

Увод у релационе базе података

9



Саша Малков
Универзитет у Београду
Математички факултет
2023/2024

[PM13]
Увод у РБП
Саша Малков



Тема 6.3

Концептуално пројектовање

– Појединачни концептуални модели

[PM13] Увод у релационе базе података – Саша Малков – 2023/24 – час 9

1

Концептуално пројектовање

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 – 2023/24 – час 9

2

Концептуално пројектовање / Појединачни концептуални модели

Анализа захтева



- Анализа захтева
 - Веома важна!
 - Веома захтевна!
- Тесно повезана за процесом анализе захтева у контексту пројектовања информационог система
- Да би се разумеле потребе за подацима често морају да се детаљно разумеју сви процеси који се одвијају у систему

Универзитет у Београду - Математички факултет

[PM13] Увод у релационе базе података – Саша Малков – 2023/24 – час 9

3



Циљеви анализе захтева

- Основни циљеви:
 - Превођење функционалних захтева у контекст трајних података
 - Разумевање и описивање података и њихове улоге
 - скупова потребних података и њихових односа
 - структуре потребних података
 - свих појединачних трансакција које се извршавају над подацима
 - Дефинисање нефункционалних захтева
 - интегритет
 - перформансе
 - безбедност
 - и разни административни аспекти
 - ...



Циљеви анализе захтева (2)

- Основни циљеви:
 - ...
 - Одређивање и обликовање додатних услова имплементације
 - технологије
 - хардверске и софтверске претпоставке
 - апликативни и кориснички интерфејси
 - и друго
 - Израда исцрпне документације за све наведено



Циљеви анализе захтева (3)

- Један од најважнијих циљева анализе захтева, која се врши у оквиру концептуалног пројектовања, је **декомпозиција система на скуп слабије повезаних делова / сегмената**
 - Идеја је да се пројектовање система подели на мање целине, које се лакше изучавају и моделирају
 - Касније, у фази интегрисања, ти делови ће се спојити у целину



Концептуално моделирање података

- Надаље се у овом кораку бавимо концептуалним моделирањем једног дела система
 - углавном на сличан начин као што се у традиционалном приступу моделирао цео систем
- Главни кораци
 - Класификација скупова података
 - препознавање ентитета / класа и атрибута
 - Препознавање односа
 - генерализација / специјализација и хијерархије
 - асоцијације и сложенији односи



Класификација скупова података

- Свака именица је кандидат за ентитет или атрибут
 - није увек очигледно шта је ентитет (или класа) а шта атрибут
- Неке смернице
 - Ентитети би требало да садрже описне информације
 - Ако нешто има вишеструку или сложену вредност (а не скаларну) онда је вероватно ентитет
 - Атрибути би требало да припадају оним ентитетима које најнепосредније описују



Садржај ентитета

- Ентитети садрже описне информације
 - ако имамо описне информације о неком податку, онда би то требало да буде ентитет
 - “предмет” има “назив” и “број ЕСПБ”, па би требало да буде ентитет
 - ако неки податак има само идентификатор (који може да буде описног карактера али нема додатних описа) онда би то требало да буде атрибут
 - “оцена” се идентификује својом вредношћу и нису потребни додатни описи, па је то атрибут



Садржај ентитета (2)

- Пример: “град”
 - ако нам је сасвим довољан назив града, онда може да буде атрибут
 - ако нас занима и локација града или број становника, онда је то ентитет



Вишеструке вредности

- Ако нешто може да има више вредности, сложену вредност, или вредност која је подложна променама, онда је то ентитет
 - то некада не мора да буде неопходно, али је једноставније за анализу и разматрање, а касније може да се оптимизује ако је потребно
- Пример:
 - ако нпр. “удружење студената” има само назив, онда је то атрибут
 - али ако један студент може да буде члан више удружења, онда је боље да се удружење прогласи за ентитет
 - такође, пошто више студената може да буде у истом удружењу, евентуална промена назива удружења може да буде проблематична ако је у питању само атрибут...



Додељивање атрибута ентитетима

- Атрибути се додељују оним ентитетима које најнепосредније описују
 - То некада није сасвим очигледно
 - Обично може да се установи из функционалних зависности
 - ако се погрешно, може да се поправи у фази логичког пројектовања
 - Некада одговарајућа зависност постоји само у домену али не и у подацима
 - тј. вредност атрибута се у начелу одређује на основу ентитета, али не може да се установи функционална зависност
 - обично се ради о (потенцијално прикривеним) транзитивним зависностима



Препознавање односа

- За сваки однос је потребно препознати
 - учеснике
 - и ред односа (тј. број учесника)
 - кардиналност
 - и опционост / обавезност
 - додатне атрибуте који описују однос
 - јасан назив и семантику односа у домену
 - као и одговарајуће функционалне зависности



Препознавање генерализације

- Генерализација = Сваки “посебан” случај “је” истовремено и “општи” случај
- Установљавање смера генерализација није увек очигледно
- У случају ОО дизајна основни критеријум је понашање
 - квадрат је специјалан случај правоугаоника
- Код база података смо често у искушењу да игноришемо понашање и посматрамо само структуру
 - структура правоугаоника је допуна (тј. специјалан случај) структуре квадрата



Препознавање генерализације (2)

- При грађењу хијерархија ипак не сме да се потпуно запостави понашање
- Фокусирање на структуру “може да прође” само у случају плитких хијерархија
 - У случају дубљих хијерархија се испољавају неугодне последице погрешних одлука



Препознавање генерализације (3)

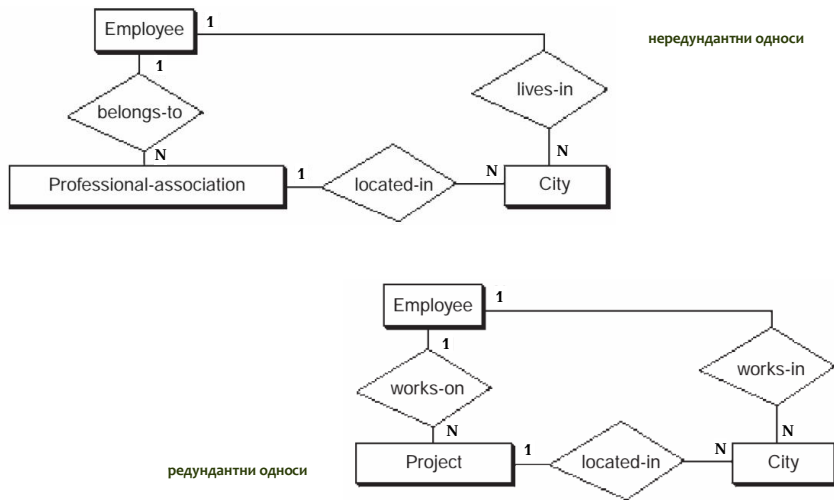
- Што апстрактније – то боље:
 - Атрибуте је потребно “попети” што више уз хијерархију
 - Кључне атрибуте је потребно везати за базу класу (ентитет)
 - Односе је потребно “попети” што више уз хијерархију



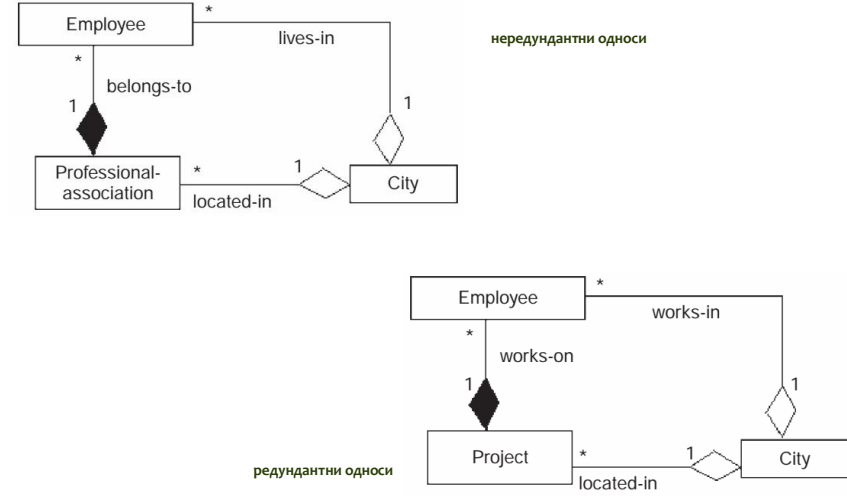
Редундантност односа

- Редундантни односи
 - имају за резултат тешко отклоњиву редундантност у моделу и имплементацији
 - ометају конзистентност и нормализацију
- Потребно је елиминисати редундантне односе
- ОПРЕЗ!
 - “исти” односи могу бити редундантни и нередундантни
 - редундантност зависи од семантике и домена

Редундантни односи, пример 1, ЕР



Редундантни односи, пример 1, УМЛ





Односи вишег реда

- Однос је “вишег реда” ако обухвата више од два учесника
- ОПРЕЗНО са односима вишег реда
 - увек је потребно да се додатно размисли да ли су неопходни
 - врло често је боље (и семантички и технички) да се однос вишег реда подели на више једноставнијих односа
- Упутство:
 - Увек покушати са више бинарних односа
 - Ако се испостави да је немогуће, тек онда користити односе вишег реда



Односи вишег реда, пример

- На пример:
 - У распореду часова, сложен однос је “наставник А држи час из предмета Б у учионици В у термину Г”
 - Међутим, однос наставника и предмета је важан и независно од распореда и издваја се у концепт групе
 - “група Д” = “наставник А држи предмет Б”
 - може да има и додатне атрибуте, нпр. ознаку
 - може више наставника да држи различите групе за исти предмет
 - може да има и однос са студентима који чине групу
 - почетни однос је онда једноставнији (иако и даље сложен)
 - “група Д има час у учионици В у термину Г”

[PM13]

Увод у РБП

Саша Малков



Тема 6.4

Концептуално пројектовање

-

Обједињени концептуални модел



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



Раздвојени погледи

- Концептуално моделирање се често одвија по деловима
 - одвојено за различите апликације
 - или делове апликације
- Основна мотивација:
 - анализа и моделирање мањих делова домена се изводе једноставније и ефикасније
 - мањи модели се лакше обликују и разумеју
- Сваки појединачан модел се назива “погледом” или “локалном схемом”
 - нема непосредног додира са “погледима” у језику SQL
 - мада се они користе за имплементацију



Раздвојени погледи, пример

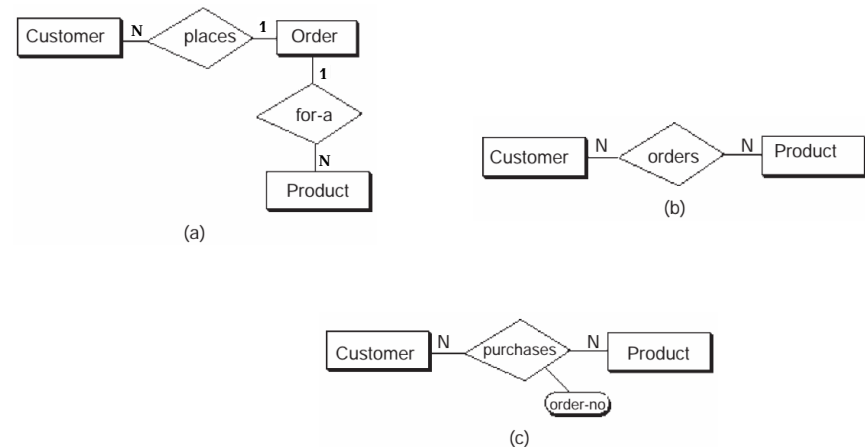
- ИС факултета
 - из угла студента
 - потребно је да се зна да је неки наставник задужен за неки предмет
 - није битно када је ко изабран у које звање, па ни у ком је звању
 - из угла наставника
 - који студенти су уписали предмет, испунили обавезе, пријавили и полагали испит
 - није битно финансијско стање, нису битни подаци о другим наставницима
 - из угла кадровске службе
 - није битно ко је задужен за који предмет
 - важно је када је ко изабран у које звање, када је био на одсуствима



Интегрисање погледа (локалних схема)

- Општа (глобална) концептуална схема се добија интегрисањем локалних схема (погледа)
- То је некада једноставно али може да буде и веома сложено
- Зато се *интегрисање погледа* често описује као посебан корак
 - иако је заправо саставни део концептуалног моделирања
- ВАЖНО: Термини “локална схема” и “глобална схема” у контексту пројектовања имају другачији смисао него у контексту дистрибуираних база података.

Пример 2, различити модели за исти домен





Кораци при интегрисању погледа

- Основни кораци:
 - Поређење схема и препознавање конфликта
 - Преуређивање схема и разрешавање конфликта
 - Спајање и реструктурирање схема



Препознавање конфликта

- Врсте конфликта
 - конфликти имена
 - структурни конфликти
 - конфликти кључева
 - конфликти зависности



Конфликти имена

- Неуједначено именовање ентитета и односа
 - речници ентитета и односа морају бити уједначени
- Синоними
 - различити називи за исте концепте
 - први и најлакше решив проблем
- Хомоними
 - исти назив се користи за различите концепте
 - може да буде велики проблем ако је различито тумачена спецификација захтева
 - теже решиво од синонима и зато има приоритет при решавању



Структурни конфликти

- Различити структурни елементи се употребљавају за моделирање истих концепата
 - (пример 2 са наруцбеницама)
- Да би погледи могли да се споје, одговарајући концепти морају да имају идентичне структуре



Конфликти кључева

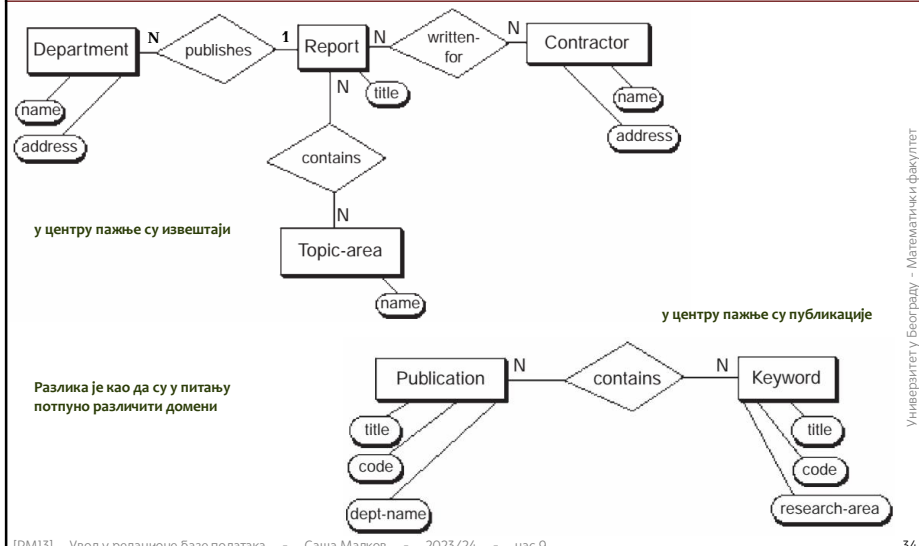
- Истом ентитету су у различитим погледима додељени различити кључеви
- Неопходно је да се кључеви уједначе
- У супротном прети опасност од редундантних кључева



Конфликти зависности

- Исти односи се у различитим погледима моделирају на различите начине
 - различите врсте односа (слаби и јаки ентитетски односи, агрегација или асоцијација, и слично)
 - различите кардиналности
 - могу да буду и суштински различити односи
- Решавају се уједначавањем карактеристика односа
 - у неким случајевима и додавањем нових односа

Пример 3.1, већи концептуални конфликт



Разрешавање конфликта

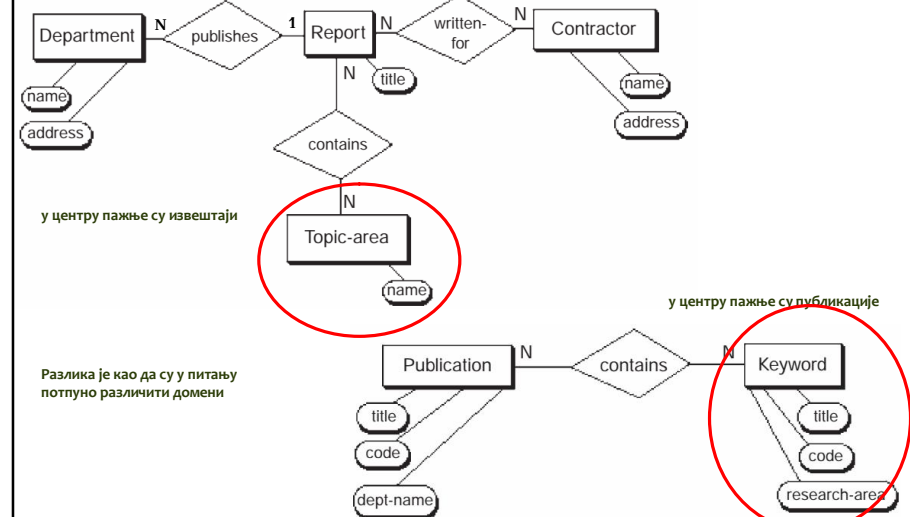
- Може да захтева додатну или поновљену анализу захтева
- Може да захтева озбиљно модификовање погледа
- Добро је да учествују пројектанти конфликтних погледа
- Технике разрешавања конфликта су исте као и за прављење концептуалног модела



Разрешавање конфликта (2)

- На пример, у моделу 3.1 се у два погледа користе различити термини за исту ствар: *Topic-area* и *Keyword*
- Штавише, имају и различите атрибуте
- ...

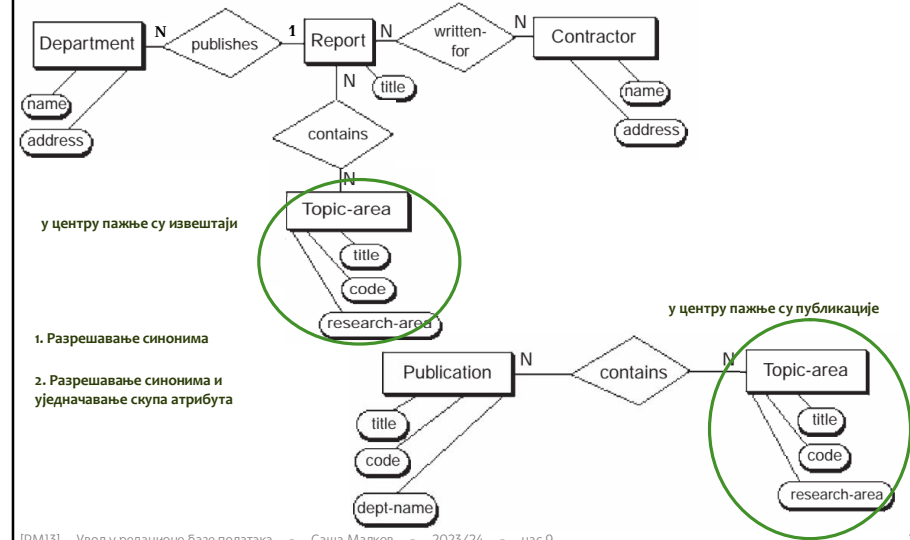
Пример 3.1, већи концептуални конфликт



Разрешавање конфликта (3)

- ...
- У моделу 3.2 (на наредној страни) је овај конфликт решен:
 - у оба модела се користи назив *Topic-area*
 - задржани су називи атрибута из модела публикација

Пример 3.2, већи концептуални конфликт

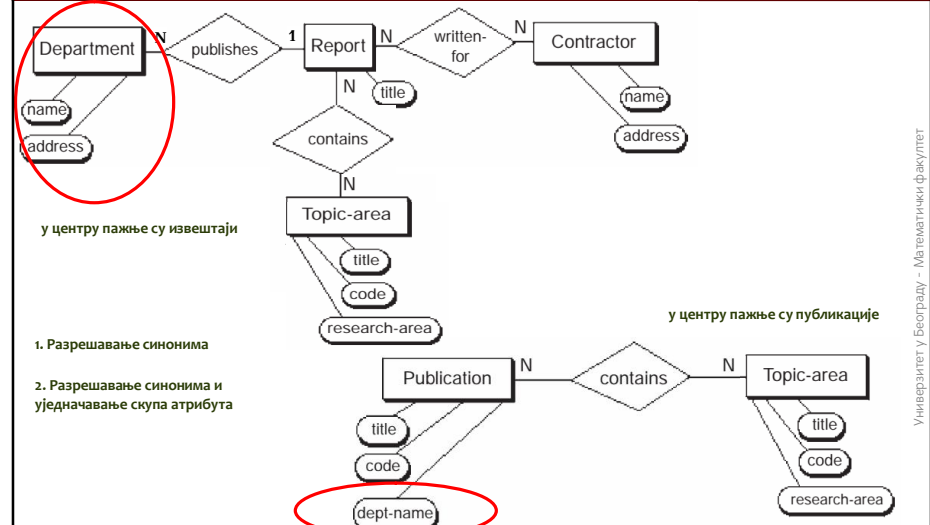




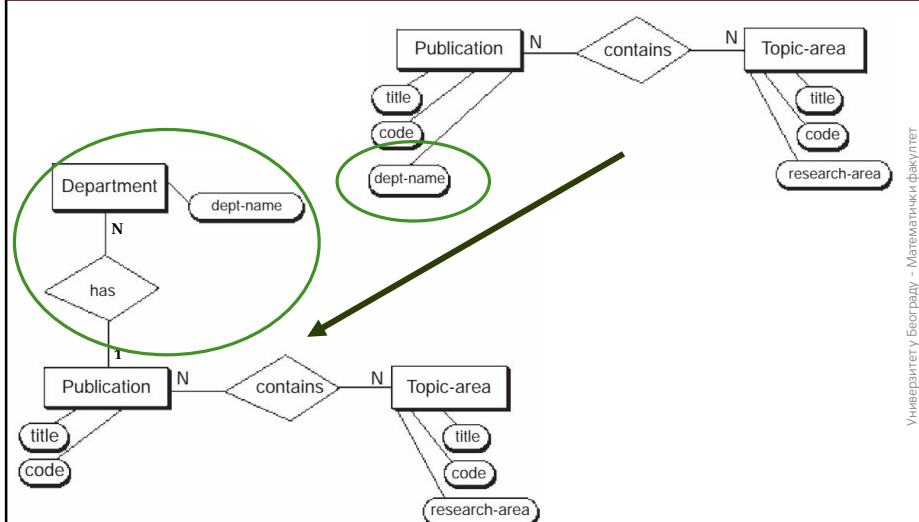
Разрешавање конфликта (3)

- Даље, видимо да се у публикацији департман помиње само кроз атрибут *dept-name* а у извештајима представља ентитет
- То је структурни конфликт – потребно је да се уведе ентитет и уз публикације

Пример 3.2, већи концептуални конфликт



Пример 3.3, већи концептуални конфликт



Спајање и реструктурирање схема

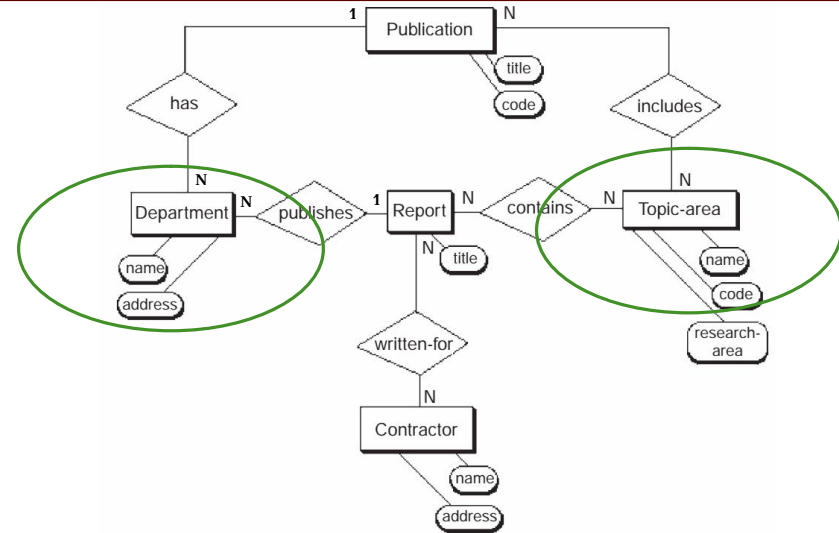
- Наступа након усаглашавања конфликта
- Руководи се принципима
 - **потпуности** – сви концепти из локалних схема морају очувани и у потпуности да се пренесу у глобалну схему
 - **минималности** – сви редундантни концепти (и ентитети и односи) морају да се уклоне из глобалне схеме
 - **разумљивости** – глобална схема мора да буде разумљива свим корисницима



Принцип потпуности

- Након разрешавања конфликта најпре спајамо све одговарајуће концепте и остварујемо потпуност

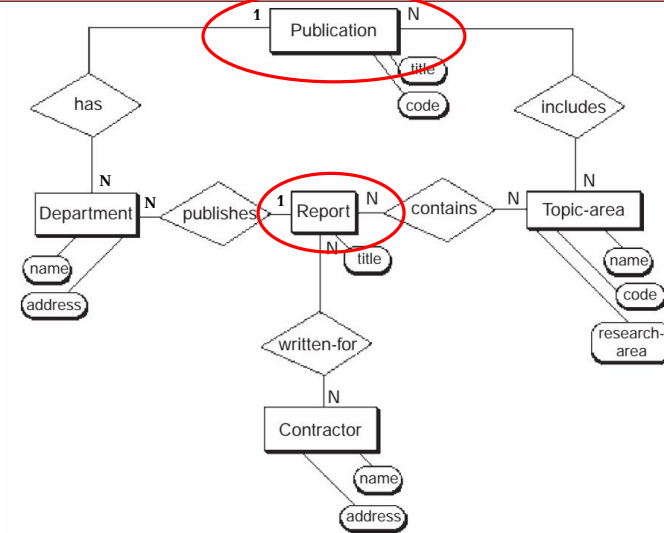
Пример спајања схема - потпуно, али не и минимално



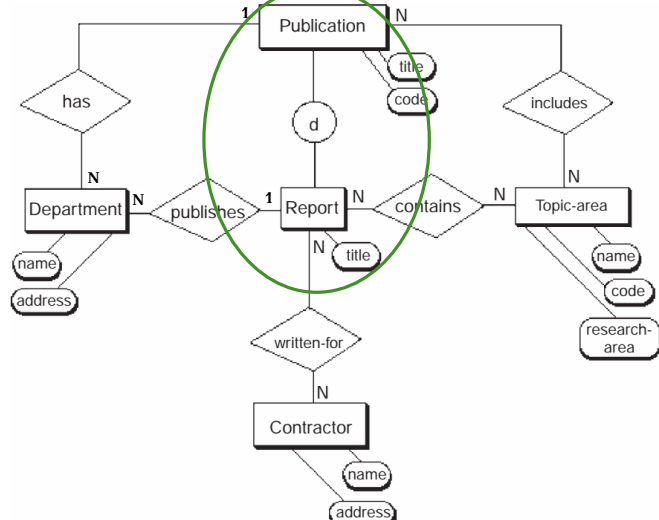
Минималност

- Препознајемо редундантне концепте
- Покушавамо да их разрешимо
 - увођењем генерализације
 - уклањањем редундантних односа

Пример спајања схема - редундантност публикације и извештаја

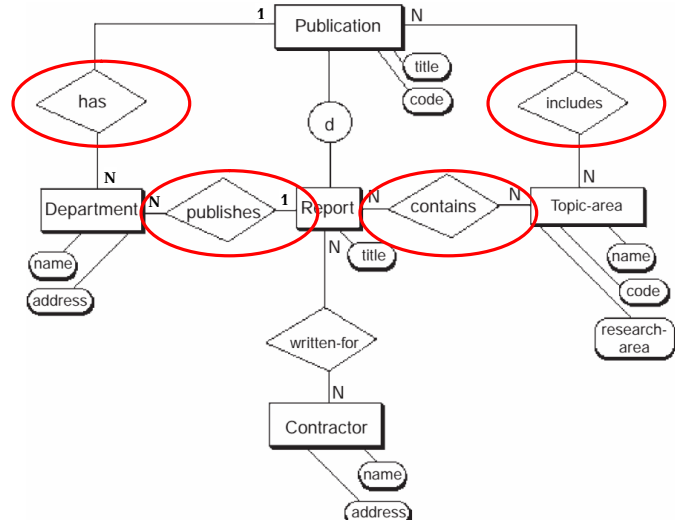


Пример спајања схема – уводимо генерализацију између публикације и извештаја



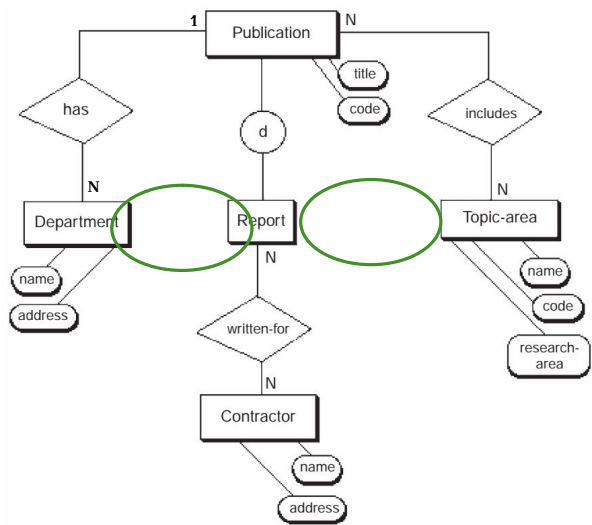
Универзитет у Београду - Математички факултет

Пример спајања схема – редундантност односа



Универзитет у Београду - Математички факултет

Пример спајања схема – уклањамо редундантне односе са извештајем



Универзитет у Београду - Математички факултет

Концептуално пројектовање / Обједињени модел / Интегрисање погледа



Разумљивост

- По потреби додатно уређујемо модел да би био разумљивији

Универзитет у Београду - Математички факултет



Груписање ентитета

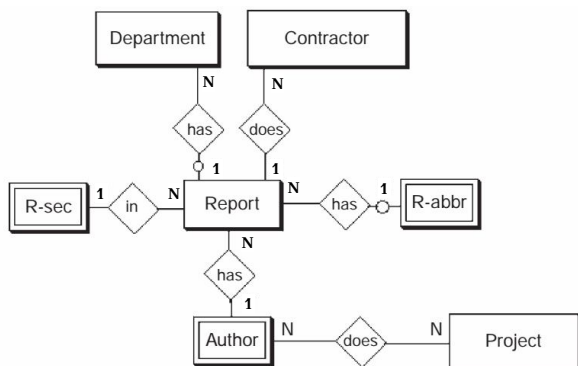
- Након интегрисања погледа може да се добије велика схема
 - слабо прегледна због величине
- Циљ груписања ентитета је препознавање снажније повезаних делова модела
 - ради представљања једног великог модела помоћу више мањих дијаграма



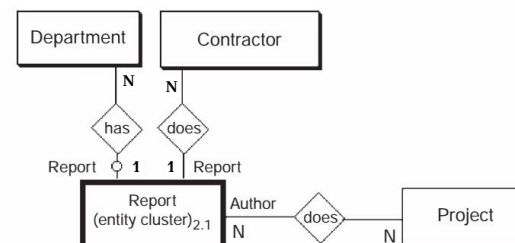
Техника груписања

- Детаљи представљања ентитета се обришу и ентитет се заокружи двоструком линијом
- Група ентитета се замењује симболом *трипле* са подебљаним оквиром

Пример груписања ентитета



Пример груписања ентитета





Принципи груписања

- Принципи:
 - Груписање према доминантности
 - Груписање према апстрактности
 - Груписање према условима
 - Груписање према односима
- Примењују се итеративно и/или рекурзивно



Груписање према доминантности

- Доминантни ентитети се уочавају на основу односа
 - учествују у већем броју односа и посредно повезују веће делове дијаграма
- Један доминантан ентитет се групише са свим припадајућим недоминантним ентитетима
 - слаби ентитети се придружују јаком
 - делови дијаграма који описују односе вишег реда се групишу
 - ако постоји агрегација, делови се придружују целини



Груписање према апстрактности

- Врши се ако постоје хијерархије
- Хијерархија се представља једним базним ентитетом
 - ако постоје битни специфични односи неких елемената хијерархије, онда се цела хијерархија дели на више група



Груписање према условима

- Ако постоје неки сложени услови који морају да важе у неким односима, одговарајући ентитети се групишу
 - циљ је склањање сложених услова из великог дијаграма, ради повећања читљивости
 - ако ти сложени услови имају висок значај за целину дијаграма, онда овакво груписање није пожељно



Груписање према односима

- Односи вишег реда и одговарајући ентитети могу да се групишу
 - оваква група представља однос као једну целину



Поступак груписања

- Препознати елементе који се групишу у оквиру функционалних области
 - функционалне области су пословне јединице или групе података које се најчешће користе заједно
- Груписати ентитете
 - свака група мора да у потпуности припада једној функционалној области
 - ако постоје конфликти између више алтернативних груписања, не правити такве групе
- Направити групе вишег реда
 - рекурзивном применом поступка
- Проверити исправност дијаграма
 - сви интерфејси између ентитета / објеката и других делова / нивоа дијаграма морају да буду конзистентни

Литература за тему

- Teorey, Lightstone, Nadeau, Jagadish, **Database Modeling and Design**, 5.ed, Elsevier, 2011.
- Watt, Eng, **Database Design**, 2.ed, Open Edition, 2014.
- 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.