

Tehnologija upravljanja podatkov

**Univerzitetni študij RI in UI, 3. letnik
Magistrski študij RI**

Matjaž Kukar, 2021/22





Splošne informacije...

- Predavanja

- Izr. prof. dr. Matjaž Kukar, ponedeljek 11:15 (???),
matjaz.kukar@fri.uni-lj.si
- Govorilne ure:
 - ???, kabinet 2.04 (2. nadstropje, desno od dvigala)

- Vaje:

- Doc. dr. Luka Šajn

- Spletna stran:

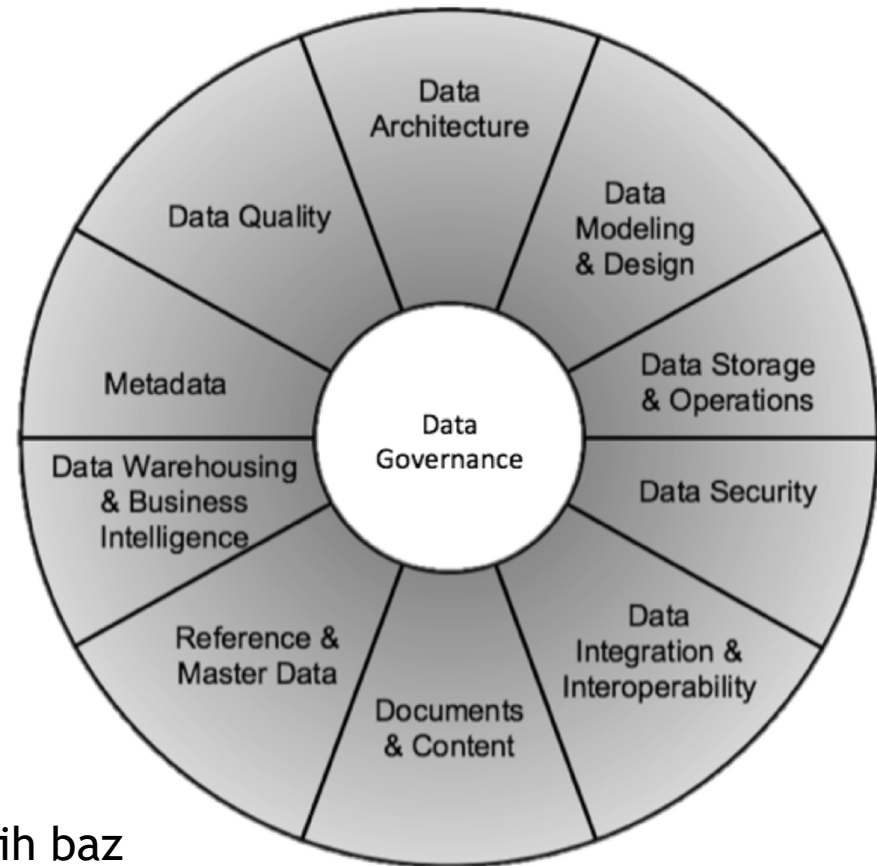
- Tehnologija upravljanja podatkov
<https://ucilnica.fri.uni-lj.si>



Upravljanje s podatki (*data management*)

- Definicija (Data Management Association – DAMA)
Upravljanje s podatki sestavljajo razvoj in izvajanje arhitektur, usmeritev in praktičnih postopkov za podporo celotnemu življenjskemu ciklu podatkovnih potreb sodobnega podjetja.

V kontekstu informacijskih tehnologij:
Tehnologija upravljanja podatkov =
napredne teme s širšega področja podatkovnih baz



Pričakovano predznanje

- Nadgradnja predmeta "Osnove podatkovnih baz"
 - Opisovanje in shranjevanje podatkov v PB
 - Zgradba SUPB, upravljanje z diskom in pomnilnikom
 - Organizacija in indeksiranje datotek
 - Poizvedovanje v PB
 - Relacijski podatkovni model, algebra in račun; SQL
 - Načrtovanje PB
 - Pristopi k načrtovanju PB
 - Konceptualno, logično in fizično modeliranje ???
- Nadgradnja programerskih predmetov
 - Osnovno znanje programiranja in uporabe orodij
- Razvoj informacijskih sistemov
 - Presek nekaterih tem (predvsem načrtovanje), poudarek na PB

Predpriprava na predavanja in vaje

- Primeri na predavanjih-vajah v Pythonu. Ga obvladate?
 - Uradna Python dokumentacija, <http://docs.python.org>
 - J. Demšar: *Python za programerje*. FRI, Ljubljana, 2009.
 - Learn Python the hard way, <http://learnpythonthehardway.org/book/>
 - Learn Python (interaktivno), <http://www.learnpython.org/>
- Aktualna verzija (oktober 2021):
 - Python 3.9; 3.7 ali novejši je OK (**priporočena distribucija Anaconda**)
- Zakaj Python?
 - manjši obseg pisanja (večina primerov na enem zaslonu)
 - hitrejša programiranje
 - uporaba v podatkovni znanosti



Vsebina predavanj

- Eksterni vidiki obvladovanja podatkov:
 - Načrtovanje podatkovnih baz
 - Konceptualno, logično in fizično načrtovanje
 - Normalizacija relacij
 - Analiza uporabniških zahtev
 - Podatkovne baze in podatkovna skladišča
 - Namen in načrtovanje podatkovnih skladišč
 - Zagotavljanje kvalitete shranjenih podatkov
 - Analiza shranjenih podatkov (OLAP, Data Mining)
 - Sodobne nerelacijske podatkovna baze (NoSQL)
 - Dokumentne: MongoDB, grafne: Neo4j, stolpične: MonadDB
 - ... vaše izkušnje?



Vsebina predavanj

- Interni vidiki obvladovanja podatkov:
 - Dostop do podatkov
 - Zagotavljanje dostopnosti in konsistentnosti podatkov (upravljanje s transakcijami)
 - Upravljanje sočasnosti dostopa do podatkovne baze
 - Varovanje in obnavljanje podatkovne baze
 - Relacijska algebra in SQL (ponovitev)
 - Optimizacija in evalvacija poizvedb
 - Načrtovanje izvajanja poizvedb
 - Vrednotenje zahtevnosti osnovnih operacij
 - Alternativne strategije izvajanja poizvedb
 - Upravljanje delno strukturiranih in nestrukturiranih podatkov
 - JSON, XML, prostorski podatki, tekst, zvok, slika ...

Izhodiščna literatura

1. Relacijski SUPB

- a) Thomas M. Connolly, Carolyn E. Begg (2015). Database Systems, A Practical Approach to Design, Implementation and Management, 6th Edition, Pearson
- b) Ramez Elmasri, Shamkant B. Navathe (2016). Fundamentals of Database Systems, 7th Edition, Pearson
- c) S. Sumathi, S. Esakkirajan: Fundamentals of Relational Database Management Systems, Springer, 2007
- d) Carlos Coronel, Steven Morris (2018). Database Systems: Design, Implementation and Management, 13th Edition, Cengage Learning
- e) Raghu Ramakrishnan, Johannes Gehrke (2003). Database Management Systems, Third Edition, McGraw-Hill
- f) Paul Wilton and John W. Colby (2005): Beginning SQL, Wrox

Izhodiščna literatura

2. Nerelacijski (NoSQL) SUPB

- a) Ian Robinson, Jim Webber and Emil Eifrem: Graph Databases, O'Riley (free download), <https://neo4j.com/graph-databases-book>
- b) Dan McCreary and Ann Kelly: Making Sense of NoSQL: A guide for managers and the rest of us, Manning Publications, 2013
- c) Pramod J. Sadalage and Martin Fowler: NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence , Addison-Wesley, 2012
- d) Alex Giamas: Mastering MongoDB 3.x: An expert's guide to building fault-tolerant MongoDB applications, Apress, 2017

3. Spletni viri (sproti)



Vsebina vaj

I. Praktična obravnava nekaterih tem s predavanj

- Spoznavanje s programsko opremo
- Načrtovanje podatkovne baze in podatkovnega skladišča
- Programski dostop do PB
- Normalizacija, priprave na normalizacijo, denormalizacija
- Transakcije
- Optimizacija poizvedb
- Nerelacijski SUPB

II. Domače naloge (vsaj 50%)

III. Seminarska naloga v obliki projekta (vsaj 50%)

- Na koncu semestra (december, januar)
- Nekoliko obsežnejša kot domače naloge
- Obvezen predstavitveni seminar (vaje ali predavanja)
- Pravočasnost izdelave, oddaje in zagovora!



Orodja (minimalne verzije)

- PowerDesigner 12.5
 - MariaDB ≥ 10.3 (ali MySQL ≥ 8)
 - PostgreSQL ≥ 10
 - MySQL Workbench, HeidiSQL
 - Python ≥ 3.7 + dodatki
 - MongoDB, Neo4j, MonadDB
 - Gonilniki za dostop do baze (npr. ODBC)
-
- Lastni računalniki?
 - Docker?



Izpitni red

1. Obveznosti vaj: iz domačih nalog in seminarske naloge morate doseči posamično najmanj 50% možnih točk (pogoj za pristop k pisnemu izpitu).
2. Pisni izpit morate za pozitivno oceno pisati najmanj 50%
3. Vaje sestavljajo pol ocene in veljajo eno šolsko leto!
4. Sodelovanje na predavanjih/vajah se nagrajuje (subjektivno, do 10%)

**Torej: domače naloge $\geq 50\%$, seminarska naloga $\geq 50\%$,
izpit $\geq 50\%$**

Kako izgleda vaše predznanje v praksi?

- Programiranje: **Python**, Java, C/C++/C#, ...
- Programski dostop do podatkovne baze
- Načrtovanje podatkovnih baz (ER): OPB, RIS
- Normalizacija PB: OPB, RIS
- SUPB: **MySQL/MariaDB, PostgreSQL**, SQLite, Oracle, Microsoft SQL Server, NoSQL ?????
- Orodja ORM

Teme seminarskih nalog

- Specifična tematika, večja količina podatkov
- Praktična izvedba
- Dva študenta v skupini
- Teme (objavljene bodo na učilnici):
 - Praktična izvedba obdelave (analiza, vizualizacija, ...) nad izbranimi podatki (več nalog), na platformi SUPB + orodje
 - Specifične pregledne teme z vzorčno implementacijo (več nalog)
 - Stare teme seminarskih nalog za vzorec

Domača naloga

- Python, Python, Python !!!
 - Vzpostavite si delovno okolje
 - Preberite skripto "Python za programerje" in katerega od spletnih virov
 - Naredite čim več primerov
- Razmislite o svojih kompetencah s področja predmeta in o tem kakšen tip seminarske naloge ali seminarja bi vam najbolj ustrezal
- Oblikujte skupine po **natanko dva** študenta

Terminologija

- Sistem za upravljanje s podatkovnimi bazami (SUPB)
- Podatkovna baza (data base)
- Podatkovno skladišče (data warehouse)
- Sodobni (ne)relacijski SUPB:
 - NoSQL = Not only SQL
 - NewSQL

Sistemi za upravljanje s PB (SUPB)

- Angleško: Database Management System (DBMS)
- Sistem za upravljanje s podatkovno bazo – SUPB je programska oprema za obvladovanje velikih količin podatkov, shranjenih v vneprej točno določeni obliki (logičnem podatkovnem modelu)
- Alternativa – shranjevanje v aplikaciji lastni obliki; problemi: neprenosljivost, nefleksibilnost ...
- Obstaja veliko vrst SUPB. Omejili se bomo predvsem na relacijske in nekatere sodobne nerelacijske:
 - Oracle, Microsoft SQL Server, Postgres, MySQL/MariaDB,
 - MongoDB, Neo4j, MonadDB, ...

Podatkovna baza in podatkovno skladišče

- Podobno, vendar ne enako!
- Podatkovna baza (PB oz. DB):
 - OLTP sistem (on-line transactional processing)
 - opisuje trenutno stanje
- Podatkovno skladišče (PS oz. DW):
 - OLAP sistem (on-line analytical processing)
 - opisuje zgodovino vsebin OLTP sistema (pogosto več OLTP sistemov)
- Oba pristopa tečeta na SUPB, vendar z različnimi prioritetami izvajanja (OLTP - hitro izvajanje transakcij, OLAP - hitra analiza)
- Razlikujejo se tudi postopki načrtovanja

Načrtovanje PB, PS, NoSQL

- Podatkovne baze
 - konceptualno načrtovanje:
 - obsežne PB, veliko število tabel
 - korak pri načrtovanju inf. sistemov
 - logično načrtovanje (normalizacija relacij oz. tabel):
 - direktno za manjše PB (nekaj deset tabel)
 - preverjanje rezultatov konceptualnega načrtovanja
 - fizično načrtovanje
- Podatkovna skladišča
 - načrtovanje zvezdnih shem
- Načrtovanje nerelacijskih PB
 - novo, nezrelo področje

Zakaj razlike?

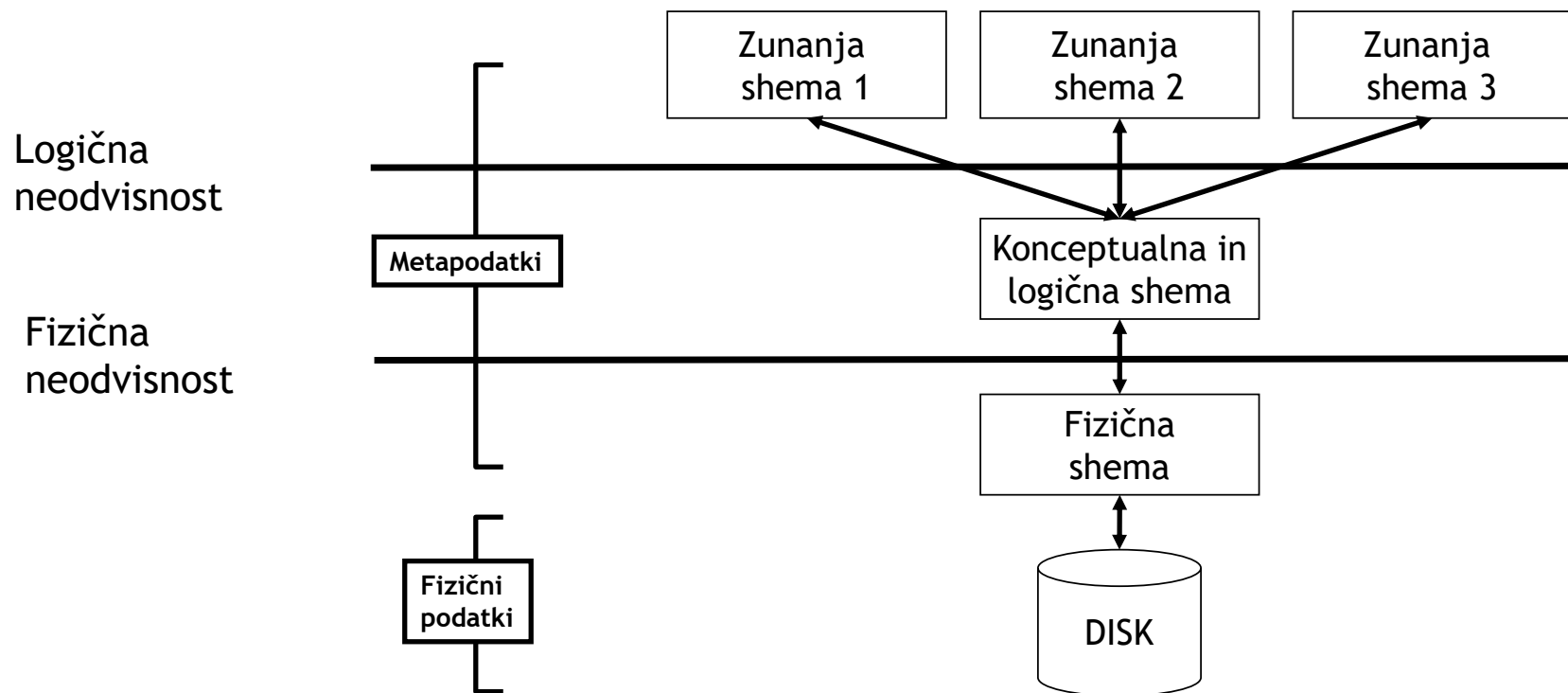
- PB: hitro izvajanje transakcij
- PS: hitro izvajanje analiz podatkov
- NoSQL: nerelacijski podatkovni modeli

Poglavje 1

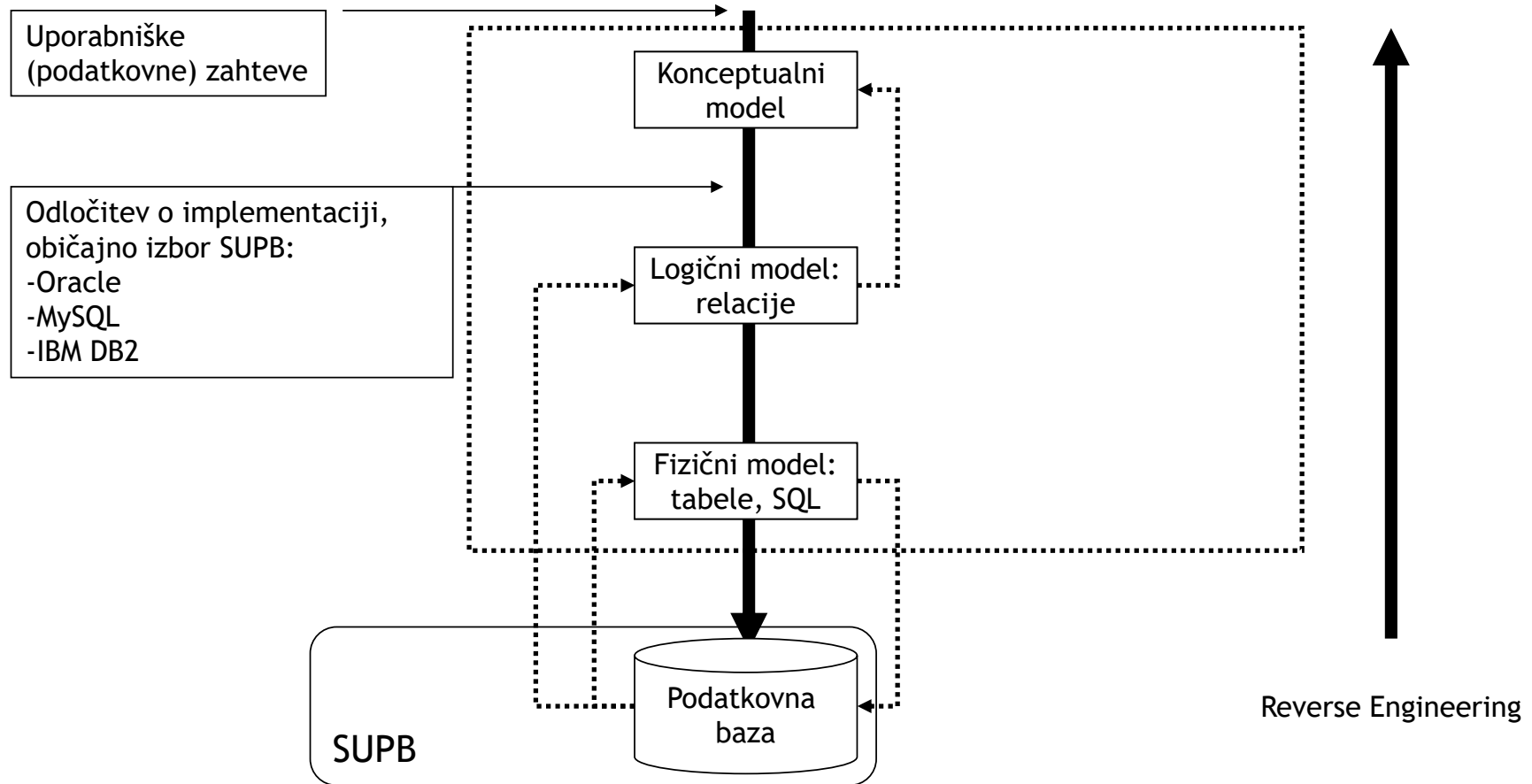
**Konceptualno načrtovanje
transakcijskih podatkovnih baz**

Trinivojska predstavitev podatkov...

- Podatki so v PB opisani na treh nivojih:
 - Zunanje sheme (zunanji, uporabniški nivo)
 - Konceptualna (logična) shema
 - Fizična (notranja) shema



1.1 Trije nivoji načrtovanja – trije modeli..



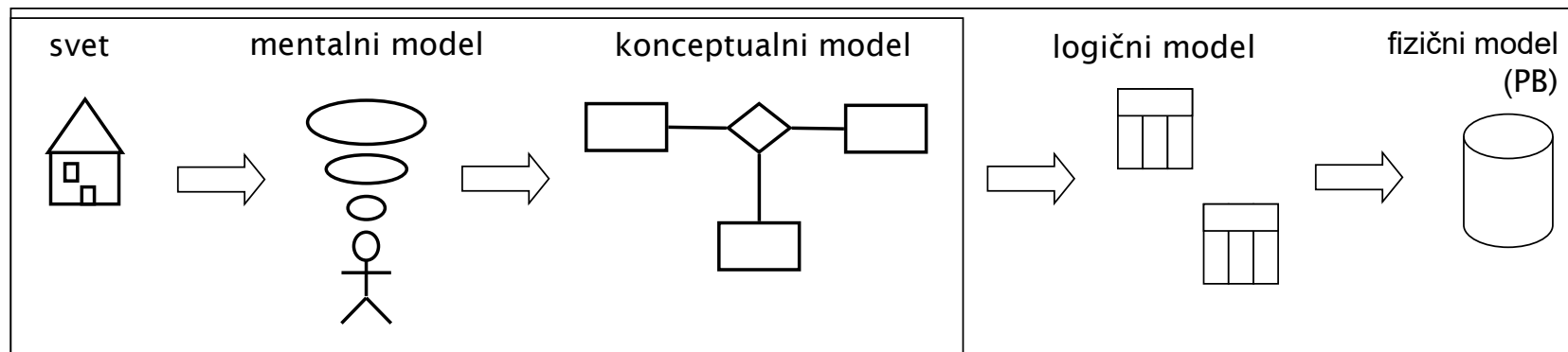


Konceptualni podatkovni model

- Formalizem s katerim opišemo, kaj bi želeli hraniti v PB ter kakšne povezave obstajajo med elementi, ki jih želimo hraniti, se imenuje konceptualni model.
- Konceptualni model je torej način, kako na visoki ravni abstrakcije razumljivo in ne preveč tehnično opišemo podatke, ki jih želimo hraniti ter skrijemo nepomembne podrobnosti.
- Konceptualni model odraža uporabnikovo zaznavanje realnega sveta oziroma poslovnega problema, ter načrtovalčevo videnje problema (stična točka).

1.1 Trije nivoji načrtovanja – trije modeli

- Načrtovanje PB: od realnega sveta do fizične podatkovne baze za potrebe poslovne domene



1.2 Kaj je konceptualno načrtovanje?..

- Konceptualno načrtovanje je postopek opredelitve podatkovnih potreb oz. zahtev (poslovne ali druge) domene s pomočjo konceptualnega modela
- Konceptualno načrtovanje preko konceptualnega modela poskrbi za opis pomena podatkov, potrebnih za poslovno domeno
- Konceptualno načrtovanje je neodvisno od dejanskega podatkovnega modela (relacijski, ...)
- Konceptualnega načrtovanja ne moremo avtomatizirati, za njegovo izvedbo je odgovoren analitik. Gre za prenos semantike v model.

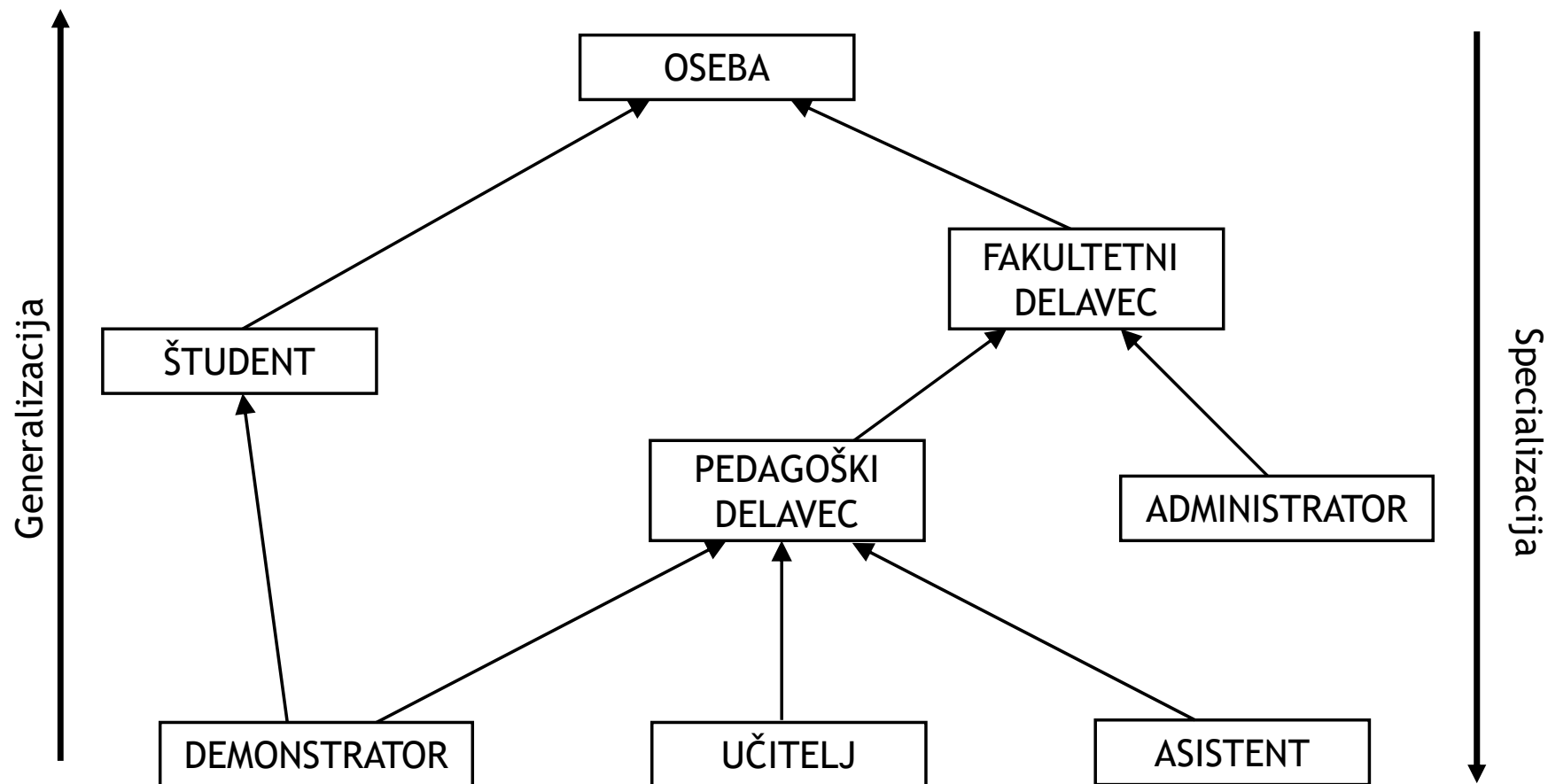
1.2 Kaj je konceptualno načrtovanje?

- Je najbolj kritično, saj se napake narejene pri konceptualnem načrtovanju prenašajo naprej na naslednje modele
- Pri konceptualnem načrtovanju je zelo pomembno sodelovanje uporabnikov in interakcija z uporabniki. Uporabniki so nosilci znanja o poslovni domeni, so poznavalci semantike in običajno vedo, kaj hočejo, a to težko natančno izrazijo
- Konceptualno načrtovanje mora upoštevati tudi poslovna pravila (kot omejitve modela)

1.3 Tehnike konceptualnega načrtovanja..

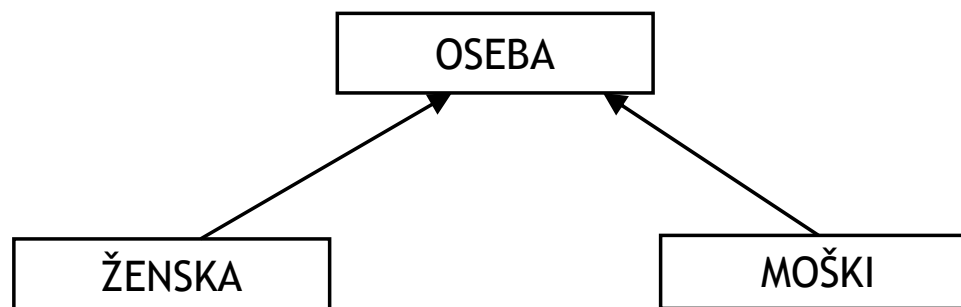
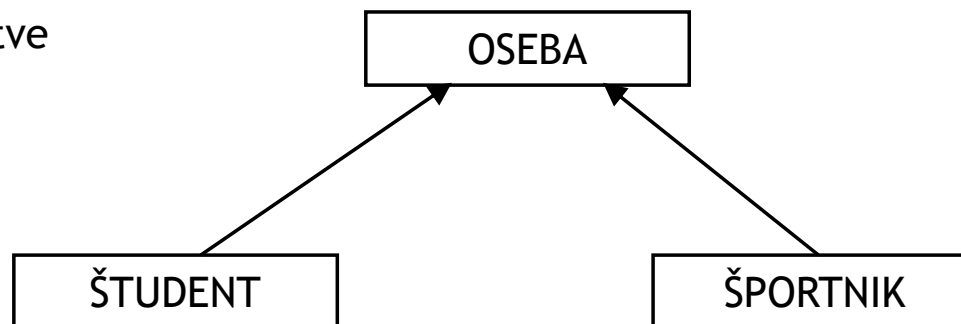
- Percepcija
- Abstrakcija
- Objekti, koncepti (tipi) in término
- Klasifikacija (objekti→koncepti)
- Ureditev hierarhij konceptov (tipov)
- Agregacija (sestavljanje konceptov)

1.3 Tehnike konceptualnega načrtovanja..

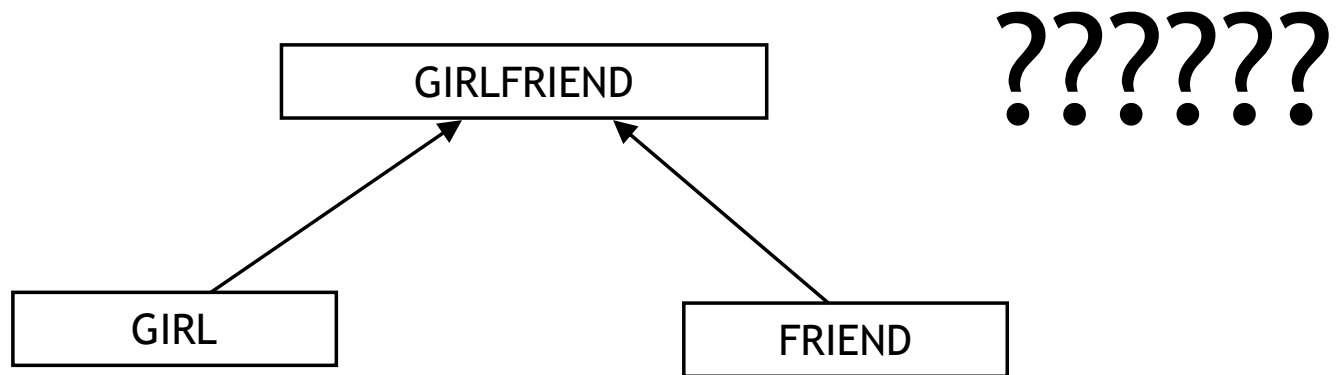


1.3 Tehnike konceptualnega načrtovanja..

Primera ureditve
tipov:
- pokrivanje?

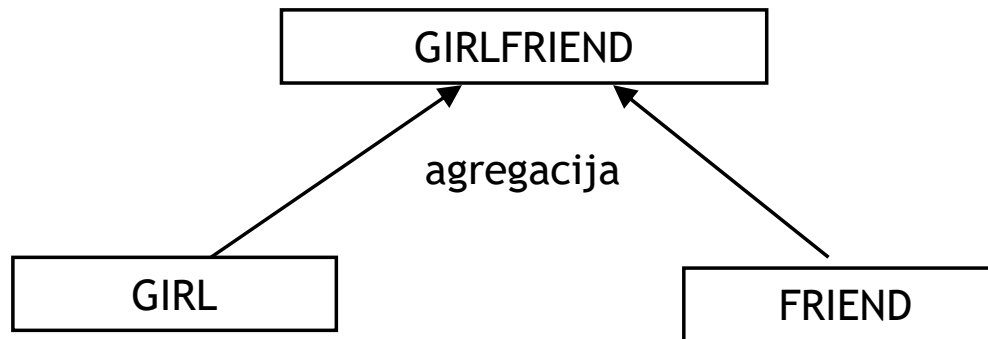


1.3 Tehnike konceptualnega načrtovanja..



1.3 Tehnike konceptualnega načrtovanja

- Preko agregacije definiramo nov – agregiran (sestavljen) tip na osnovi obstoječih tipov



- Kakšne je razlika z generalizacijo? Agregirani tip vsebuje VSE lastnosti agregiranih tipov in ne samo skupnih!
- Pomensko agregirani tip ponuja nekaj novega.

1.4 Lastnosti konceptualnega modela..

- Namembnost konceptualnega modela
 - Razumevanje problema (načrtovalec)
 - Vrednotenje razumevanja (uporabnik)
 - Pomoč pri načrtovanju in implementaciji (razvijalec)
- Zahtevane lastnosti konceptualnega modela
 - Izraznost (prikaz različnih konceptov),
 - Preprostost (enostaven za uporabo in razumevanje),
 - Minimalnost (vsak koncept predstavljen enolično),
 - Formalnost (natanče, nedvoumno definiran pomen konceptov)
 - Grafična popolnost (vsi koncepti razpoznavno predstavljeni)
 - Berljivost (pregledna predstavitev gradnikov in celotnega modela).

1.5 Entitetni model..

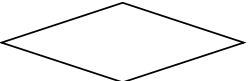
- Najpogostejše uporabljana tehnika za predstavitev konceptualnih podatkovnih modelov sta entitetni model (model entiteta-razmerje) ter razredni diagram. Obravnavali bomo entitetni model.
- Nazivi, ki se uporabljajo:
 - Konceptualni podatkovni model
 - Podatkovni model
 - Entitetni model
 - ER model
- Razširjeni entitetni model

1.5 Entitetni model..

- Entitetni tip
- Atribut
- Razmerje
- Identifikator

1.5 Entitetni model..

Grafična predstavitev gradnikov entitetnega modela

- Originalni Chenovi diagrami
oznaka  za razmerja (relacije)
- Običajni ER diagrami: vranja noga (crow foot, IE)
oznaka  za nekatera razmerja (relacije)
 - Chenovi diagrami so bolj izrazni, vendar kompleksnejši in dopuščajo različne interpretacije razmerij.
- UML (Unified Modeling Language)
-

1.5 Entitetni model..

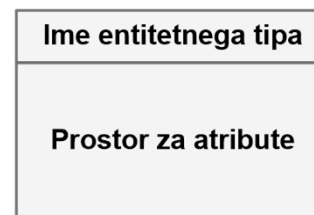
- Entitete so posamezne instance (primerki) tipov objektov iz poslovne domene: dogodki, predmeti, osebe, pravila, dejstva
- O entitetah obstaja določena predstava o tem:
 - kakšne lastnosti dejansko imajo
 - kakšne lastnosti jim moramo določiti (morajo imeti), da bodo izpolnjevale poslanstvo entitetnega modela
- Na osnovi predstave o tem in percepcije, lahko entitete klasificiramo v entitetne tipe: vse entitete, ki ustrezajo določeni predstavi, pripadajo posameznemu entitetnemu tipu.
- Primer: študenti

1.5 Entitetni model..

- Vsak trenutek pripada posameznemu entitetnemu tipu množica entitet tega entitetnega tipa, ki jo imenujemo entitetna množica
- Entitetna množica je časovno spremenljiva: entitete nastajajo, se spreminjajo in tudi izginjajo (izstopajo iz množice).
- Entitetna množica je v nekem trenutku lahko tudi prazna.
- Natančno moramo poznati pomen entitetnega tipa: kaj predstavljajo entitete, ki mu pripadajo

1.5 Entitetni model..

- V praksi se pogosto uporablja poenostavljen izraz entiteta, čeprav bi se moral uporabljati izraz entitetni tip



1.5 Entitetni model..

- Entitete imajo določene lastnosti, posamezne entitete (iz istega entitetnega tipa) se med seboj razlikujejo po njihovi vrednosti
- Le del entitetnih lastnosti je zanimiv oz. pomemben za opazovano poslovno domeno (abstrakcija)
- Lastnosti, ki so pomembne za opazovano poslovno domeno, vključimo v konceptualni model tako, da jih kot attribute določimo entitetnemu tipu

1.5 Entitetni model..

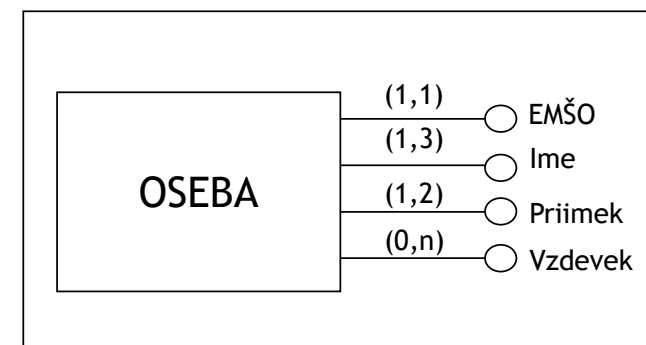
- Torej: z atributi formalno opišemo lastnosti entitet
- Govorimo o več vrstah lastnosti:
 - Entitetna imena: naziv, ime, opis
 - Prave entitetne lastnosti: višina, teža, cena, vrednost
 - Lastnosti, ki jih določimo za potrebe poslovnih procesov, poslovnih funkcij in poslovnih pravil: šifre, statusi, ...
- Atribut določimo za tisto lastnost, ki je za problemsko/poslovno domeno pomembna
- Vsak atribut ima določene lastnosti: kardinalnost (števnost), tip, dolžina

1.5 Entitetni model..

- Kardinalnost atributa omejimo z minimalno in maksimalno vrednostjo (min,max):

– Totalni atribut	(1,n), kjer je $n \geq 1$
– Parcialni atribut	(0,n), kjer je $n \geq 1$
– Enovrednostni atribut	(m,1), kjer je $m \in \{0,1\}$
– Večvrednostni atribut	(m,n), kjer je $m \in \{0,1\}$ in $n > 1$

- Minimalna števnost 0 pomeni, da je atribut lahko brez vrednosti (ni obvezen).
- Maksimalna števnost n pomeni, da atribut lahko zavzame poljubno število vrednosti.

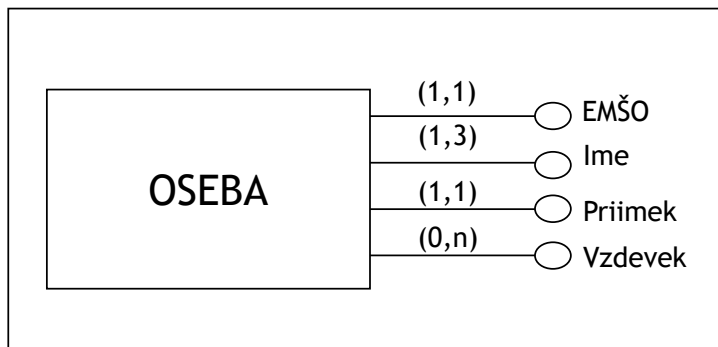


1.5 Entitetni model..

- Atribut pripada določenemu abstraktnemu tipu: numerični, znakovni, ...
- Za večino tipov je potrebno določiti tudi dolžino.
- V Chenovih ER diagramih imajo atributi lahko poljubno kardinalnost
- V običajnih ER diagramih imajo atributi kardinalnost vedno (0,1) ali (1,1)
- Večvrednosten atribut predstavimo z drugimi gradniki (nov entitetni tip in razmerje)

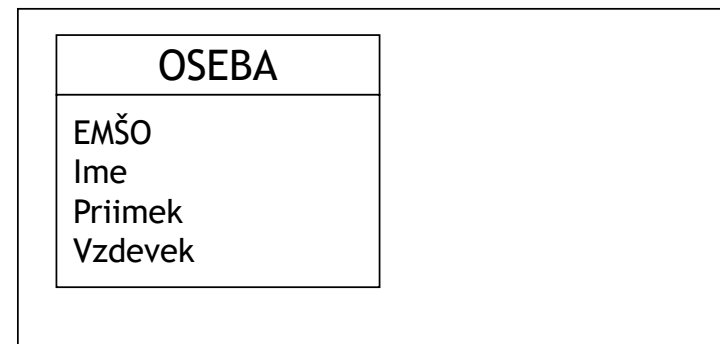
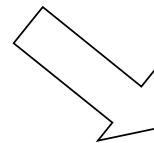
Atribut

1.5 Entitetni model..



Chenov diagram

Običajen ER diagram



1.5 Entitetni model..

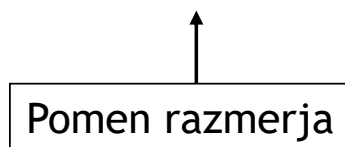
- Entitete niso svet zase, medsebojno se povezujejo preko razmerij (relacij)
- Razmerje ima določen pomen in števnost
- Predstavitev razmerja v modelu entiteta-razmerje je povezava
- Med opazovanim parom (v splošnem podmnožici) entitetnih tipov je lahko več razmerij: OSEBA, KRAJ – stalno bivališče, začasno bivališče
- Razmerij je lahko veliko, izberemo samo tista, katerih evidenco želimo voditi!

1.5 Entitetni model..

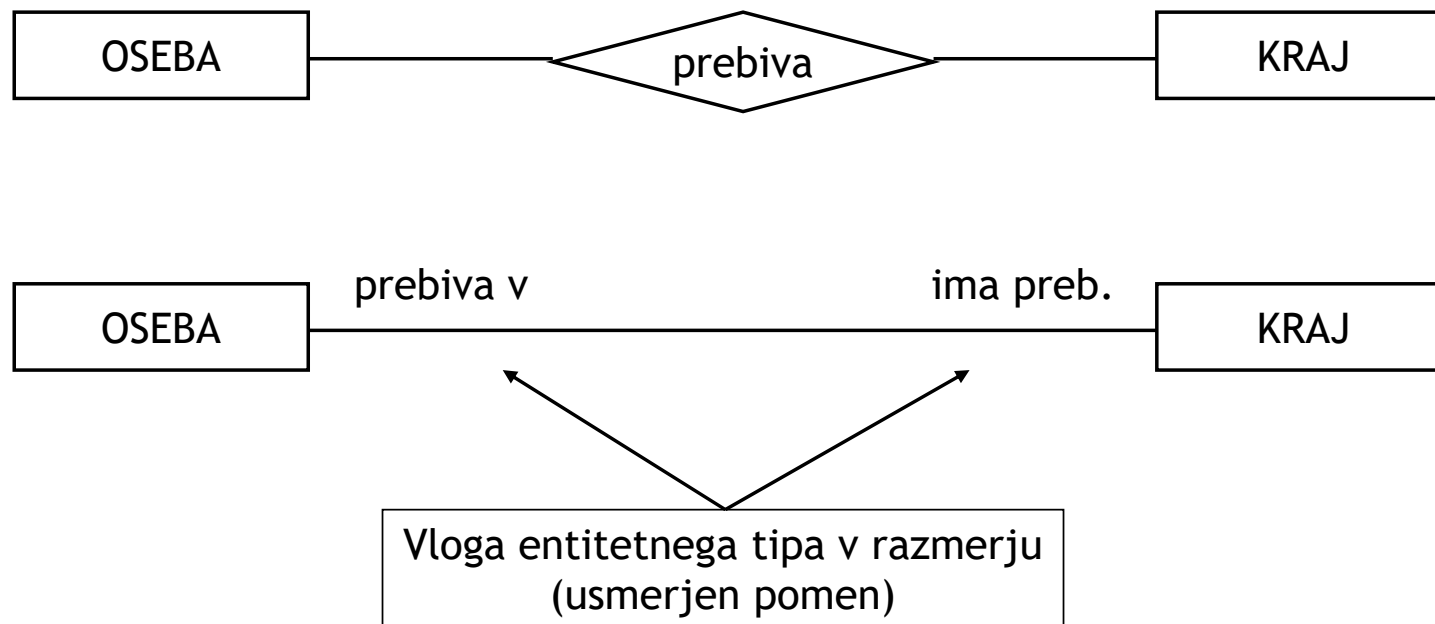
Chenov diagram



Običajen ER diagram



1.5 Entitetni model..

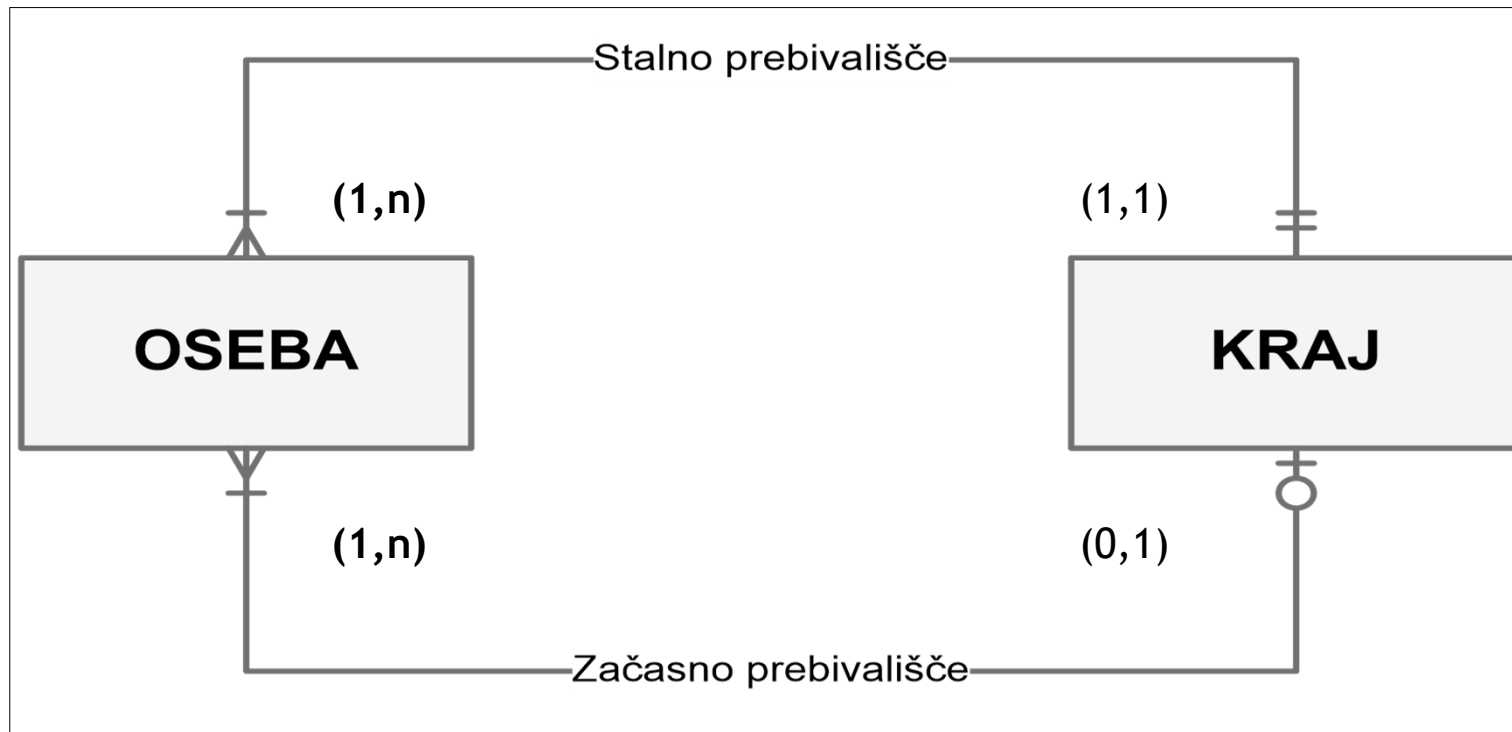


1.5 Entitetni model..

- Kardinalnost (števnost) predstavlja število entitet entitetnega tipa, ki so v razmerju glede na pomen razmerja. Minimalna in maksimalna števnost določata veljaven razpon števila entitet.
- Vsak entitetni tip ima svojo kardinalnost v razmerju glede na vlogo. Pomen razmerja med entitetnima tipoma OSEBA, KRAJ (na naslednji strani):
 - Ena (poljubna) oseba ima stalno bivališče v enem kraju (pošti)
 - V enem (poljubnem) kraju (pošti) ima stalno bivališče več oseb

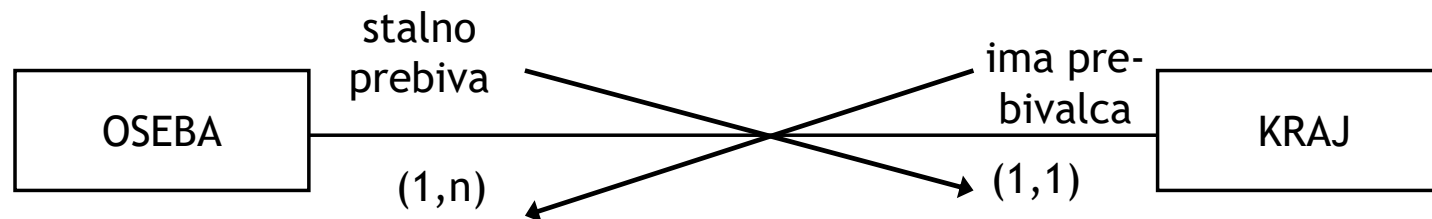
1.5 Entitetni model..

- Razmerji med entitetama OSEBA in KRAJ



1.5 Entitetni model..

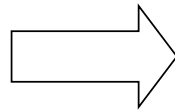
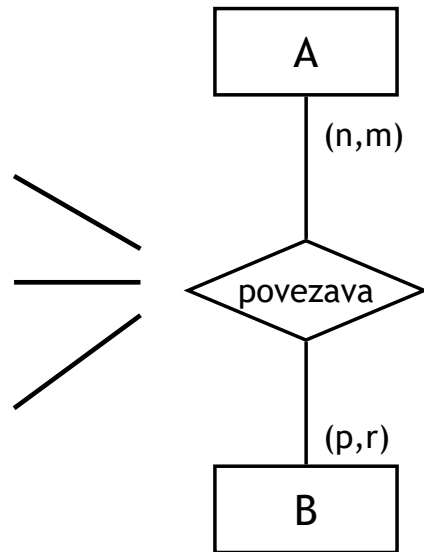
- Vloga in števnost v delnem razmerju



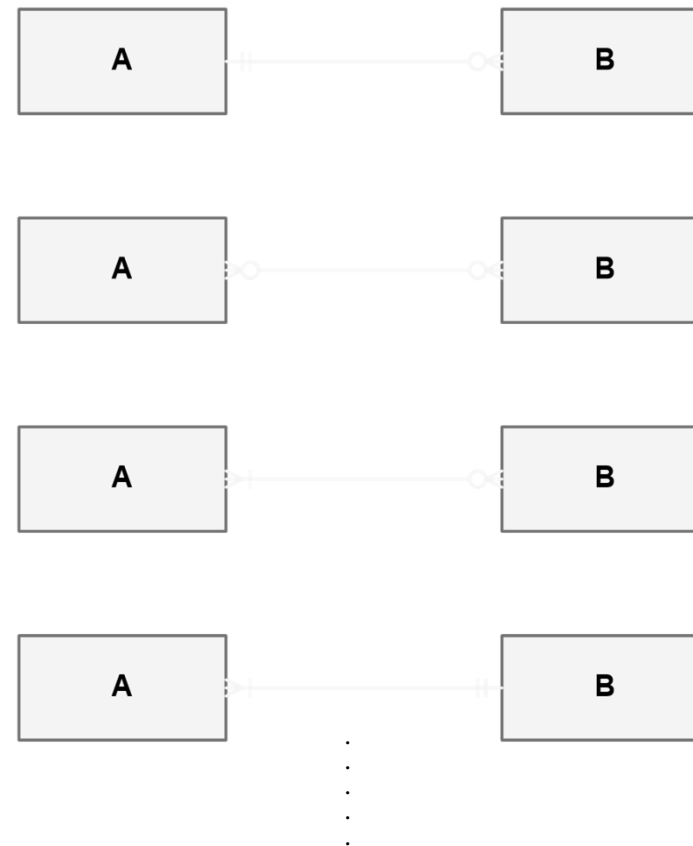
- Oznake za števnost na ponoru delnega razmerja:
 - 0: --o-- (krožec na povezavi)
 - 1: --|-- (pravokotna črta na povezavi)
 - več (m ali n): vranja noga  (crow foot)
 - Razpon določimo z dvema števkama (minimalna, maksimalna) ali kombinacijo grafičnih oznak
 - (1,1) se označuje kot --|-- ali --||--
 - (0,1) se označuje kot --o-- ali --o|--

1.5 Entitetni model..

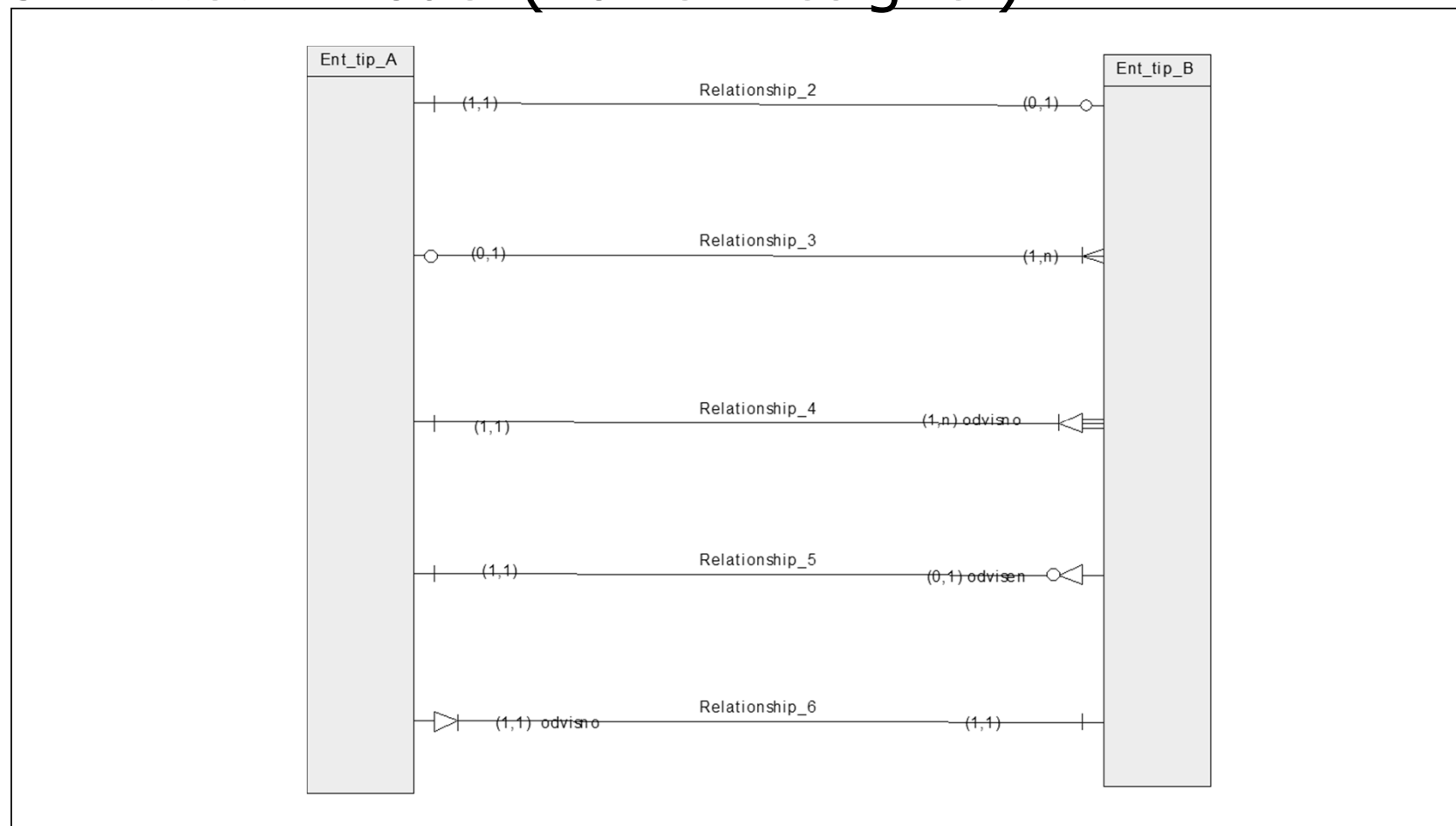
Chenov diagram (omogoča
tudi več kot dvomestna
razmerja)



Razmerje - kardinalnost



1.5 Entitetni model (Power Designer)



1.5 Entitetni model..

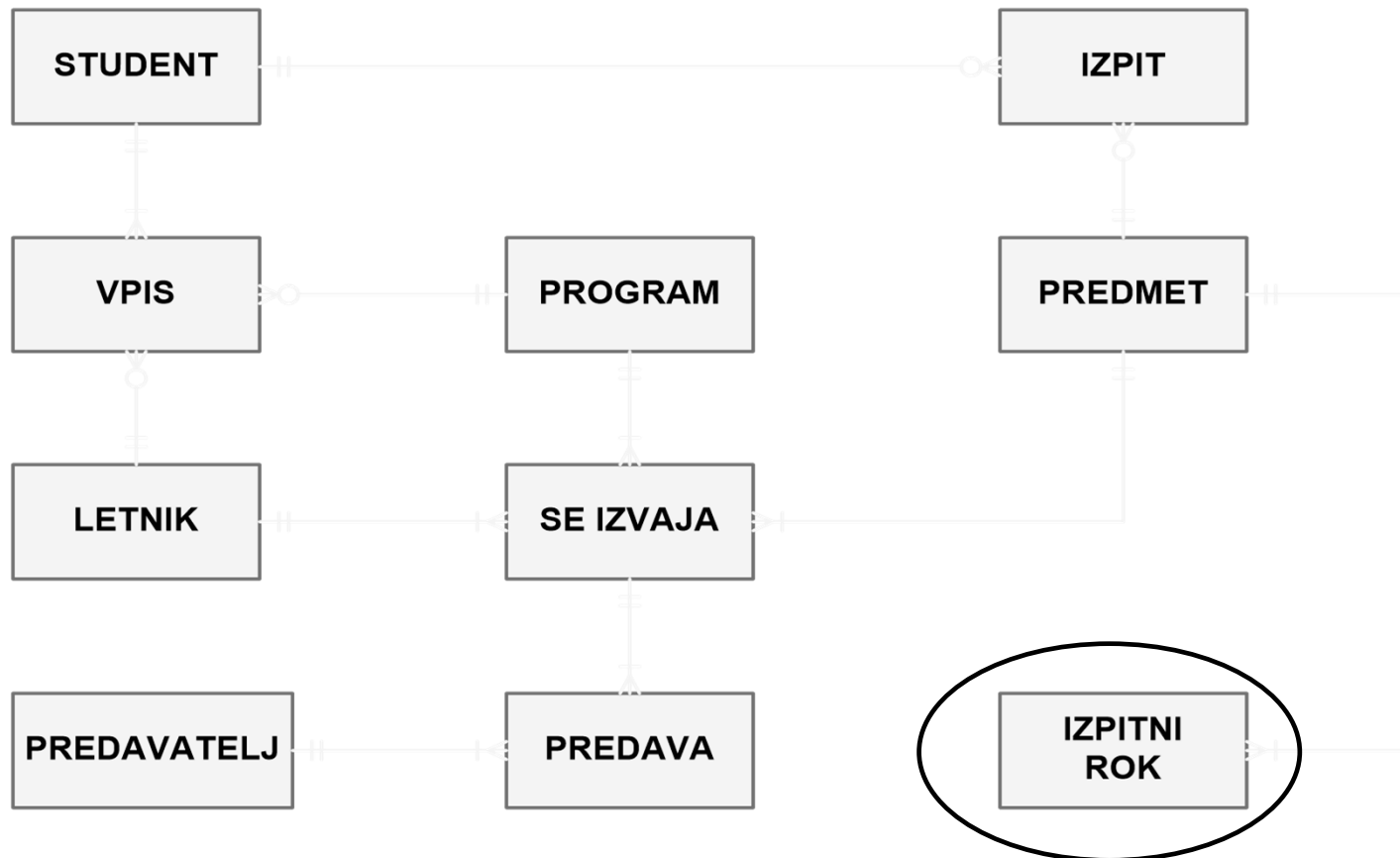
- Mandatnost pove, ali sta dve entiteti vedno v razmerju ali lahko tudi nista v razmerju: obvezno, neobvezno razmerje
- Mandatnost lahko obravnavamo pod okriljem števnosti, zaradi česar dodatno uvedemo števnost 0

1.5 Entitetni model..

- Razmerje tudi opisuje lastnost entitete
- Primer: OSEBA, KRAJ (poštna številka)
- Razmerje ima atributiven značaj
- Dilema: atribut (lastnost) ali razmerje?
 - Stalno (1,1) in začasno (0,1) prebivališče
 - Izpitni rok (3,3) ali (1,n)

Primer: eŠtudent

1.5 Entitetni model..



1.5 Entitetni model..

- Enolični identifikator entitete (krajše: entitetni identifikator ali samo identifikator) je podmnožica lastnosti entitetenega tipa (atributov in razmerij do drugih entitetnih tipov), ki enolično razlikujejo posamezne entitete znotraj entitetne množice
- Glede na to, ali tvorijo identifikator entitete le lastni atributi entitetenega tipa ali pa je v enoličnem identifikatorju tudi kakšno razmerje, ločimo med močnim entitetnim tipom in šibkim entitetnim tipom

1.5 Entitetni model..

- Imamo lahko več enoličnih identifikatorjev, vendar moramo enega izbrati – določiti
- Izbrani – določeni enolični identifikator je podlaga za (primarni) ključ v relacijskem modelu

Identifikator:

Abstrakten pojem

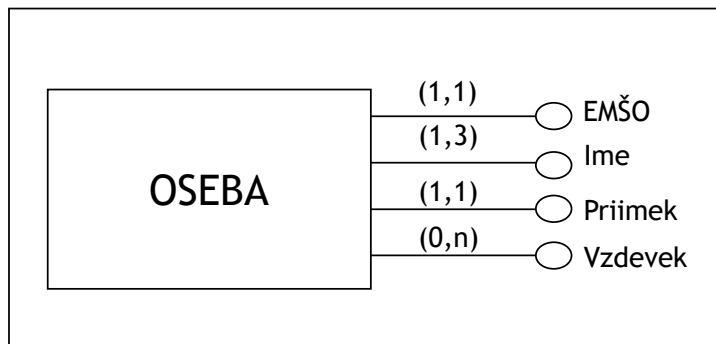
Ključ:

Relacijski model

- Identifikator se preslika v ključ relacije.

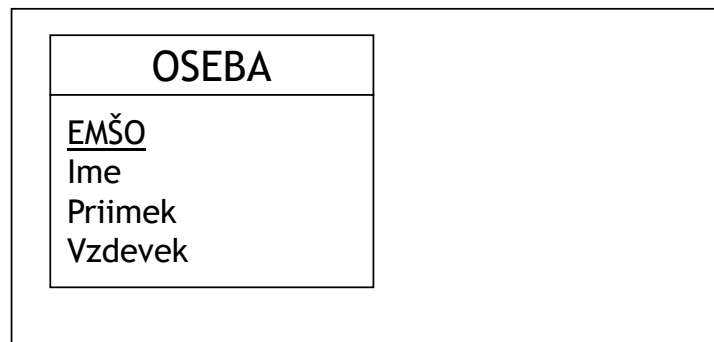
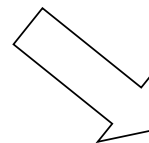
Enolični identifikator entitete

1.5 Entitetni model..

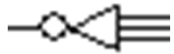


Chenov diagram
(pobarvamo piko ali pike)

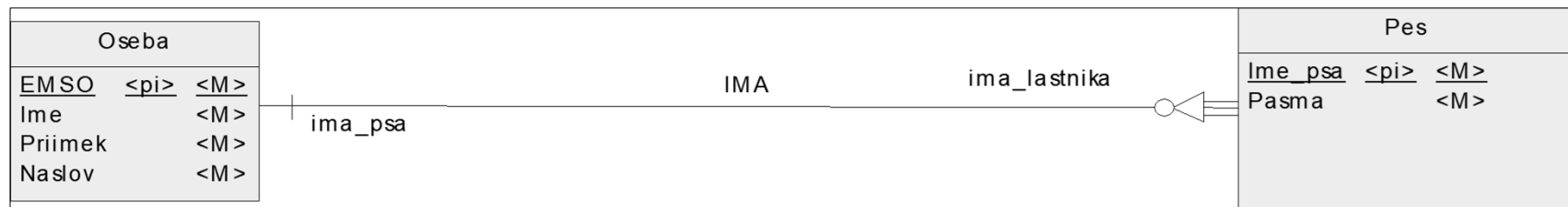
Običajen ER diagram
(podčrtamo atribut ali
attribute)



1.5 Entitetni model..

- Glede na attribute, ki sestavljajo identifikator, ločimo entitetne tipe na
 - Močne:
 - za identifikacijo objekta zadoščajo lastni atributi
 - Šibke:
 - za identifikacijo objekta lastni atributi ne zadoščajo
 - potrebujemo še nekaj **tujih** atributov
 - dobimo jih kot primarne ključe povezanih entitetnih tipov, od katerih je naš šibki entitetni tip **odvisen**.
 - odvisnost označimo s puščico (trikotnikom) na povezavi: 
 - Šibki entitetni tipi pogosto nastanejo po pretvorbi iz večmestnih (>2) ali večvrednostnih ($m:n$) razmerij, ter razmerij z atributi

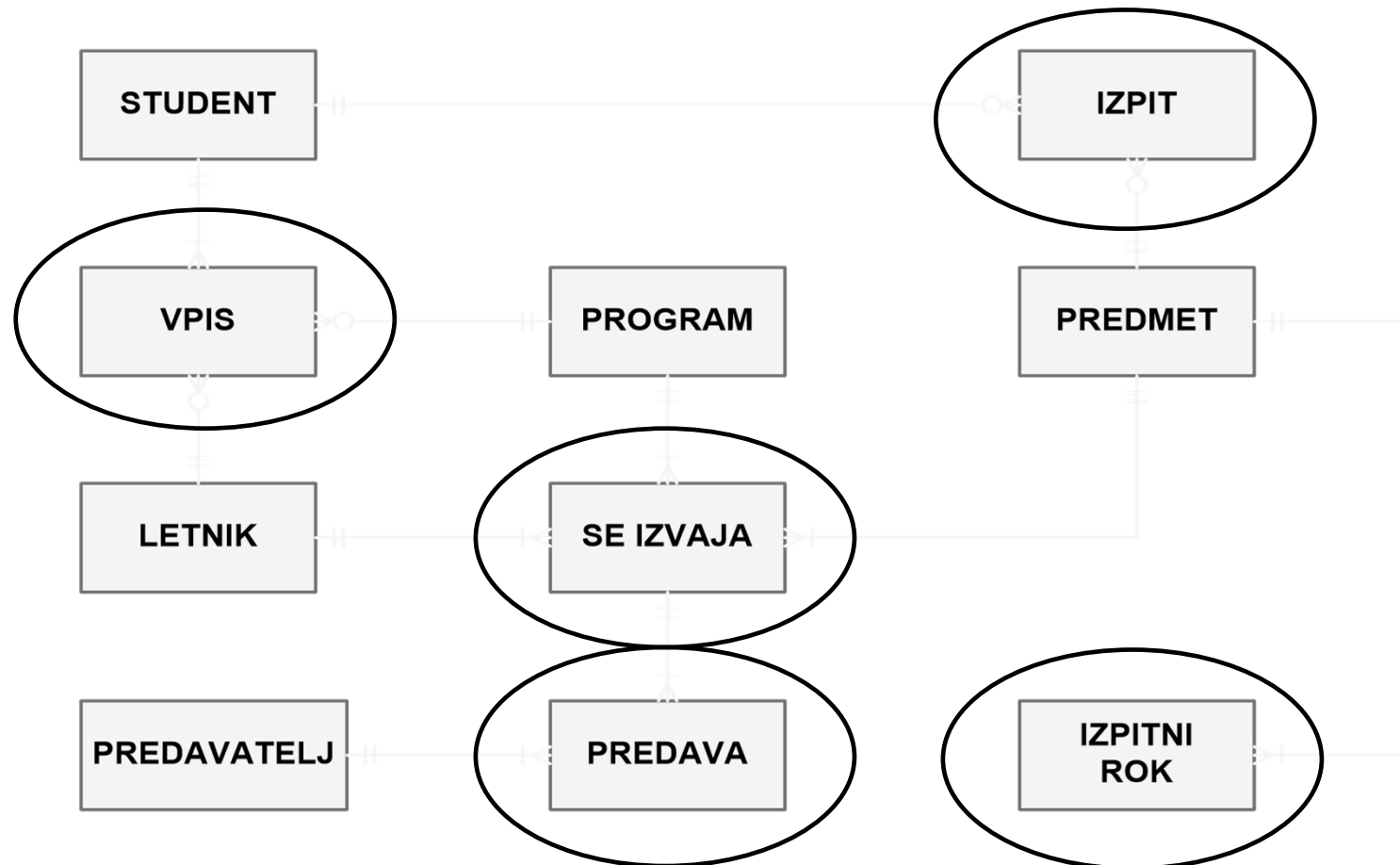
1.5 Entitetni model..



- Trikotnik označuje odvisnost (dependency) šibkega entitetnega tipa od močnega (v smetri puščice trikotnika). Kardinalnost: (0,n)
- Identifikator šibkega ent. tipa pes: Ime_psa in EMSO skupaj!
- Identifikator močnega entitetnega tipa Oseba: EMSO

Primer: eŠtudent

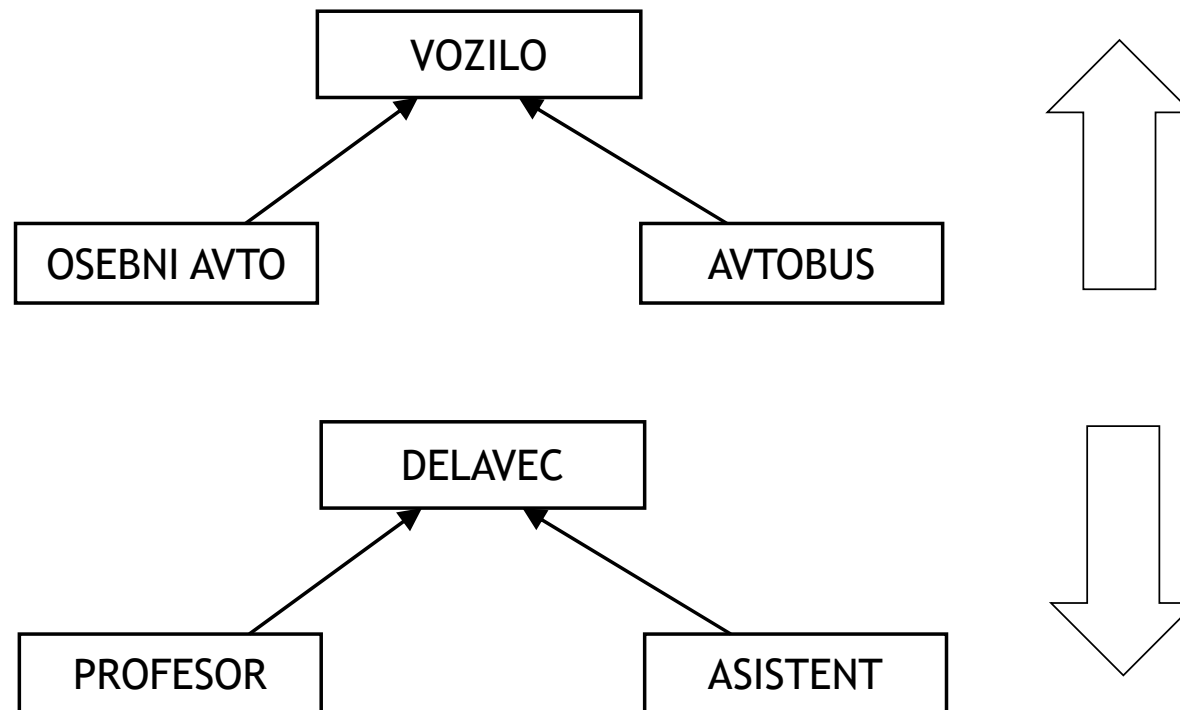
1.5 Entitetni model..



1.5 Razširjeni entitetni model..

- Razvrščanje sorodnih entitetnih tipov v hierarhije
- Izognemo se redundantnosti, povečamo semantiko
- Generalizacija
 - Imamo entitetna tipa B in C
 - Generaliziramo (posplošimo) ju v nadtip A
 - Pri tem se skupni atributi prenesejo na nadtip
- Specializacija:
 - Imamo entitetni tip A
 - Specializiramo ga v podtipa B in C
 - Podtipa ne vsebujeta skupnih atributov.

1.5 Razširjeni entitetni model..



1.5 Razširjeni entitetni model..

- Entitetni tip A s podtipoma B in C
- B in C pokrivata A totalno in ekskluzivno, če velja:
 $E_B \cup E_C = E_A$ in $E_B \cap E_C = \{\}$
- B in C pokrivata A totalno in prekrivno, če velja:
 $E_B \cup E_C = E_A$ in $E_B \cap E_C \neq \{\}$
- B in C pokrivata A delno in ekskluzivno, če velja:
 $E_B \cup E_C \subset E_A$ in $E_B \cap E_C = \{\}$
- B in C pokrivata A delno in prekrivno, če velja:
 $E_B \cup E_C \subset E_A$ in $E_B \cap E_C \neq \{\}$

1.5 Razširjeni entitetni model..

