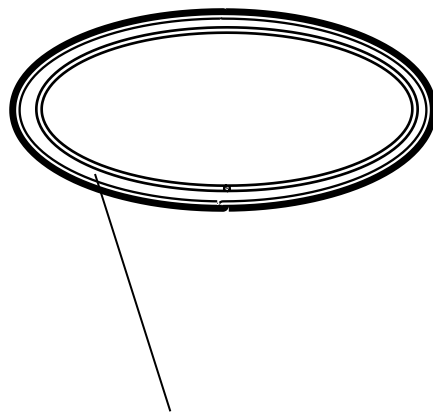
Slab entitet

ER dijagram - Notacija

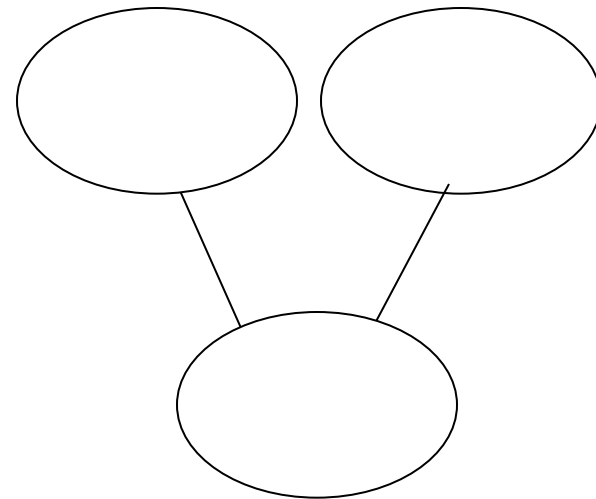
Atributi



atribut



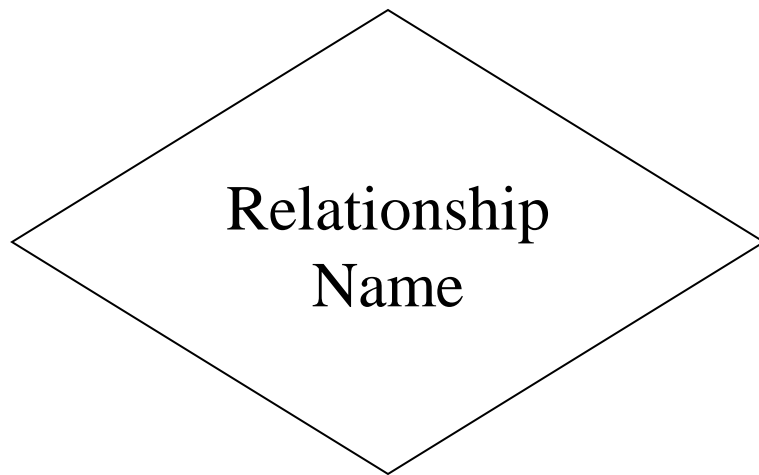
atribut sa više vrednosti



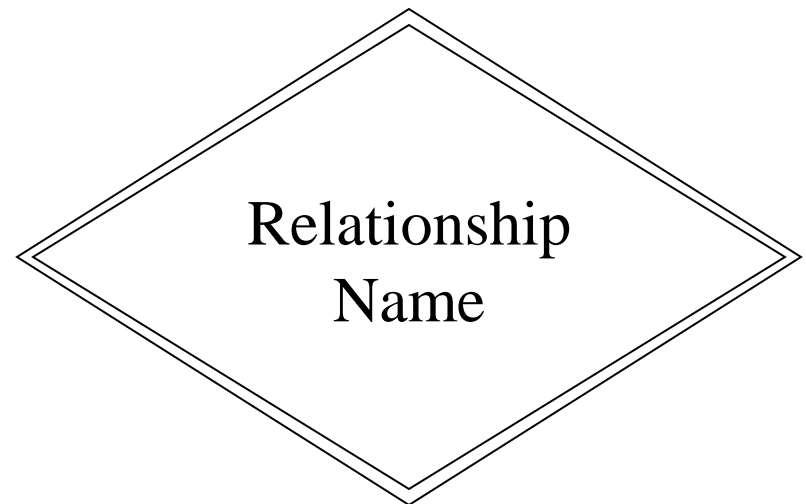
složeni atribut

ER dijagram - Notacija

Odnosi



veza

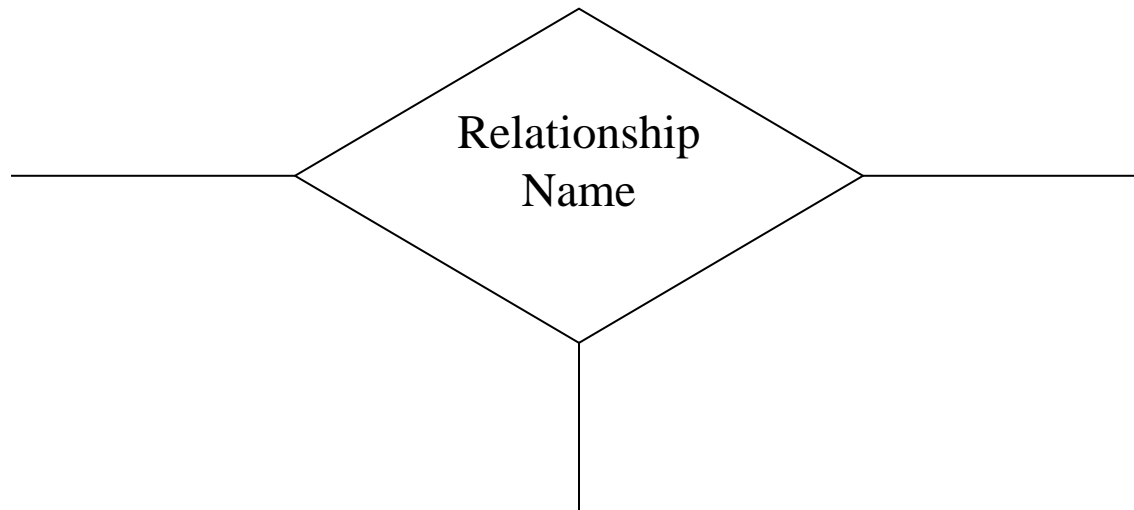


identifikaciona veza

ER dijagram - Notacija



- N-arna veza - učestvuje više od 2 entiteta



Entitet (1)

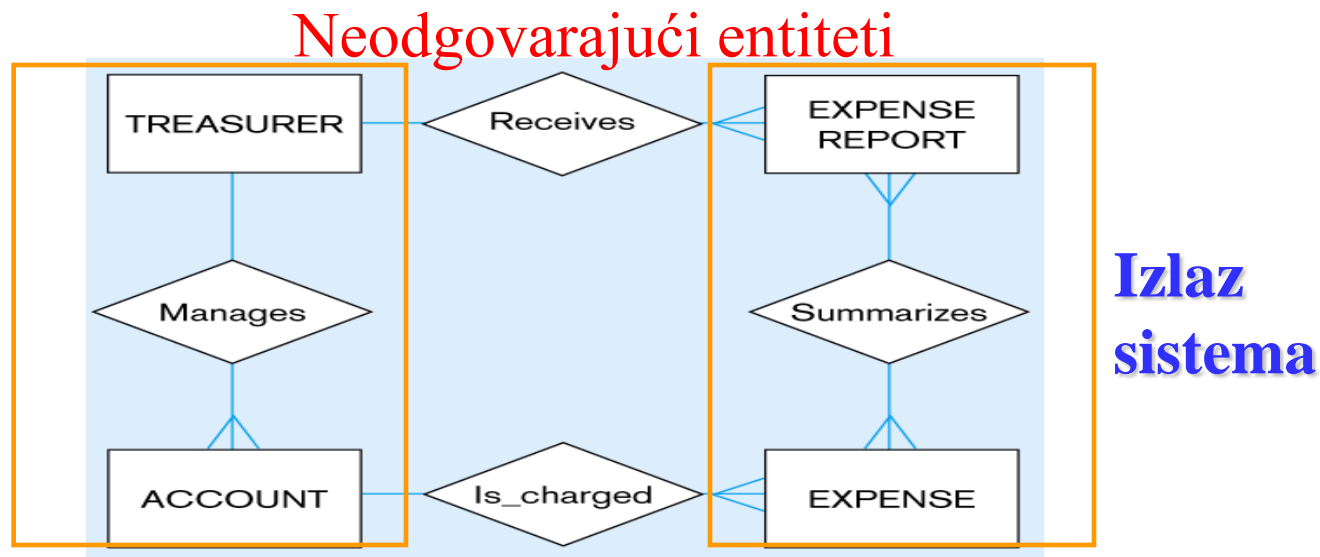


- **Fundamentalna stvar od značaja je koji podaci mogu da se čuvaju**
- Šta treba da bude entitet:
 - Objekat koji će imati mnogo instanci u bazi podataka
 - Objekat koji će biti sastavljen od više atributa
 - Objekat koji pokušavamo da modelujemo
- Šta ne treba da bude entitet:
 - Pojedinačni korisnik baze podataka
 - Izlaz baze podataka (npr. neki izveštaj)

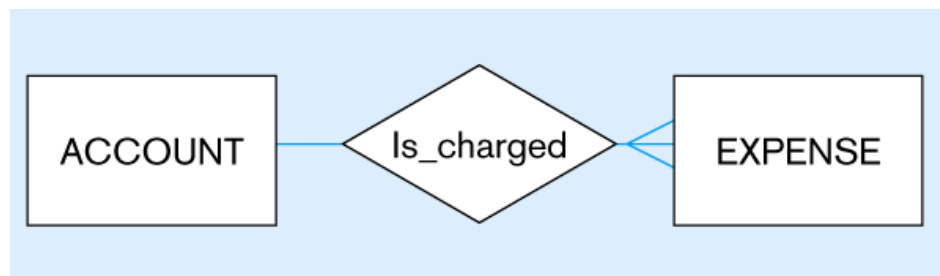
Entitet (2)



**Korisnik
sistema**



**Izlaz
sistema**



Odgovarajući entiteti

Instanca entiteta (1)

- Instanca entiteta: jedno pojavljivanje entiteta
- Predstavlja se jednim redom u tabeli

Entitet: Student

Student indeks	Prezime	Ime
20080006	Purić	Sofija
20080035	Lenhart	Jan
20080076	Čejović	Jovan
20080097	Dašić	Darija
20080102	Đurica	Aleksandra
20080104	Karlaš	Bojan

instanca,
primerak

Primer: 6 instanci

Instanca entiteta (2)



CUSTOMER
entity contains:
CustNumber
CustName
Address
City
State
Zip
ContactName
PhoneNumber

Two instances of CUSTOMER:

12345
Ajax Manufacturing
123 Elm St
Memphis
TN
32455
P. Schwartz
223-5567

67890
Jefferson Dance Club
345-10th Avenue
Boston
MA
01234
Frita Bellingsley
210-8896

Atributi



- Opisuje osobine/karakteristike entiteta
- Entitet: Zaposleni
- Atributi:
 - Ime i prezime
 - Adresa
 - MobilniTelefon
 - DatumZaposlenja
 - RadniStaz
 - Plata

Klase atributa



- Prosti atributi
- Složeni (kompozitni) atributi
- Izvedeni atributi
- Atributi sa jednom vrednošću
- Atributi sa više vrednosti

Prosti/složeni atributi

- Prosti atributi ne mogu biti dalje deljeni.
 - Primeri: Godine, Pol, Bračni status
- Složeni atributi mogu biti dalje deljeni na podatribute koji bliže određuju atribut.
 - Primeri:
Adresa → Ulica, Broj, Grad, PTT broj, Država
Telefon → PozivniPrefiks, BrojTelefona

Primer entiteta i atributa

- Entitet:

- Teniseri

- Atributi:

- ID#: "123" (atribut sa jednom vrednošću)
- Ime i prezime: "Novak Đoković" (složeni)
- Adresa: "Monte Carlo, Monaco" (složeni)
- Telefoni: "(44)1-123-4567, (381)11-234-5678" (atribut sa više vrednosti)
- Datum rođenja: "22-05-1987" (prost)
- Godine: 22 (izveden)

Izvedeni atributi

- Ne nalaze se fizički u bazi podataka.
- Obično se izvode pomoću nekog algoritma.
- Primer:
 - Atribut GODINE može biti dobijen ako se zna datum rođenja i trenutni datum
 - MS Access: `int((Date()-Osoba_Rodj)/365)`

Atributi sa više vrednosti

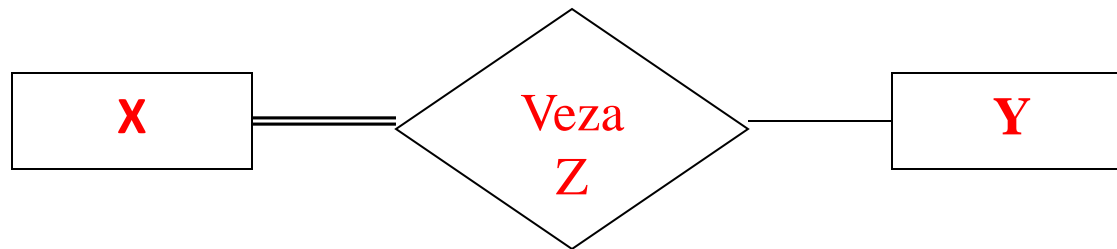


- Atribut koji može imati više vrednosti.
- Primeri:
 - Osoba može imati nekoliko fakultetskih diploma.
 - U stanu možete imati više telefonskih brojeva.
 - Auto može biti ofarban u više boja.

Ograničenja (1)

Totalno učešće

- Totalno učešće - entitet X ima totalno učešće u vezi Z, što znači da svaka instanca X učestvuje u NAJMANJE jednoj vezi.
- Primer: X je kupac, Y je proizvod, Z je kupovati. Slika ispod pokazuje zahtev da svaki kupac kupi proizvod.



Ograničenja (2)

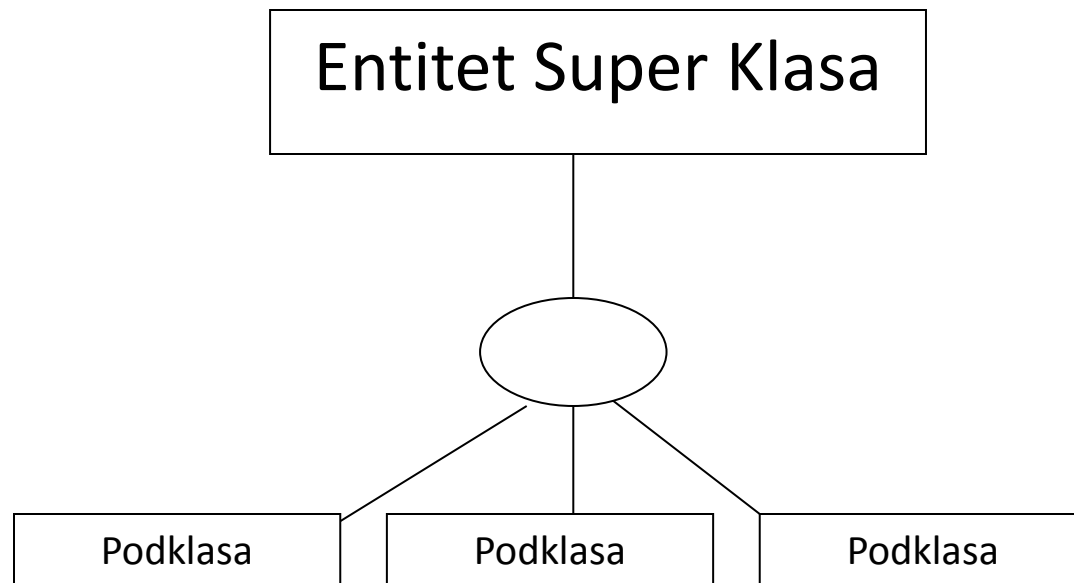
Parcijalno učešće

- Parcijalno učešće - entitet Y parcijalno učestvuje u vezi Z, što znači da neke instance Y učestvuju u vezi.
- Primer: X je kupac, Y je proizvod, Z je kupovati.
- Slika ispod pokazuje zahtev da svaki proizvod ne mora da bude kupljen od strane kupca. Neki proizvodi ne moraju biti kupljeni.

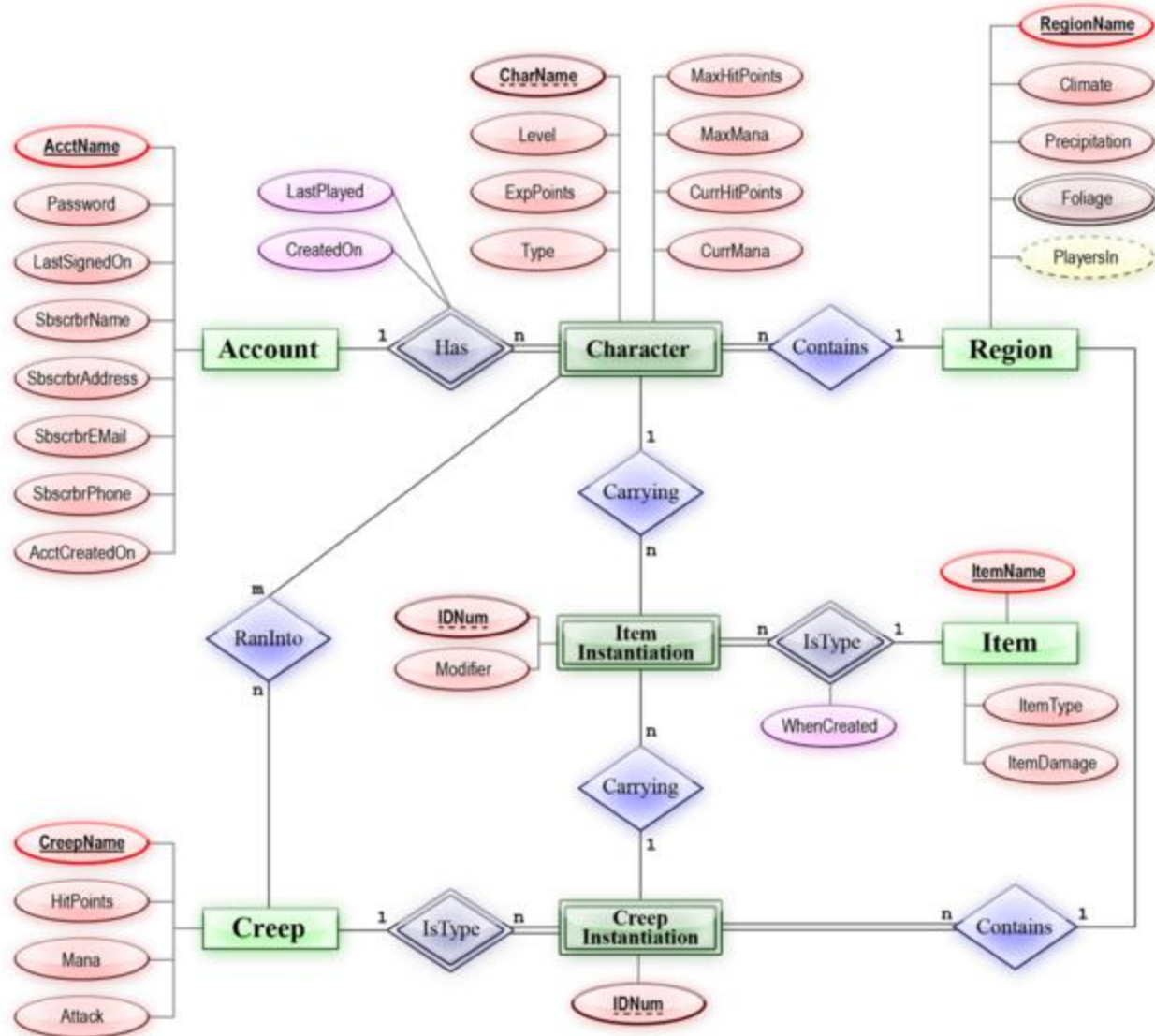


Specijalizacija / generalizacija

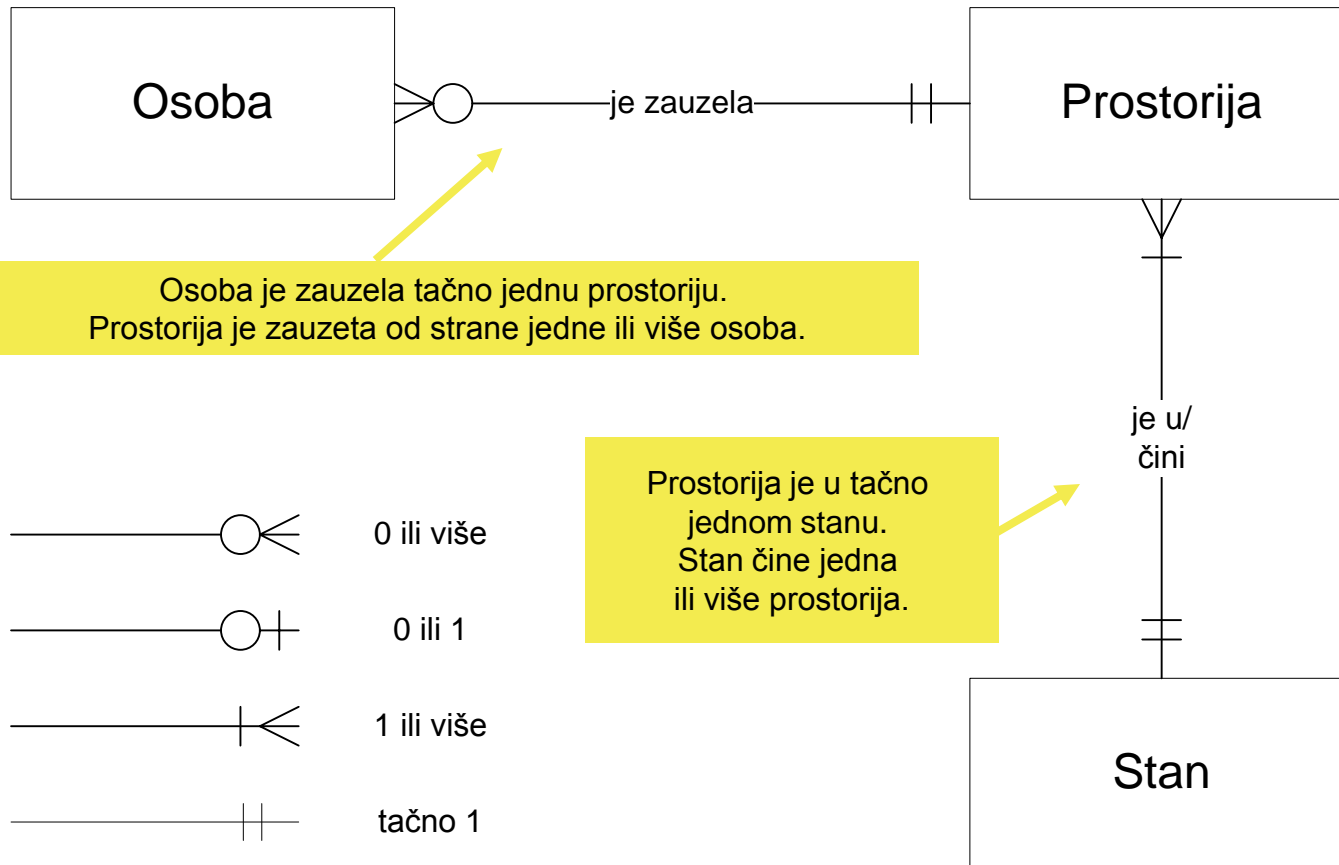
- Svaka podklasa nasleđuje sve odnose i attribute svoje super-klase.



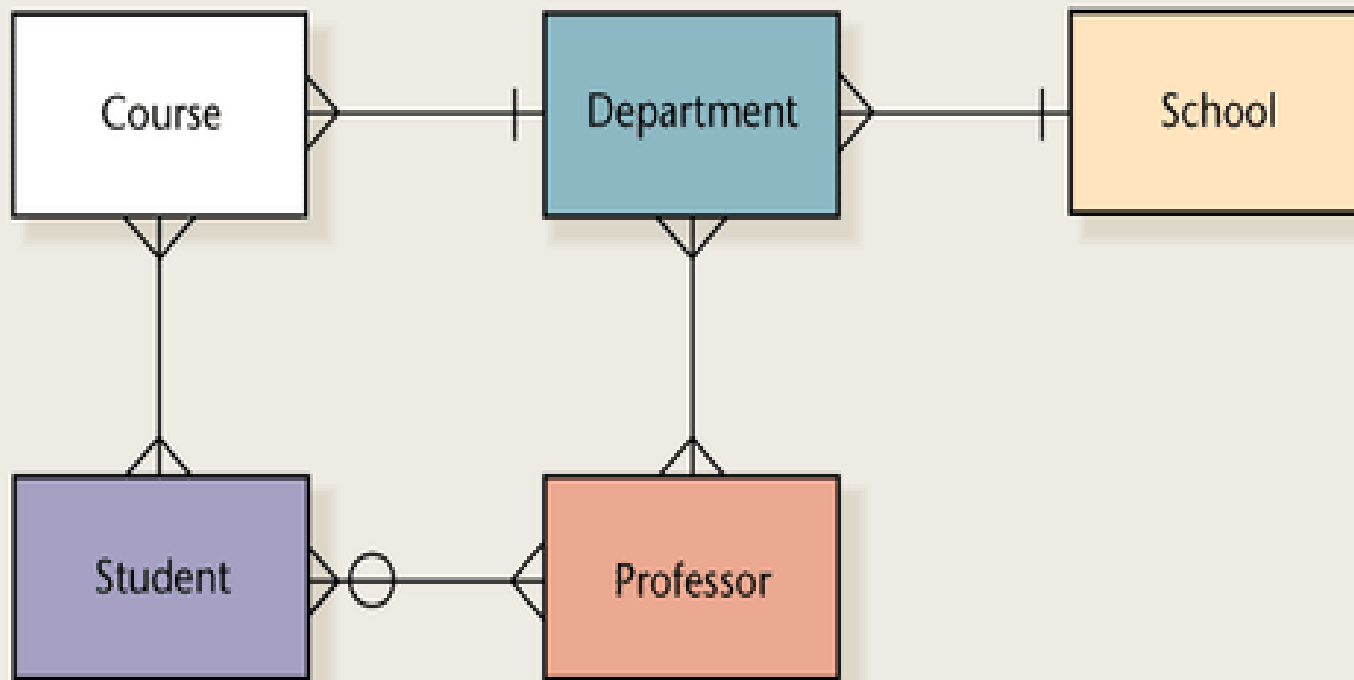
Primer ER dijagrama



IE (Information Engineering) notacija za ER modele



ER model koristeći IE notaciju



Kako pronaći entitete?



- Primer: ljudi, mesta, objekti, događaji,...
- Realni atributi: kupac, proizvod,...
- Potražite **imenice**, ALI vlastita imenica nije dobar kandidat za entitet...

Kako pronaći attribute?



- Atribut:
 - opis čije su vrednosti povezane sa pojedinačnim entitetom određenog tipa entiteta
- Potražite opis/karakteristiku entiteta

Identifikator

- **Atribut koji jedinstveno određuje instancu entiteta**
- Jedinstveno se identifikuje svaka instanca entiteta (primarni ključ)
- Jedan ili više atributa entiteta
 - **Složeni identifikator** (ključ) je identifikator koji čine 2 ili više atributa.
- Atributi se predstavljaju podvučenom linijom ispod imena atributa:
 - Primer: Zaposleni (IDzap), Student (Indeks)

Praktičan primer - FIS ETF



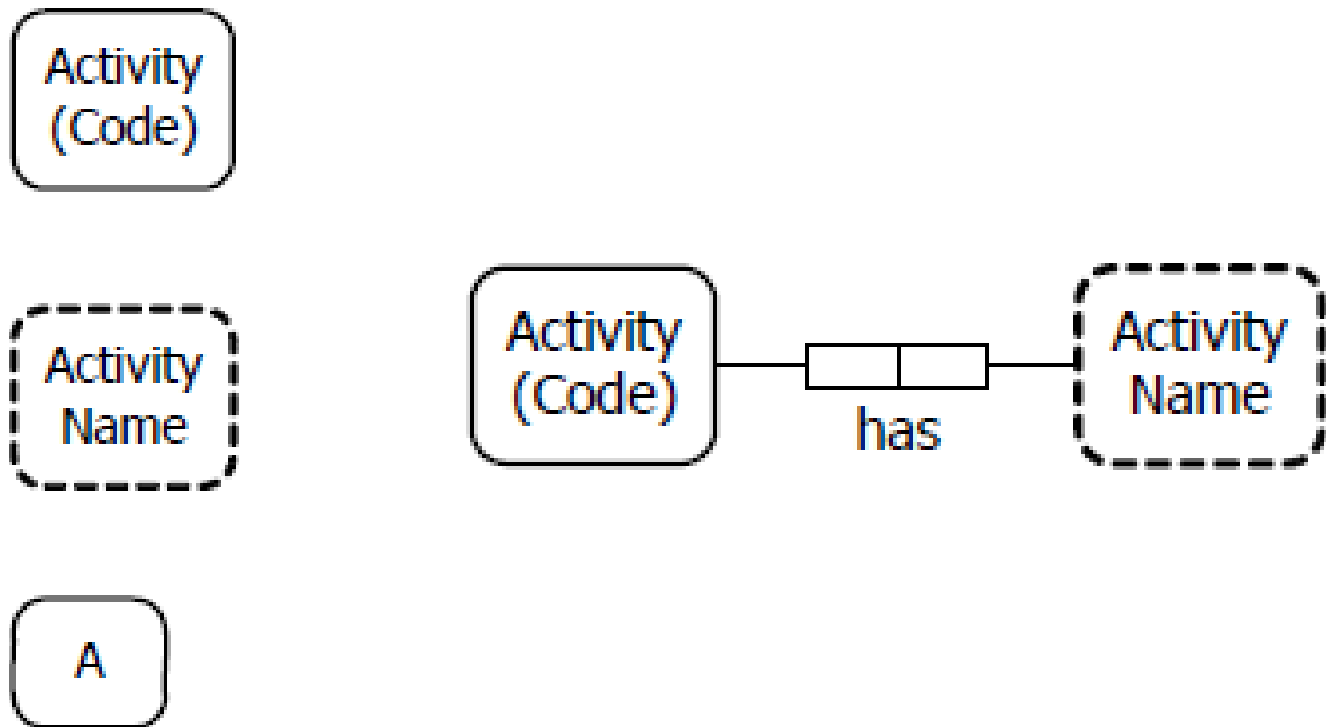
- IE notacija Fakultetskog informacionog sistema na ETF-u (deo modela prikazan u PDF-u)
- Model baze se menja na dnevnom nivou
- Trenutno ima 1361 objekat (tables, views) i 567 procedura

Konceptualni modeli - ORM2

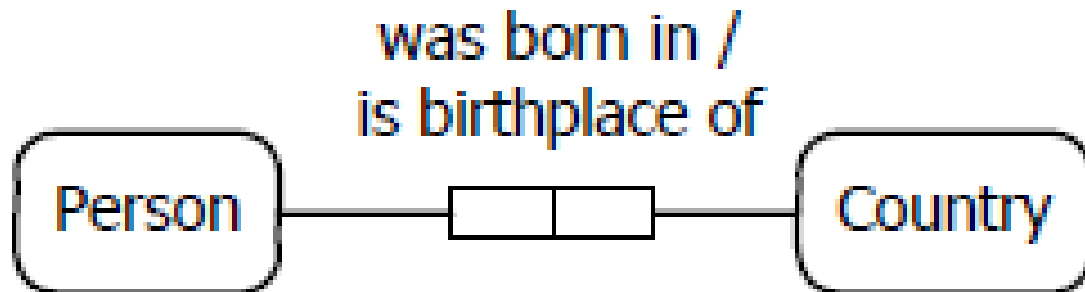
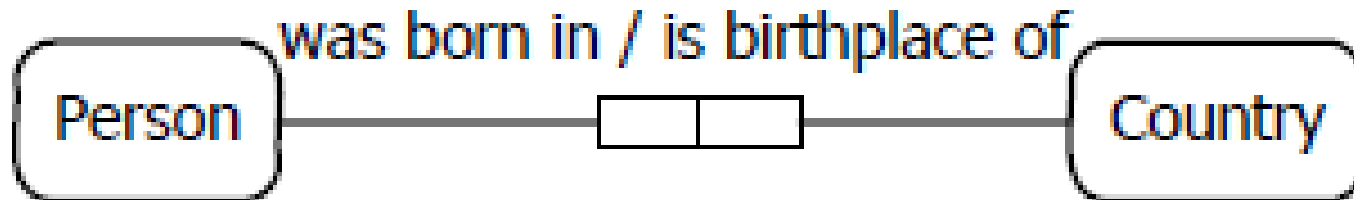
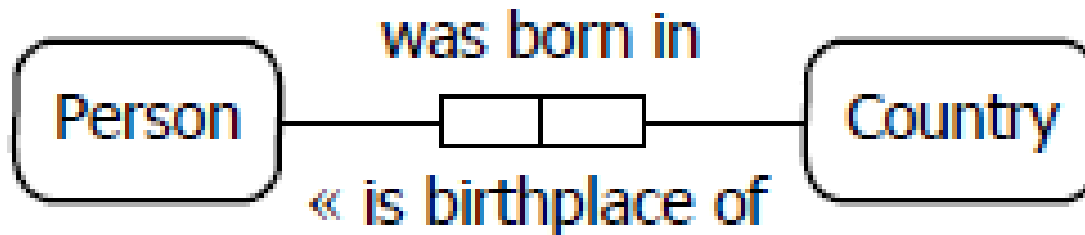
- **Object Role Modeling**
- Modelovanje semantike jednog univerzuma korišćenjem prirodnih jezika i dijagrama.
- Primer:
Zaposleni X je angažovan od DATUM
- Za razliku od ER i UML modela, ORM tretira sve elementarne činjenice kao odnose.

ORM2

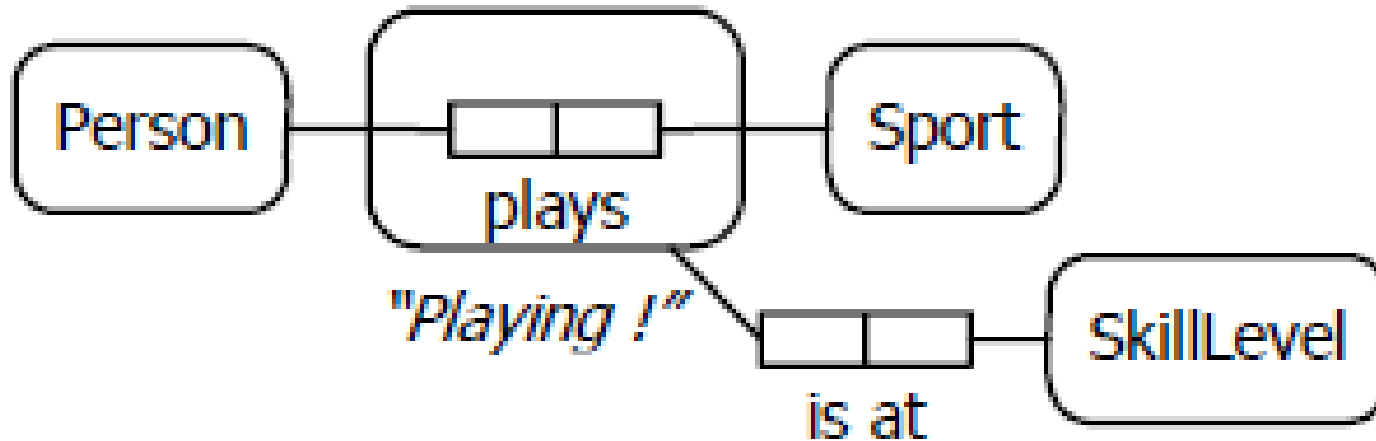
- Poslednja generacija Object-Role Modeling
- Primer aktivnosti



ORM2 - čitanje predikata



ORM2 - ugneždavanje



- Drugi primeri:

http://www.orm.net/pdf/ORM2_TechReport1.pdf

Unified Modeling Language (UML)

- Modelovanje zasnovano na principima OOP
- UML definiše standardni set dijagrama za sve faze razvoja softverskog sistema
- Programeri koriste određene tipove dijagrama:
 - Biznis modeli - dijagrami slučajeva korišćenja (use-case), dijagrami aktivnosti, dijagrami sekvenci
 - Logički modeli podataka - dijagram klasa
 - Fizički modeli podataka - dijagrami modela podataka
- Više o modelovanju veb aplikacija naknadno 😊

Vaš zadatak

Šta uraditi?

- Potrebno je napraviti:
 - logički model podataka u IE notaciji
 - dokument Specifikacija baze podataka
- Preporučeni alati:
 - MySQL Workbench
 - Erwin Data Modeler
 - Toad Data Modeler
 - MS Visio

Deadline ?

- Sve modele i dokument je potrebno poslati u SVN/GIT repozitorijum
- Priložiti i tekstualni fajl sa spiskom alata koje ste koristili za pravljenje modela
- Krajnji rok za ovu fazu biće naknadno određen.