

Увод у релационе базе података

8



Саша Малков
Универзитет у Београду
Математички факултет
2023/2024

[PM13]
Увод у РБП
Саша Малков



Тема 6

Концептуално пројектовање

[PM13] Увод у релационе базе података – Саша Малков – 2021/22 – час 8

1

Концептуално моделирање базе података



Концептуални модел

- Представља највиши ниво апстракције
- Описује препознату семантику посматраног домена
 - описује структуре и међузависности података из угла корисника
 - веома је важно да описује тачну семантику података и односа
 - која се касније губи на нижим нивоима апстракције
- Посебно важан
 - за разумевање целине система
 - за разумевање улога делова система
- Модел се обликује и записује применом
 - ЕР моделирања
 - УМЛ моделирања

Универзитет у Београду - Математички факултет

[PM13] Увод у релационе базе података – Саша Малков – 2021/22 – час 8

2

Концептуално моделирање базе података



Кораци концептуалног пројектовања

- Концептуално пројектовање има следеће кораке:
 - Прављење појединачних концептуалних модела
 - Анализа захтева
 - Концептуално моделирање података
 - Прављење обједињеног концептуалног модела
 - Интегрисање погледа (локалних схема)
 - Груписање ентитета

Универзитет у Београду - Математички факултет

[PM13] Увод у релационе базе података – Саша Малков – 2021/22 – час 8

3



Теме концептуалног пројектовања

- У наставку ћемо обрадити следеће теме везане за концептуално пројектовање:
 - ЕР-модел
 - УМЛ модел домена (дијаграм класа података)
 - Прављење појединачних концептуалних модела
 - Прављење обједињеног концептуалног модела

[PM13]
Увод у РБП
Саша Малков



Тема 6.1

Концептуално пројектовање

-

Модел ентитета и односа



Модел ентитета и односа

- Предложио га је 1976. године Питер Чен
 - Предложен је као наредни корак после мрежног, хијерархијског и релационог модела
 - “Рад представља модел ентитета и односа, који је напреднији од наведена три модела. Модел ентитета и односа усваја природнији приступ да се стварни свет састоји од ентитета и односа.”
- Основна критика постојећих модела:
 - Не чувају мета информације о ентитетима и односима међу њима
 - “Релациони модел... може да изгуби неке важне семантичке информације о стварном свету.”
 - “Рад користи модел ентитета и односа као оквир из кога могу да се развију три постојећа модела”



Основне претпоставке

- Модел ентитета и односа препознаје два различита основна концепта:
 - ентитете
 - односе
- Тежи да у самом моделу очува све битне семантичке информације о домену који се моделира
 - што ниједан од претходних модела не омогућава



Термини

- Пун назив
 - “Entity-Relationship Model”
 - “Модел ентитета и односа”
- Скраћено
 - “ER Model”
 - “ЕР-модел”
 - иако није у духу српског језика
 - “Модел ЕО” би било исправније, али би нас мало ко разумео
- Проширени модел ентитета и односа
 - Данас се често изоставља и подразумева “проширени”
 - Додају се елементи као:
 - хијерархије
 - агрегације



Појам ентитета и односа

- “Енџијети” је ствар која може да се једнозначно идентификује”
 - особа, предузеће, догађај
- “Огнос је неко међусобно придруживање ентитета”
 - отац-син, радник-предузеће



Представљање ентитета и односа

- “Ентитети се класификују у различите *скујове енџијети*.
- Сваком скупу ентитета одговара предикат који проверава да ли му неки ентитет припада.
- Скупови ентитета не морају да буду дисјунктни.”
- “Скуј односа је математичка релација између N ентитета који припадају неким скуповима ентитета, а чији елементи су односи.”
- “Улога ентитета у односу је функција коју он обавља у односу.”
- Приметимо да и “скуп ентитета” и “скуп односа” имају много тога заједничког са концептом релације у релационом моделу



Информације о ентитетима и односима

- “Информације о ентитетима и односима... се изражавају скупом парова *атрибути-вредности*. Вредности се класификују у *скујове вредности*.”
 - скупови вредности одговарају домену атрибута
- “Атрибути може да се дефинише као функција која пресликава скуп ентитета или скуп односа у скуп вредности или Декартов производ скупова вредности.”
 - и ово веома личи на релациони модел



Структура информација

- “Примарни кључ”
 - мора да постоји средство за разликовање и једнозначно реферисање елемената скупова ентитета или односа
- “Релације ентитета и односа”
 - скупови ентитета и скупови односа се моделирају релацијама (табелама)



Атрибути

- Атрибути су “елементарни” описи ентитета и односа
- Информације о ентитетима и односима се изражавају у облику скупа атрибута
- Атрибути су парови облика “*име=вредност*”
 - могу да се моделирају попут функција, налик на релациони модел



Атрибути (2)

- Атрибути могу да буду
 - основни
 - представљају тачну слику домена
 - не могу да се изведу из других атрибута
 - дефинисани
 - измишљени само ради имплементације базе података
 - једном дефинисане вредности се касније не мењају
 - разни идентификатори и сл.
 - изведени
 - могу да се израчунају из других атрибута
 - не би требало да се чувају у бази података



Атрибути (3)

- Атрибути могу да буду
 - обавезни
 - сваки појединачан ентитет из неког скупа ентитета има дефинисану вредност атрибута
 - опциони
 - не мора сваки ентитет да има дефинисану вредност атрибута
 - еквивалентно проширењу домена атрибута “празним” или “недефинисаним” вредностима



Слаби и јаки ентитети

- ЕР-модел разликује
 - слабе ентитетске релације
 - у идентификовању ентитета учествују односи са другим ентитетима
 - тј. у пракси у примарном кључу учествује референца на други ентитет
 - обичне (регуларне) ентитетске релације
 - у идентификовању ентитета не учествују односи са другим ентитетима
 - тј. у примарном кључу не учествује референца на други ентитет
 - мада некада то може да буде случај у моделу
 - и слично за односе



Слаби и јаки ентитети (2)

- То обично у литератури прераста у једноставније концепте
 - јаки ентитети
 - идентификују се сами за себе
 - углавном независни од других ентитета у бази
 - њихово постојање није условљено постојањем других ентитета
 - слаби ентитети
 - идентификују се само кроз однос са неким другим ентитетом
 - углавном не постоје самостално, без неких других ентитета
 - обично представљају саставни део или опис неког другог ентитета
- “Слаб ентитет” не значи да је то “мање важан ентитет”
 - нпр. ако примарни кључ ентитета *Сйудент* чине *ИД факултета* и *број индекса*, онда је *Сйудент* “слаб ентитет”
 - идентификован је (макар делимично) односом са ентитетом *Факултет*
 - одговара концепту композиције у ООМ

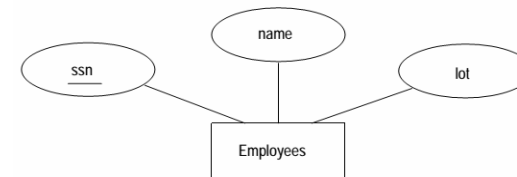


ЕР-дијаграми

- ЕР-дијаграми су саставни део модела
- Представљају основни начин записивања семантичких знања о информацијама



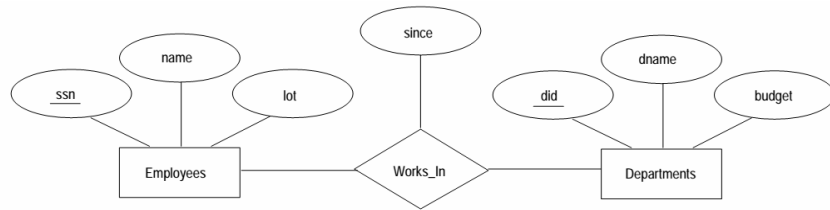
Ентитети и атрибути



- Скупови ентитета се представљају правоугаонцима
- Атрибути се представљају елипсама
- Примењивост атрибута на скуп ентитета се представља линијом



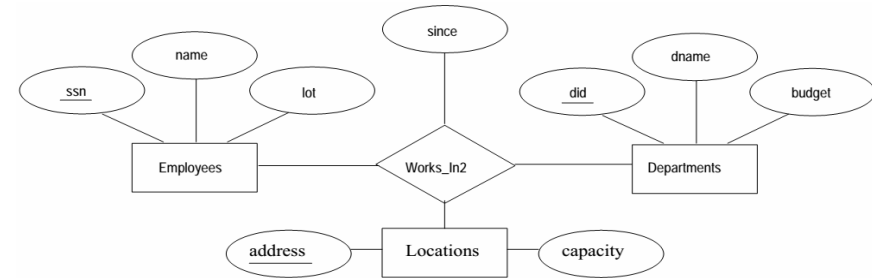
Односи



- Односи се представљају ромбовима
- и односи могу да имају описне атрибуте



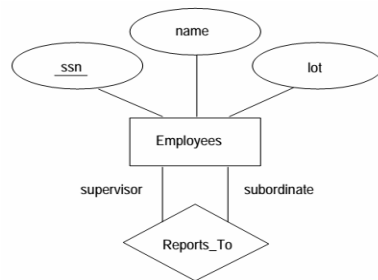
Односи



- Односи могу да укључују више ентитета
- нпр. "тернарни" односи укључују три ентитета



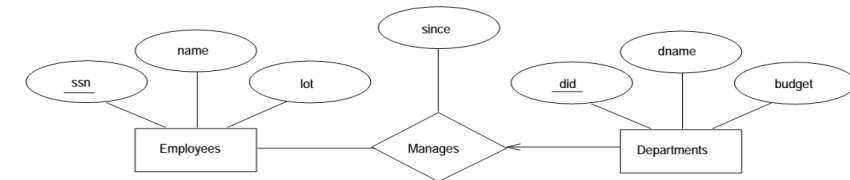
Односи



- Односи могу да буду и између више инстанци истог скупа ентитета
- У том случају морају да се именују *улоге* које ентитети имају у односима



Услови кључа

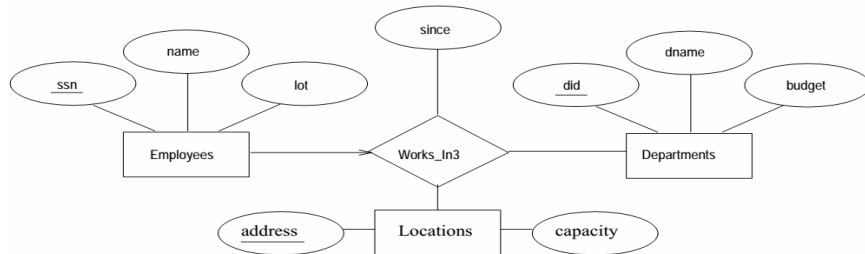


- Ако на основу ентитета може да се једнозначно одреди однос у коме учествује
- онда је то *кључни* ентитет односа
- то се означава стрелицом од ентитета према односу

- Пример: одељење има највише једног руководиоца



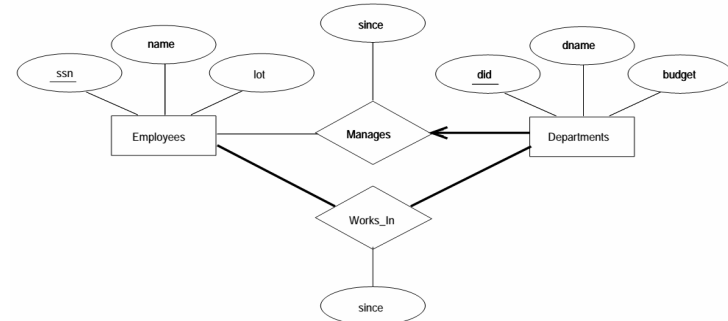
Услови кључа



•Пример: запослени ради у највише једном одељењу



Услови учешћа



- Ако сваки ентитет скупа учествује у бар једном односу, онда је то "уно учешће у односу", а иначе је "парцијално учешће"
- Пуно учешће у односу се означава дебљом линијом
- Пример:
 - сваки запослени ради у бар једном одељењу
 - свако одељење има бар једног запосленог
 - свако одељење има тачно једног руководиоца



Кардиналност

- Кардиналности односа описује у колико различитих односа тог типа може (или мора) да учествује сваки од ентитета
- Има више нотација, навешћемо изворну:
 - означава се по бројем на свакој од линија које воде од ентитета према односу
 - број означава у колико таквих односа учествује један ентитет
 - важно: то је супротно од означавања кардиналности у дијаграмима класа
 - уместо броја може да стоји опсег вредности: (А,Б) или А..Б
 - где је $A \leq B$, $A \geq 0$, $B \geq 1$
 - уместо броја може да стоји и * - означава "неки природан број"
- Кардиналност се делимично преклапа са наведеним условима
 - "бар 1" == услов учешћа, подебљана линија
 - "највише 1" == услов кључа, стрелица

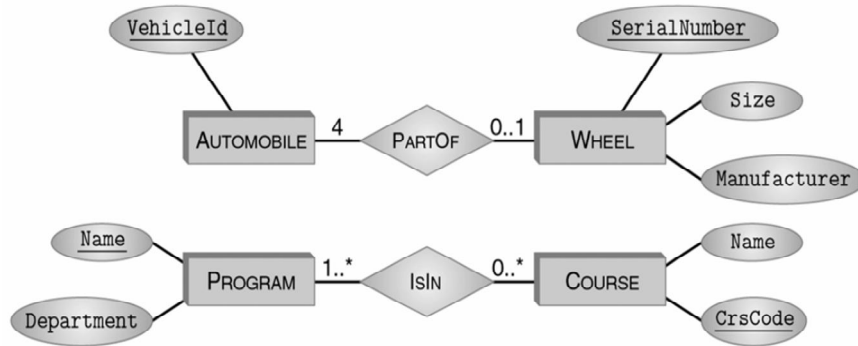


Кардиналност (2)

- Примери исправних ознака кардиналности:
 - 1
 - 4
 - 0..1
 - 2..5
 - 1..*
 - 0..*



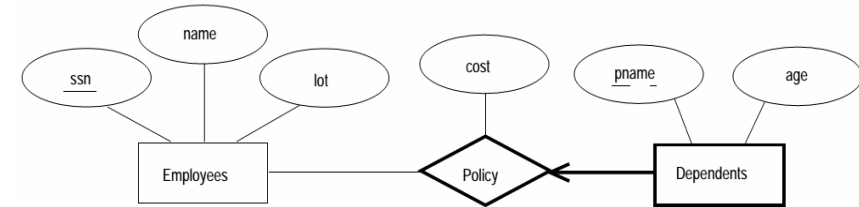
Кардиналност (3)



Универзитет у Београду - Математички факултет



Слаби ентитети



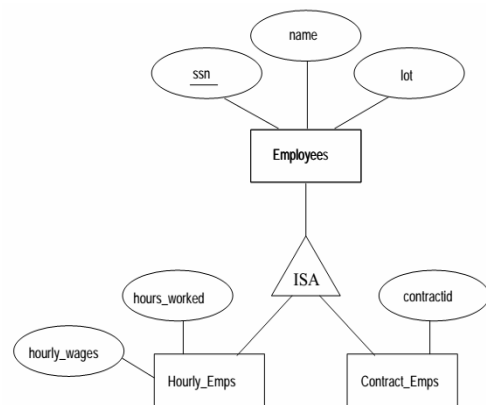
- Елементи слабог скупа ентитета се идентификују само у склопу односа са неким другим ентитетом
- тај однос мора бити "један-више" и назива се *идентификујући однос*
- слаб ентитет мора имати пуно учешће у идентификујућем односу
- и идентификујући однос и учешће слабог ентитета у њему се означавају дебљим (или двоструким) линијама

- Пример:
 - деца запосленог су осигурана преко родитеља
 - дете није у потпуности идентификовано именом, него тек односом са запосленим

Универзитет у Београду - Математички факултет



Хијерархије



• Хијерархије ентитета су тзв "ISA" хијерархије

• Општије од "наслеђивања" у OO:

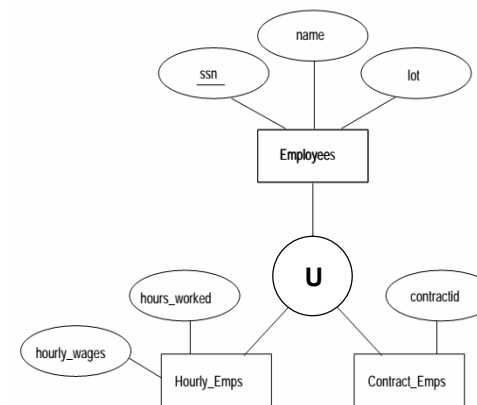
- **услов преклапања:** да ли исти ентитет може да припада *двема* поткласама или не?
- повезано са концептом вишеструког наслеђивања
- **услов прекривања:** да ли *сваки* ентитет наткласе припада некој од поткласе или не?
- повезано са концептом апстрактних класа

*Касније ћемо видети да хијерархије могу да се моделирају у релационом моделу на више начина, а избор може да зависи и од претходна два услова.

Универзитет у Београду - Математички факултет



Уније



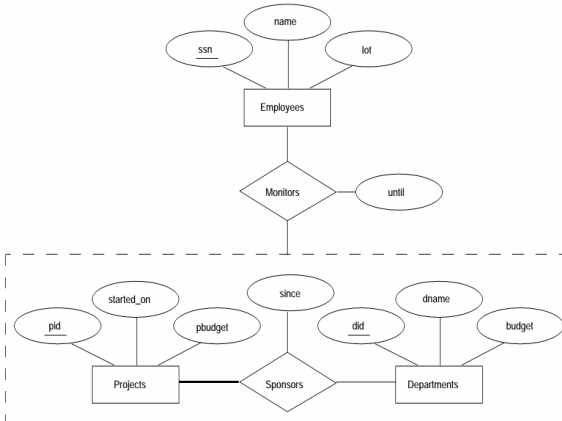
• Унија је специфична хијерархија:
 • важи прекривање
 • не важи преклапање

• У примеру: сваки запослени је или запослен по уговору или за одређен број радних сати

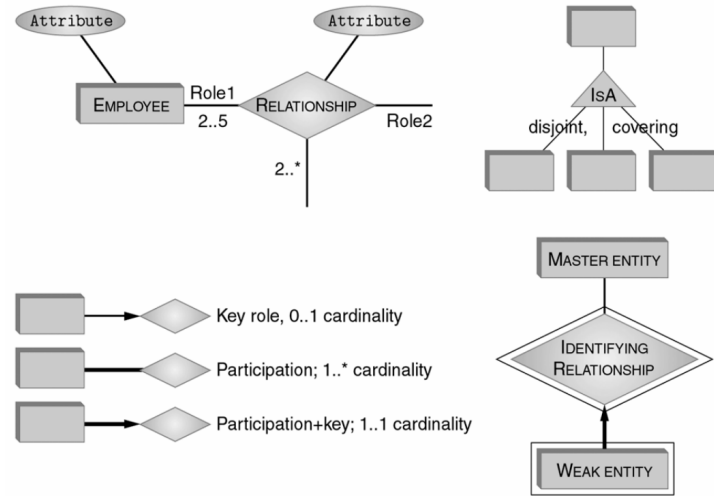
Универзитет у Београду - Математички факултет



Агрегација



- Агрегације су односи који учествују у другим односима
- Такве односе уоквирујемо испрекиданом линијом
- То није исто као тернарне релације
- Пример: пројекти могу да се спонзорису и без придружених посматрача
 - тј. такав однос је “јак” однос
- Алтернатива је да се такав однос окружи правоугаоником
- Агрегације обично имају и неке одлике ентитета



Алтернативна нотација



Основна идеја је да стрелица иде према јединствено препознатом ентитету или односу, а у случају односа 1-1 од слабијег према јачем

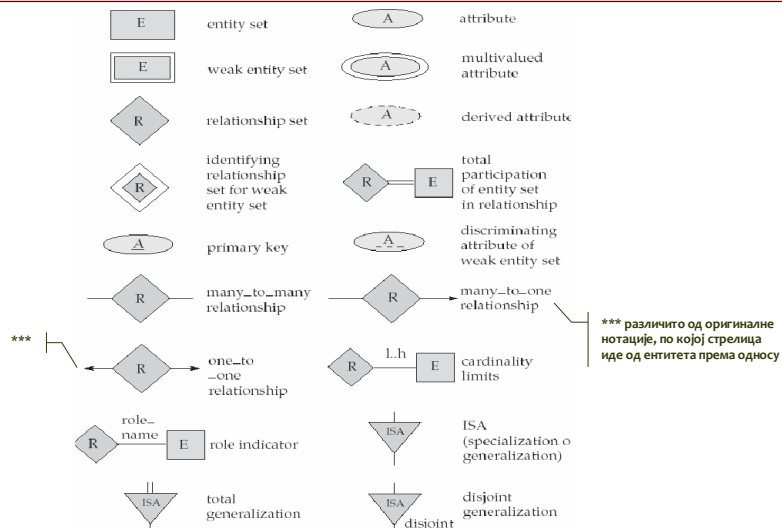
ОПРЕЗНО!!! Није уједначено, разни извори користе разне нотације...



Кардиналност

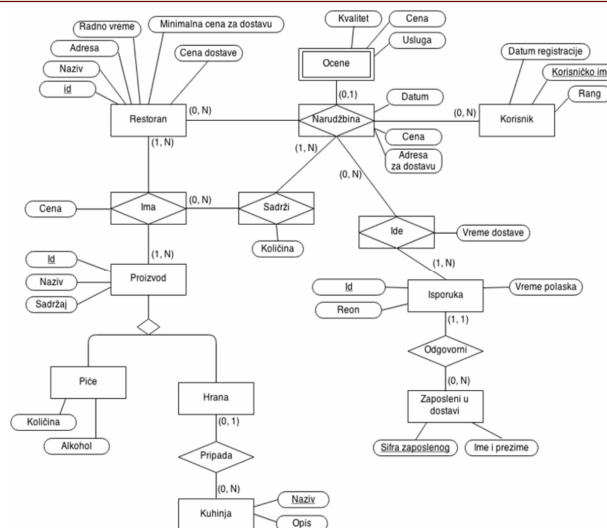
- **ВАЖНО!!!**
- У оригиналној нотацији ознаке стоје на супротним странама у односу на ознаке у дијаграмима класа у УМЛ-у!
 - У УМЛ-у број означава колико таквих ентитета учествује у таквим односима
 - у ЕР-у број означава у колико таквих односа тај ентитет може да учествује
- Неки аутори (и алати) то занемарују и раде као у УМЛ-у!
 - али то онда доводи до неусклађености бројева и графичких ознака (дебљина линије, стрелице)

Алтернативна нотација - примери

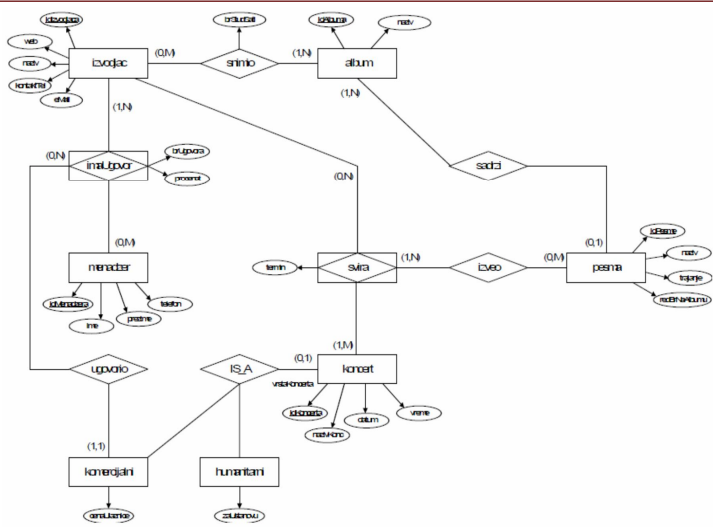


*** различито од оригиналне нотације, по којој стрелица иде од ентитета према односу

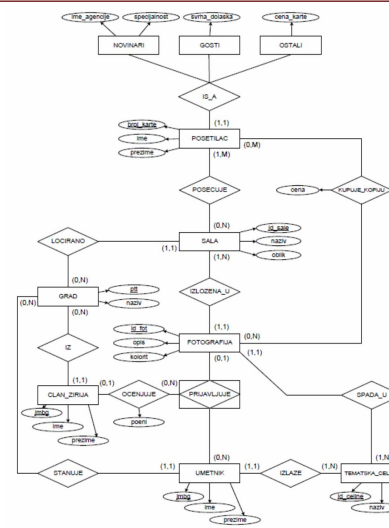
Пример: ЕР-дијаграм онлајн ресторана



Пример: ЕР-дијаграм организације концерта



Пример: ЕР-дијаграм организације изложбе





Допринос ЕР-модела

- У односу на релациони модел
 - ЕР-модел много боље описује семантичка знања о информацијама ентитетима и односима
 - Уводи нову дијаграмску технику
 - једноставно и изражајно средство комуникације
 - садржи део семантичких знања
 - Боље је прилагођен највишем нивоу посматрања информација, тј. семантичком и концептуалном моделирању



Концептуални проблем ЕР-модела

- Постоје одређене недоследности
- Почива на претпоставци да постоје два различита концепта ентитета и односа
 - али не нуди објективне критеријуме за њихово разликовање
 - начин представљање агрегација на дијаграмима додатно потврђује да односи могу да имају неке карактеристике ентитета
- Сложени атрибути су додатни проблем
 - да ли сложени атрибути имају особине ентитета?



Концептуални проблем ЕР-модела (2)

- Основни концептуални проблем ЕР-модела увиђа и сам аутор, наводећи га у фусноти рада као коментар коме не придаје посебан значај:
 - “Могуће је да неки људи виде нешто (нпр. брак) као ентитет, док неки други људи то виде као однос. Мишљења смо да одлуку о томе мора да донесе администратор предузећа. Он би требало да дефинише шта су ентитети а шта односи тако да разлика буде одговарајућа за његово окружење.”
- Сам аутор овиме признаје да су основни концепти модела само субјективно а не и суштински различити



Концептуални проблем ЕР-модела (3)

- Без објективних критеријума за њихово разликовање, два основна концепта се практично стапају у један
- Тиме се практично губи суштинска разлика између ЕР-модела и релационог модела
 - посебно у делу који се односи на нивое 2 и 3
- Свођењем разлика само на ниво 1, ЕР-модел се у теорији своди на алат за пројектовање, уместо на свеобухватни модел података
- Штавише, данас се ЕР најчешће користи чак ни као модел, него само као дијаграмска техника релационог модела
- То не умањује његов значај у домену концептуалног моделирања



Технички проблем ЕР-модела дијаграма

- Дијаграми су претрпани
 - Већ на дијаграмима са 6-7 већих ентитета дијаграми постају тешки за сагледавање
 - Основни проблем је у одвојеном представљању атрибута, што доприноси визуалној “тежини” дијаграма



Улога ЕР-модела у пројектовању БП

- ЕР-модел се највише користи у концептуалном пројектовању
 - Концептуални модел је често управо ЕР-модел
- Користи се и у логичком пројектовању, у кораку превођења концептуалног модела у логички



Основни кораци ЕР-моделирања

- Основни кораци при прављењу ЕР модела су:
 - Одређивање типова ентитета
 - Препознавање односа међу ентитетима
 - Профињавање дефиниција односа



Основни кораци (2)

- Уобичајен сценарио:
 - Полази се од текстуалног описа проблема
 - Именице представљају кандидате за ентитете
 - Опрезно, неке од њих представљају односе...



Недоумице при прављењу ЕР-модела

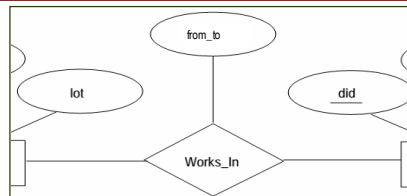
- Видели смо да је ЕР-модел добро прилагођен концептуалном моделирању
- Међутим, у пракси и ту има озбиљних недоумица:
 - Да ли је нешто боље моделирати ентитетом или атрибутом?
 - Да ли је нешто боље моделирати ентитетом или односом?
 - Који су одговарајући скупови ентитета и скупови односа? Да ли је у неком случају боље имати бинарни или тернарни однос?
 - Да ли би требало да користимо агрегације?



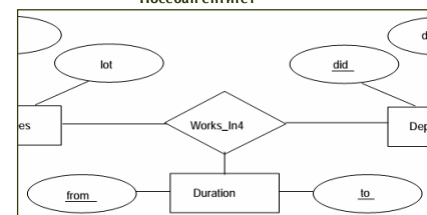
Ентитет или атрибут?

- Ако имамо сложен атрибут, можемо да га моделирамо као:
 - један сложен атрибут
 - више простих атрибута
 - један ентитет
- На основу чега доносимо одлуку?
- Чак и за просте атрибуте може да се постави питање да ли су они можда ипак ентитети?
- Пример: адреса...

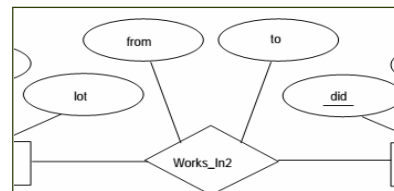
Ентитет или атрибут?



•Један сложен атрибут



•Посебан ентитет



•Два проста атрибута



Ентитет или атрибут? (3)

- Аргументи за “обичан атрибут”:
 - ако једном ентитету одговара тачно једна вредност “атрибута”
 - ако представља једноставну скаларну вредност која описује ентитет
 - ако у више ентитета може да има исте вредности, а да оне нису међусобно условљене
 - тј. мењање вредности за један ентитет не повлачи њено мењање за други
 - ако важи слично и када је то “атрибут” у различитим скуповима ентитета



Ентитет или атрибут? (4)

- Сложен атрибут може да има смисла:
 - ако желимо да сачувамо интерну структуру, али она има значаја само интерно
 - ако у више ентитета може да има исте вредности, а да оне нису међусобно условљене
 - нпр, ако “адреса” обухвата град, улицу и број, онда више запослених може да има исту адресу, а да то не значи да живе заједно, па оне и даље могу да се одвојено мењају
 - ако важи слично и када је то атрибут у различитим скуповима ентитета

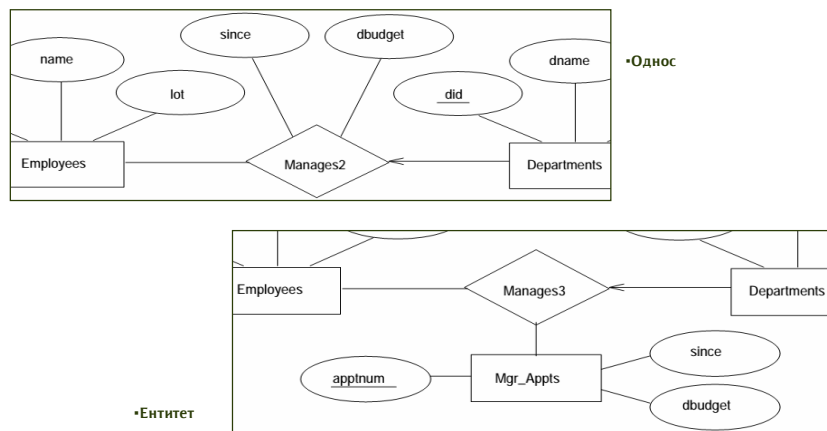


Ентитет или атрибут? (5)

- “Атрибут” мора да прерасте у ентитет:
 - ако неком ентитету одговара истовремено више вредности тог “атрибута”
 - ако постоји међузависност вредности “атрибута”
 - нпр. ако више ентитета има исту вредност атрибута и када се он промени за један ентитет, онда мора да се промени и за остале



Ентитет или однос?



Ентитет или однос? (2)

- Однос може да буде ”однос”:
 - Ако се сви његови атрибути односе стрикно на једну инстанцу односа
 - тј. нема редундантности
- Однос мора да прерасте у “ентитет”:
 - Ако се неки од његових атрибута односи истовремено на више инстанци односа
 - редундантност повлачи уочавање/издвајање новог ентитета и подизање сложености односа
- Иначе?
 - У великом броју случајева је “једнако исправно” да нешто буде било однос било ентитет

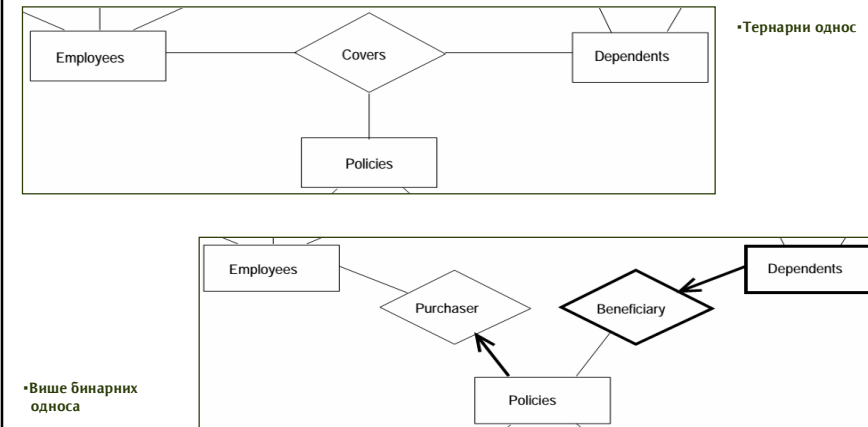


Тернарни однос или више бинарних односа?

- Већина сложених односа може да се “подели” на више бинарних агрегираних односа
- Да ли је боље поделити их или их оставити?



Тернарни однос или више бинарних односа?



Тернарни однос или више бинарних односа?

- Сложене односе има смисла поделити на више једноставнијих *искључиво* ако сваки од добијених односа има свој семантички смисао у посматраном домену
- Чак и тада, ако је семантика очигледнија из сложеног односа, онда би тај сложени однос требало да остане

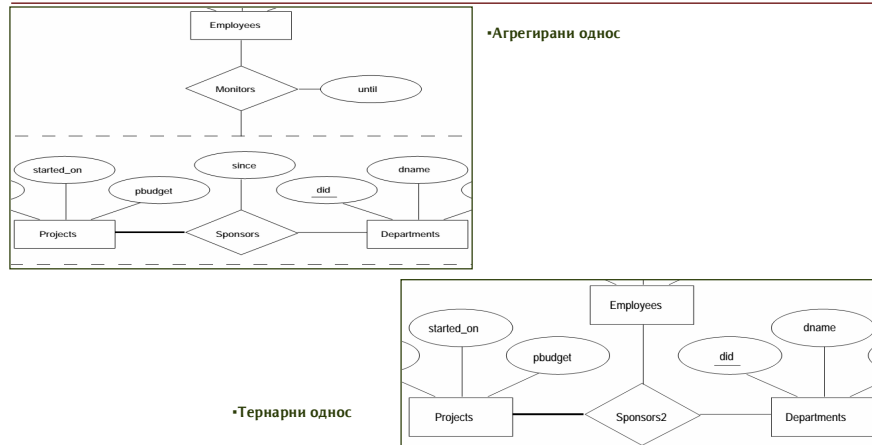


Агрегација или тернарни однос?

- Ово питање је слично претходном:
 - Агрегирани однос подразумева да се агрегирани однос може посматрати као скуп ентитета (или скуп односа)
 - Но, то може да буде и вештачки направљено?
- Шта је и када боље урадити?



Агрегација или тернарни однос?



Агрегација или тернарни однос?

- Када су агрегирани односи боље решење?
 - Ако некада постоји само део односа, а некада цео (тј. некада само део који се агрегира, а некада пун однос)
 - Ако се део атрибута односа не тиче целине односа већ само његовог дела, који укључује мање ентитета
- У осталим случајевима обично је боље да се користе сложени односи

[PM13]
Увод у РБП
Саша Малков



Тема 6.2

Концептуално пројектовање

-

УМЛ дијаграм класа домена



УМЛ и моделирање база података

- Мотивација и циљ:
 - УМЛ је опште познат језик моделирања софтвера
 - приближавање језика за моделирање података језику за моделирање програма
- У више наврата је предлагано да се направи посебан „профил“ за моделирање података
 - Незваничне радне верзије 2001-2005, али без званичних корака након тога
 - Заправо није неопходно – сасвим добро је и без тога



Модел класа података

- УМЛ дијаграм класа већ описује структуру података класа
- Идеја је да се исти дијаграм искористи за моделирање података:
 - боље представља семантику односа од дијаграма табела
 - боље представља семантику односа чак и од правог ЕР дијаграма
 - има мање недостатке који могу да се превазиђу

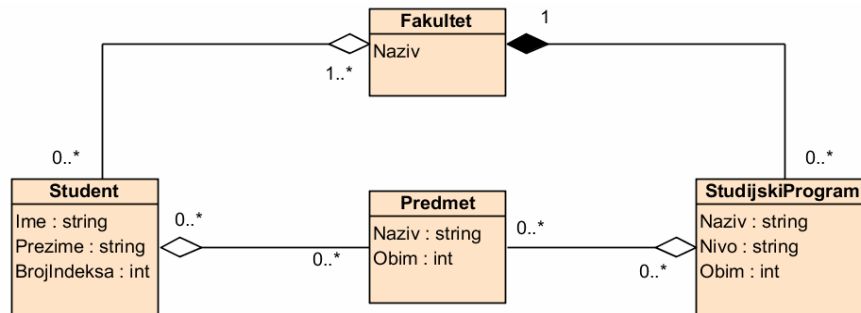


УМЛ дијаграм класа података

- Користимо тзв. *концептуални дијаграм класа*
 - назива се и *дијаграм класа домена*
 - када се користи за БП назива се и *дијаграм класа података*
- За разлику од обичног дијаграма класа:
 - у првом плану су атрибути и односи
 - не представљају се ни понашање ни енкапсулација
- За специфичне ознаке користе се различите допуне УМЛ-а



Пример дијаграма класа података



Разлика у концепту моделу (1)

- Модели класа и релација се концептуално разликују:
 - релације (табеле) су *скупови* података
 - класе су *типови* података



Разлика у концепту модела (2)

- Може да се превазиђе у пракси:
 - а) ако је потребно више скупова истог типа, уводимо више наследника класе која одређује тип
 - није сасвим семантички исправно али експлицитно се види „сличност“ типова појединачних скупова ентитета
 - базна класа, која служи само као тип, може да се посебно означи
 - нпр. као “апстрактна”
 - б) други приступ је да се свака “класа” означи као “*«type»*” или “*«persistent»*”
 - типови/класе се означавају са “*«type»*”
 - скупови ентитета се означавају са “*«persistent»*”
 - в) проблем се игнорише и сматра да је класа *скуп* а не *тип*
 - у реду је само за неке домене, где се не очекује више скупова истог типа



Допуне УМЛ-а

- УМЛ садржи стандардизоване концепте који омогућавају увођење нових начина означавања:
 - стереотипови
 - означене вредности
 - проширења



Стереотипови у УМЛ-у

- Стереотип представља врсту *шаблона*:
 - апстракцију општијег случаја нечега
 - неки предефинисани скуп особина
 - неко предефинисано понашање
- Користе се за навођење додатних декларација или напомена

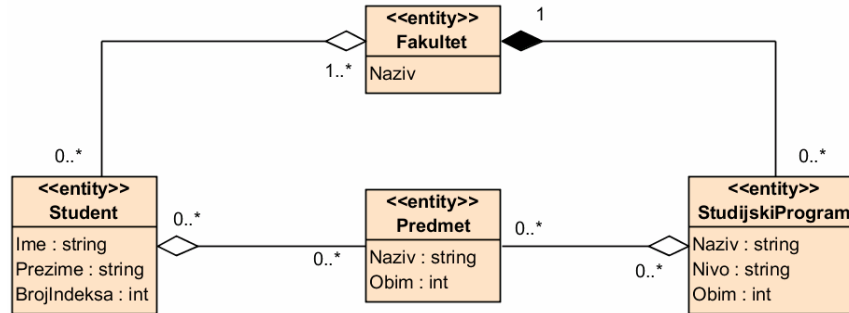


Означавање стереотипова

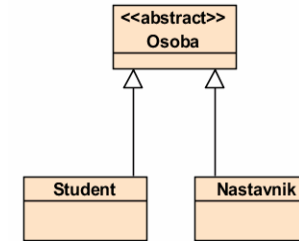
- Подразумевано означавање је навођењем назива између двоструких изломљених заграда:
 - `<<entity>>`
 - `<<abstract>>`
 - `<<type>>`
 - `<<persistent>>`
- Обично се наводи испод или изнад назива класе (или објекта)
 - може и испред



Означавање стереотипова



Означавање стереотипова



Експлицитно означавање да је потребно да важи услов прекривања

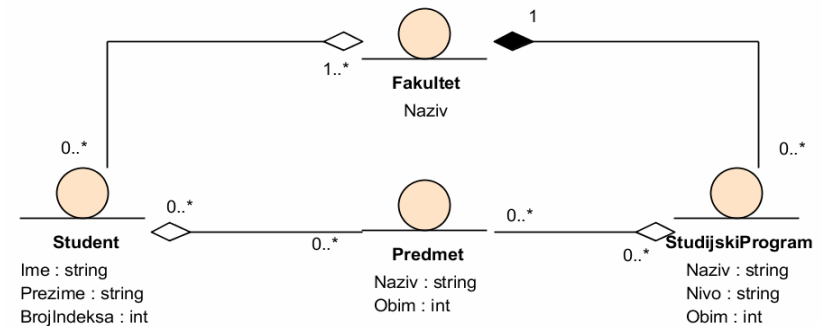


Означавање симболима

- За неке уобичајене стереотипове касније су уведене и графичке ознаке
- Оне обично стоје као допуна у десном горњем углу уобичајеног симбола за ту врсту објеката
- Могу чак и да замене основни графички симбол неке врсте објеката
 - ако стереотип представља значајну поткласу те врсте објеката, на пример „актер“ или „перзистентни“



Примери стереотипова





Означене вредности

- Општији случај стереотипова су означене вредности
- Наводе се као листа парова име-вредност између витичастих заграда:
 - { isPersistent= true, minCount=5 }
- У начелу, стереотип је еквивалентан означеној тачној вредности одговарајућег мета-атрибута:
 - <<abstract>> == { isAbstract=true }



Концепт проширења

- *Проширење* је елемент који се повезује са основним елементом стрелицом са попуњеним троуглом на крају
 - стрелица је усмерена од основног елемента ка проширењу
- Може да се користи за додавање особина или за описивање основног елемента
- Обично је опционо, али може да буде и *неопходно*, што се означава са „*required*” изнад стрелице



Односи

- Асоцијација
- Агрегација
- Композиција
- Наслеђивање

у основи није друкчије него код уобичајених дијаграма класа



Асоцијација

- Однос између две класе
- Означава да бар једна од класа „зна“ за неке објекте друге класе и на неки начин управља њима
- Може да буде:
 - *функционална* – „уради нешто за мене“
 - *структурална* – „буди нешто за мене“
- У контексту модела података *структурална* је важнија



Агрегација

- Посебна врста асоцијације
- Имплицира да се једна класа “*састоји*” од (једног или више) објеката друге класе
- Представља слабији облик структуралне асоцијације:
 - исти део може да буде “*садржан*” у више сложених објеката
 - ако се сложени објекат обрише, делови могу да наставе да постоје
- Понекад се практично свака структурална асоцијација, чија је кардиналност “1-више”, посматра као форма агрегације
 - али то није добро радити ако семантика није одговарајућа



Композиција

- Јачи вид агрегације
- Представља јачи облик структуралне асоцијације:
 - увек је бинарни однос
 - однос целина/део
 - један део може да припада само једном сложеном објекту
 - ако се сложени објекат обрише уобичајено понашање је да се обришу и сви делови



Наслеђивање

- Однос који је суштински за ОО моделирање, па тиме и за модел класа података
- Основна класа представља општији случај изведених класа
 - “*генерализација*”
- Изведене класе представљају посебне случајеве базне класе
 - “*специјализација*”
- (Релациони модел нема, а модел ентитета и односа има одговарајући концепт)



Навигација и идентификација (1)

- Модел класа података подразумева да се за реферисање других објеката користе неки видови јединствених идентификатора, тзв. *OID*
 - то начелно одговара концепту примарног кључа



Навигација и идентификација (2)

- Већина ООБП подразумева да идентификатор објекта нема никакво семантичко значење и његова вредност често чак не може ни да се види или промени
 - такви кључеви идентификују *променљиве*, а не податке
 - зато то строго формално није у складу са релационим моделом података
- * *ово се не односи на све врсте сурогатних кључева, о томе ће бити више речи групом ирилично*



Навигација и идентификација (3)

- Ако не прејудуцирамо имплементацију, онда можемо да не наводимо сурогат кључеве
 - такав приступ је сасвим прихватљив, штавише пожељан, на концептуалном нивоу
 - прихватљив је и на логичком нивоу
 - све док не преведемо пројекат у релациони модел
 - није прихватљив на физичком нивоу



Односи у моделу података

- У моделу класа података већина односа има структуралну природу
 - због тога обично већина асоцијација може да прерасте у агрегацију, или чак композицију
 - али не увек!!!



Асоцијација или агрегација?

- Да ли *факултет* садржи *студенте*?
 - суштински можемо да одговоримо: “не”
 - студенти *нису* физички део факултета
 - на нивоу података можемо да одговоримо: “*га*”
 - подаци о *уписаним* студентима *јесу* део података о факултету
 - не постоји студент који није везан за неки факултет
 - у терминима ЕР, студент је слаб ентитет
- Да ли *студент* садржи *предмете*?
 - суштински “не”
 - на нивоу података “*га*”, али “*уписане предмете*”
- У оба случаја можемо да користимо агрегацију
 - факултет *има* скуп уписаних студената
 - студент *има* скуп уписаних предмета
- Али чак и композицију
 - Студент није студент ако није на неком факултету
 - Уписан предмет није уписан ако не одговара неком студенту



Асоцијација или агрегација? (2)

- Да ли је грешка ако се користи агрегација:
“*факултет* садржи *студент*”?
- Не! Али само ако се прецизира да су то *уписани* студенти.
- Да ли је грешка ако се ту не користи агрегација?
- Не!
- Па шта је онда боље?
- Зависи од контекста...
 - ако правимо ИС факултета, боље је са агрегацијом
 - ако правимо нешто сасвим друго, где је информација о студијама *сјоредна* а не *јримарна*, онда може да буде боље да користимо асоцијацију
- Основни критеријум је разумљивост
 - Шта ће неко ко чита дијаграм лакше и боље да разуме?



Односи у моделу података (2)

- Исте односе често можемо да моделирамо *исправно* на различите начине
- Избор често зависи од нивоа модела:
 - на концептуалном нивоу, циљ је да очувамо семантику односа из домена проблема
 - на логичком моделу, циљ је да остваримо стабилну и једнозначну структуру
 - на физичком нивоу, циљ је (поред претходног) да омогућимо добре перформансе



Разлике између нивоа моделирања

- Разлике у начину представљања модела на концептуалном / логичком / физичком нивоу сличне су као код дијаграма табела
- Концептуални дијаграм не мора да садржи (и обично не садржи):
 - додатне колоне сурогат кључева
 - додатне колоне страних кључева
 - ознаке кључева (осим можда једног кандидат-кључа)
 - типове колоне
- Логички и физички дијаграм садрже и:
 - колоне сурогат кључева (ако су потребне)
 - колоне везаних табела који чине стране кључеве
 - ознаке кључева и врста кључева
 - типове колоне

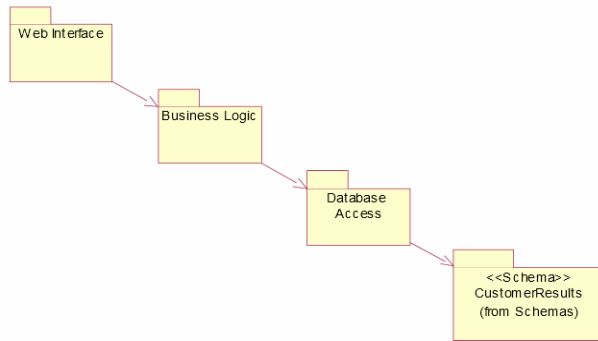


Груписање

- На концептуалном нивоу
 - табеле се групишу у схеме
 - класе се групишу у пакете
- На физичком нивоу
 - табеле се
 - групишу у просторе за табеле
 - рапоредују на чворове (сервере)
 - (функционалне) класе се
 - групишу у компоненте
 - распоредују на чворове (сервере)



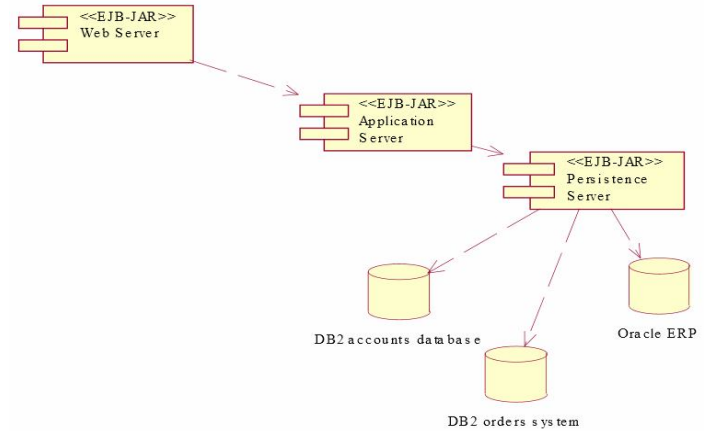
Концептуално груписање



Универзитету Београду - Математички факултет



Физичко груписање



Универзитету Београду - Математички факултет



Представљање погледа

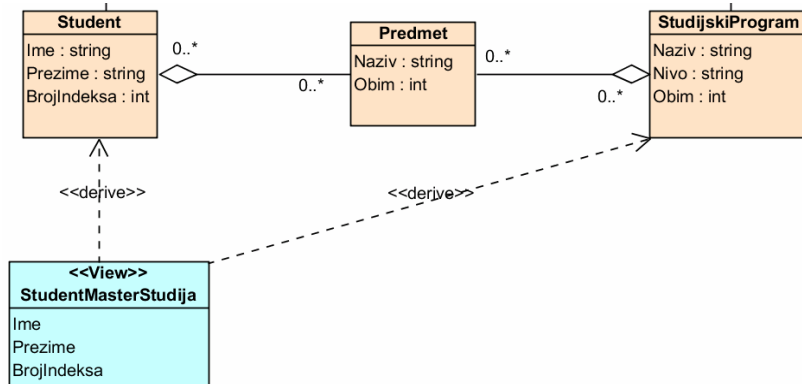


- Погледи се представљају са ознаком стереотипа у углу
 - може и симбол
 - или друга боја
 - уводи се и однос зависности према базним класама/табелама

Универзитету Београду - Математички факултет



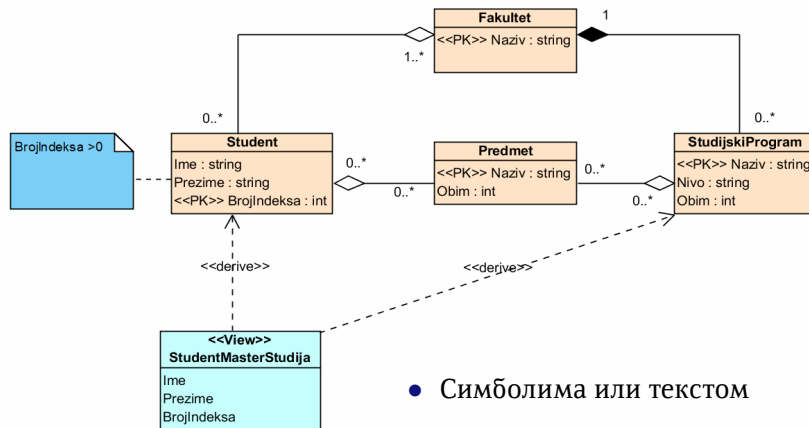
Представљање погледа (2)



Универзитету Београду - Математички факултет



Означавање кључева

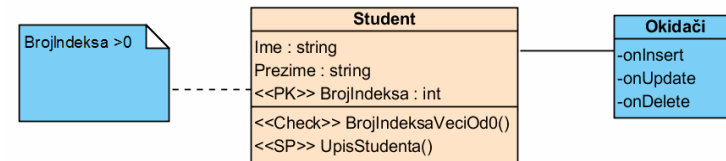


- Симболима или текстом

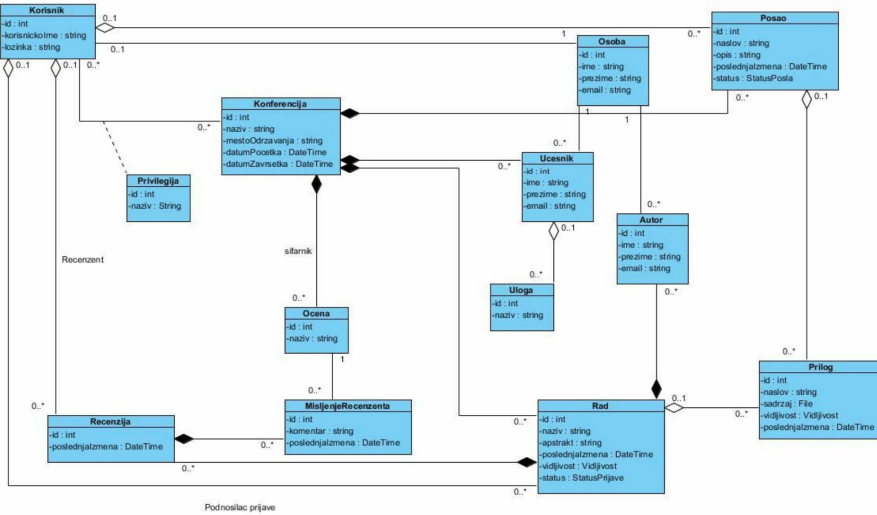


Услови интегритета, индекси,...

- Разне друге ствари:
 - услови интегритета
 - индекси
 - окидачи
 - серверске процедуре
 - ...
- Могу да се наводе у оквиру класа у одељку понашања
 - али већина алата не пружа такву слободу
- Као повезане “класе” или коментари



Пример 1 – Систем за подршку организовања научних скупова



Литература за тему

- Гордана Павловић-Лажетић, *Увод у релационе базе података, 2. изд. Математички факултет, 1999.*
 - доступно онлајн: <http://poincare.matf.bg.ac.rs/~gordana/urbp-2016.htm>
- Ramakrishnan, Gehrke, *Database Management Systems, 2.ed, 2000.*
- Teorey, Lightstone, Nadeau, Jagadish, *Database Modeling and Design, 5.ed, Elsevier, 2011.*
- Muller, *Database Design for Smarties – Using UML for Data modeling, Academic Press (1999)*
- Alvaro Monge, *Database design with UML and SQL, 4.ed.*
 - <https://web.csulb.edu/colleges/coe/cecs/dbdesign/dbdesign.php>
- Алати за моделирање, препоручује се:
 - Visual Paradigm Suite, бесплатна верзија *Community Edition*
 - <http://www.visual-paradigm.com/>