Увод у релационе базе података

10

Саша Малков

Универзитет у Београду
Математички факултет
2023/2024

[PM13]

Увод у РБП

Саша Малков

Тема 5.2

Логичко пројектовање
- Пречишћавање схеме -

Пречишћавање схеме

- Подсетимо се, логички модел се прави итеративно:
  - 1. корак: превођење концептуалног модела у логички
    - Прва итерација се прави на основу концептуалног модела
  - 2. корак: пречишћавање схеме, итеративно
    - Свака наредна итерација се прави мењањем претходне (енгл. *flexing*)

Циљеви пречишћавања схеме

- Основни циљ је доследно и потпуно усклађивање модела са моделираним доменом, познатим захтевима и теоријским моделом базе података
- Критеријуми су бројни:
  - Да ли модел испуњава функционалне захтеве?
  - Да ли модел испуњава нефункционалне захтеве?
  - Да ли је модел комплетан?
  - Да ли је у складу са изабраним моделом података (релационим)?
  - Да ли модел гарантује интегритет података?
  - Да ли модел пружа потребну флексибилност?
  - Да ли модел омогућава ефикасан рад?
  - Да ли је модел довољно употребљив?

Пречишћавање схеме и релациони модел

- У случају релационог модела, специфичан део пречишћавања схеме је отклањање свих видова редундантности

Проблеми редундантности

- Редундантно чување података
  - неки подаци се поновљено чувају, па више копија истог податка непотребно оптерећује просторне капацитете
- Аномалије ажурирања
  - ако се једна копија поновљеног податка ажурира, база података ће бити неконзистентна ако се не ажурирају и остале копије
- Аномалије додавања
  - записивање неких података може да доводи до неконзистентности ако се не запишу још неки додатни подаци
- Аномалије брисања
  - брисање неких података може да доводи до неконзистентности ако се не обришу још неки подаци

Пример редундантности

- **УписанКурс(**
  број индекса,
  име и презиме студента,
  шифра предмета,
  назив предмета,
  име наставника)
- Постоји потенцијалан проблем са редундантним подацима о наставницима, студентима и предметима

…

211/2015	Горан Петровић	М101	Анализа 1	Милан Марић
211/2015	Горан Петровић	М102	Линеарна алгебра	Петар Симић
212/2015	Тања Митровић	М101	Анализа 1	Милан Марић
212/2015	Тања Митровић	М102	Линеарна алгебра	Петар Симић
211/2015	Тања Митровић	М101	Линеарна алгебра	Милена Савић

- Последњи ред је неисправан
  - наведен је ради илустрације
  - биће игнорисан у неким корацима

Недефинисане вредности и редундантност

- Недефинисане вредности могу да помогну у решавању неких аномалија додавања и брисања
  - Ако морамо да додамо податак а не знамо друге потребне податке, онда можемо да наведемо недефинисане вредности
    - нпр. ако студент уписује курс а не знамо наставника, можемо да наведемо да је наставник недефинисан
  - Ако хоћемо да обришемо неке редундантне податке, а да не бришемо целе редове, онда можемо да уместо брисања упишемо недефинисане вредности
    - нпр. ако бришемо податак о наставнику на курсу а не желимо да обришемо податке о студентима који су уписали курс, можемо да упишемо да је наставник недефинисан
- ...али праве нове проблеме...

Декомпозиција

- Редундантност се појављује на местима на којима постоје ***не сасвим природне*** везе између атрибута
- Уобичајено и ***исправно*** средство за решавање редундантности је декомпозиција
- Идеја је да се једна релација са много атрибута замени већим бројем релација са по мање атрибута

...

- На пример, можемо да поделимо релацију:
  - УписанКурс(
    број индекса,
    име и презиме студента,
    шифра предмета,
    назив предмета,
    име наставника)
- на релације:
  - УписанКурс(
    број индекса,
    име и презиме студента,
    шифра предмета,
    назив предмета)
  - НаставникКурса(
    шифра предмета,
    име наставника)
- (тако смо решили само део проблема)

...

211/2015	Горан Петровић	M101	Анализа 1
211/2015	Горан Петровић	M102	Линеарна алгебра
212/2015	Тања Митровић	M101	Анализа 1
212/2015	Тања Митровић	M102	Линеарна алгебра

M101	Милан Марић
M102	Петар Симић

Питања у вези декомпозиције

- Сваки пут када размишљамо о декомпозицији морамо да се запитамо:
  - Који проблем желимо да решимо предложеном декомпозицијом?
  - Да ли га заиста и решавамо?
  - Да ли производимо неке нове проблеме?
- Одговоре на ова питања дају
  - анализа ***функционалних зависности***
  - теоријске основе ***нормалних форми*** релација
  - ... зато ћемо прво обрадити ове теме

[PM13]

Увод у РБП

Саша Малков

Тема 5.3

Логичко пројектовање
- Функционалне зависности -

Функционалне зависности, дефиниција

- ***Функционална зависност*** (ФЗ) је однос између два скупа атрибута на скупу торки једне релације:
  - Нека је R релација и нека су X и Y непразни скупови атрибута релације R.
  - Кажемо да скуп торки r релације R задовољава функционалну зависност $X \to Y$ акко за сваки пар торки $t1$ и $t2$ из r важи:
    - $t1.X = t2.X \Rightarrow t1.Y = t2.Y$

Функционалне зависности, пример

- Из дефиниције видимо да се функционалне зависности дефинишу на скуповима торки
  - можемо да кажемо да представљају уопштење концепта кључа на скупу торки

Функционалне зависности, пример

Индекс	Име	Презиме	Шиф.п.	Назив предмета	Име наставника
211/2015	Горан	Петровић	М101	Анализа 1	Милан Марић
211/2015	Горан	Петровић	М102	Линеарна алгебра	Петар Симић
212/2015	Тања	Митровић	М101	Анализа 1	Милан Марић
212/2015	Тања	Митровић	М102	Линеарна алгебра	Јована Антић
212/2015	Тања	Митровић	П100	Програмирање 1	Драгана Перић

- Које функционалне зависности можемо да уочимо на наведеном скупу торки?
  - {Индекс} → {Име, Презиме}
  - {Шифра предмета} → {Назив предмета}
  - {Индекс, Шифра предмета} → {Име наставника}
  - {Име, Шифра предмета} → {Име наставника}
  - {Презиме, Назив предмета} → {Име наставника}
  - … и још транзитивних зависности

Функц. зависности на релацијама

- Ако говоримо о релацијама, онда имамо транзицију појма:
  - уместо да *уочавамо* зависности на конкретним подацима…
    - …т.ј. по дефиницији на скупу свих торки релације
  - …ми ћемо да *прописујемо* да је садржај релације *исправан* акко га чини скуп редова за који важе дате функционалне зависности
- Функцонална зависност на нивоу релације представља *правило интегритета*
  - установљена ФЗ *мора* увек да важи на скупу свих редова релације

Функционалне зависности, пример

Индекс	Име	Презиме	Шиф.п.	Назив предмета	Име наставника
211/2015	Горан	Петровић	М101	Анализа 1	Милан Марић
211/2015	Горан	Петровић	М102	Линеарна алгебра	Петар Симић
212/2015	Тања	Митровић	М101	Анализа 1	Милан Марић
212/2015	Тања	Митровић	М102	Линеарна алгебра	Јована Антић
212/2015	Тања	Митровић	П100	Програмирање 1	Драгана Перић

- Неке ФЗ важе на посматраном скупу редова, али не представљају опште правило
- Од претходно уочених ФЗ на посматраном скупу торки неке не могу да постану ФЗ на нивоу релације:
  - **{Име, Шифра предмета} → {Име наставника}**
    - два студента са истим именом могу да слушају предмет код различитих наставника
  - **{Презиме, Назив предмета} → {Име наставника}**
    - два студента са истим презименом могу да слушају предмет код различитих наставника
    - два предмета са истим називом могу имати различите наставнике на разчичитим смеровима
  - … и друге

Функц. зависности на релацијама (2)

- ФЗ на релацијама установљавамо
  - разматрањем ФЗ уочених на примерима скупова података
  - разматрањем општих пословних правила која важе у домену
  - свођењем на основни скуп ФЗ које се не могу извести из других ФЗ
- Зато нам је важно да познајемо особине ФЗ

“Аксиоме” извођења ФЗ

- Аксиоме извођења ФЗ (називају се и Армстронгове аксиоме) су комплетне и затворене на скупу свих ф.зависности
  - Називају се *аксиоме извођења* зато што се све сложеније ФЗ могу извести из основних применом ових аксиома
  - Заправо нису аксиоме, већ могу да се тривијално изведу из дефиниције ФЗ
  - Важе еквивалентне аксиоме на скупу торки

“Аксиоме” извођења ФЗ

- За све скупове атрибута X, Y и Z неке релације важе:
  - **Рефлексивност**

    $X \rightarrow X$
    - користи се и у облику

      $X \supseteq Y \;\Rightarrow\; X \rightarrow Y$

  - **Проширивост**

    $X \rightarrow Y \;\Rightarrow\; (\forall Z)\, XZ \rightarrow Y$
    - користи се и у облику

      $X \rightarrow Y \;\Rightarrow\; (\forall Z)\, XZ \rightarrow YZ$

  - **Транзитивност**

    $X \rightarrow Y \wedge Y \rightarrow Z \;\Rightarrow\; X \rightarrow Z$

Функционалне зависности, пример

Индекс	**Име**	**Презиме**	**Шиф.п.**	**Назив предмета**	**Име наставника**
211/2015	Горан	Петровић	М101	Анализа 1	Милан Марић
211/2015	Горан	Петровић	М102	Линеарна алгебра	Петар Симић
212/2015	Тања	Митровић	М101	Анализа 1	Милан Марић
212/2015	Тања	Митровић	М102	Линеарна алгебра	Јована Антић
212/2015	Тања	Митровић	П100	Програмирање 1	Драгана Перић

- Примери ***рефлексивности***:
  - {Индекс, Име, Презиме, Назив предмета} → {Презиме}
  - {Индекс, Име, Презиме, Назив предмета} → {Индекс, Име, Презиме}
  - {Индекс, Име, Презиме, Назив предмета} → {Индекс, Назив предмета}
  - … и друге

Функционалне зависности, пример

Индекс	**Име**	**Презиме**	**Шиф.п.**	**Назив предмета**	**Име наставника**
211/2015	Горан	Петровић	М101	Анализа 1	Милан Марић
211/2015	Горан	Петровић	М102	Линеарна алгебра	Петар Симић
212/2015	Тања	Митровић	М101	Анализа 1	Милан Марић
212/2015	Тања	Митровић	М102	Линеарна алгебра	Јована Антић
212/2015	Тања	Митровић	П100	Програмирање 1	Драгана Перић

- Примери ***проширивости*** (на скупу торки):
  - {Име} → {Презиме} ⇒ {Име, Индекс} → {Презиме, Индекс}
  - {Шиф.п.} → {Назив предмета} ⇒ {Шиф.п., Индекс} → {Назив предмета, Индекс}
  - … и друге

Функционалне зависности, пример

Индекс	Име	Презиме	Шиф.п.	Назив предмета	Име наставника
211/2015	Горан	Петровић	М101	Анализа 1	Милан Марић
211/2015	Горан	Петровић	М102	Линеарна алгебра	Петар Симић
212/2015	Тања	Митровић	М101	Анализа 1	Милан Марић
212/2015	Тања	Митровић	М102	Линеарна алгебра	Јована Антић
212/2015	Тања	Митровић	П100	Програмирање 1	Драгана Перић

- Примери ***транзитивности*** (на скупу торки):
  - {Име, Шиф.п.} → {Презиме, Шиф.п.},
    {Презиме, Шиф.п.} → {Име наставника}
    ⇒ {Име, Шиф.п.} → {Име наставника}
  - ... и друге

Најважније изведене особине ФЗ

- Изводе се из аксиома
  - Често се и оне називају аксиомама
- Најважније су:
  - ***Адитивност*** (или ***унија***)
  - ***Пројективност*** (или ***декомпозиција***)
  - ***Псеудотранзитивност***

Адитивност

- **Адитивност**

$$X \to Y \land X \to Z \Rightarrow X \to YZ$$

  - Ако не би важило, онда би за неко $X=x$ пројекција на YZ имала две торке, па би онда бар једна од пројекција на Y и Z такође имала две торке, а то је супротно претпоставци...

Пројективност

- **Пројективност**

$$X \to YZ \Rightarrow X \to Y \land X \to Z$$

  - За свако $X=x$ пројекција на YZ има тачно једну торку, па онда и пројекција на Y или Z такође има тачно једну торку...

Псеудотранзитивност

- **Псеудотранзитивност**

$$X\to Y \ \wedge\ YZ\to W \ \Rightarrow\ XZ \to W$$

  - $Z\to Z$, из рефлексивности
  - $XZ\to Z$, из рефлексивности
  - $XZ\to Y$, на основу претпоставке и проширивости
  - $XZ\to YZ$, на основу претходна два корака и адитивности
  - $XZ\to W$, на основу претходног, претпоставке и транзитивности

- Често се користи у нешто специфичнијем облику:

$$X\to Y \ \wedge\ XY\to Z \ \Rightarrow\ X \to Z$$

Значај функционалних зависности

- Значај ФЗ је вишеструк, али за нас је најважније да:
  - постојање сложених ФЗ може да укаже на редундантности
    - на пример, ако у релацији има транзитивних ФЗ, онда су испуњени предуслови за појаву редундантног садржаја релације
  - постојање ФЗ нам сугерише да можемо да декомпонујемо релацију без губитка информација
    - наравно, у складу са уоченим ФЗ
    - на пример, раздвајањем транзитивних ФЗ у различите релације
    - шта је гаранција очувања вих информација?

Функц. зависности и редундантност

- Нека су X, Y, Z подскупови атрибута релације R
- Нека важе ФЗ: $X\to Y$ и $Y\to Z$
  - (приметимо да онда важи и $X\to Z$)
- Ако бар једна од ових ФЗ није 1-1, онда у релацији R могу да се појаве редундантне вредности

- На пример, нека $X\to Y$ није 1-1, онда релација може да садржи торке (пројектоване на XYZ):
  - (x_1, y_1, z_1)
  - (x_2, y_1, z_1)
  - очигледно је да је Z редундандан скуп атрибута и да мора да се мења када год се мења Y

Функц. зависности и редундантност (2)

- У начелу, скоро свака релација у којој важи више функционалних зависности представља могућност за појављивање редундантних торки
  - осим ако су све те ФЗ 1-1

- Из угла редундантности, у идеалном случају би у свакој релацији требало да постоји тачно једна ФЗ
  - Улога релације би онда била да моделира ту једну ФЗ
  - У пракси то баш не може тако
    - Одлично је из угла трансакција
    - Оптерећујуће из угла упита
    - Није увек ни теоријски оствариво

- Да ли можемо релацију са више ФЗ да поделимо на више релација а да не изгубимо информације?
  - (одговор на ово питање ћемо потражити за неколико минута)

Затворење скупа функц. зависности

- У разматрању ФЗ често је потребно сагледати међузависности између скупова зависности или између скупова атрибута
- Нека је F неки скуп ФЗ релације R. ***Затворење од F***, у означи F^+, је најмањи скуп ФЗ који садржи F, такав да применом Армстронгових аксиома на F^+ не може да се добије ниједна ФЗ која није већ у F^+.
- Нека је F неки скуп ФЗ релације R и нека је X неки скуп атрибута те релације. Неки скуп атрибута је ***затворење од X над F***, у означи X^+, акко постоји ФЗ $X \to X^+$ у F^+ и ако за сваку другу ФЗ $X \to Z$ из F^+ важи $Z \subseteq X^+$.
  - т.ј. X^+ је максимални скупа атрибута зависних од X

Скуп зависних атрибута

- ***Скуп зависних атрибута*** X^+ неког скупа атрибута X је скуп свих атрибута који су (посредно или непосредно) функционално зависни од атрибута из X
  - т.ј. представља затворење у односу на све ФЗ у релацији
- (Алгоритме за конструкцију затворења ФЗ и атрибута научити из књиге ГПЛ)

Функционалне зависности и интегритет

- Концепт интегритета у релационом моделу је тесно повезан са функционалним зависностима
- Видећемо како се кључ-кандидат, примарни кључ и јединствени кључ повезују са функционалним зависностима

Наткључ и кључ-кандидат

- Скуп атрибута X релације R је наткључ акко је $X^+ = Attr(R)$
  - т.ј. неки скуп атрибута можемо да користимо као кључ за идентификовање торки релације акко су сви остали атрибути релације (непосредно или посредно) зависни од њега
- Наткључ X релације R је кључ-кандидат акко за сваки други наткључ Y важи: $Y \subseteq X \Rightarrow Y=X$.
  - т.ј. неки наткључ је кључ-кандидат ако не постоји наткључ који је његов прави подскуп
  - приметимо да може постојати и други наткључ и други кључ кандидат који није његов подскуп

Интегритет кључа

- Услову кључа одговара функционална зависност облика:
  - { **атрибути кључа** } → { **сви остали атрибути** }
- Приметимо да је ово проблематично ако кључ садржи атрибуте који могу да имају недефинисане вредности
  - ФЗ се дефинишу на поређењу, тј.
    - ако су једнаки кључеви, онда су једнаки и остали атрибути
  - Како се пореде недефинисане вредности?

Интегритет јединствености ентитета

- Због тога се интегритет јединствености ентитета дефинише да представља додатно проширење и формализацију интегритета кључа
- За примарни кључ се бира кључ кандидат који не сме да има недефинисане вредности:
  - 1. { **атрибути примарног кључа** } → { **сви остали атрибути** }
  - 2. атрибути примарног кључа не смеју да имају недефинисане вредности

Интегритет јединственог кључа

- Интегритет јединственог кључа је потенцијално слабији облик јединствености од интегритета примарног кључа
- Имамо три алтернативна облика:
  - 1. { **атрибути јединственог кључа** } → { **сви остали атрибути** }
  - 2а. атрибути јединственог кључа не смеју да имају недефинисане вредности
    - исто као и јединственост ентитета
  - 2б. ако атрибути јединственог кључа имају недефинисане вредности, сматра се да су оне међусобно идентичне
    - различити редови не могу да садржи исте вредности дефинисаних атрибута и исте комбинације недефинисаних атрибута
  - 2в. ако атрибути јединственог кључа имају недефинисане вредности, сматра се да су оне међусобно различите
    - различити редови могу да садрже исте вредности дефинисаних атрибута и исте комбинације недефинисаних атрибута

Декомпозиције релационе схеме

- Декомпозиција релационе схеме је замењивање релационе схеме двема (или више) релационим схемама које заједно садрже све атрибуте полазне схеме
  - Обично се говори о релацијама, а мисли се на релационе схеме

Потпуност декомпозиције

- Декомпозиција је ***потпуна*** (комплетна) ако гарантује да ће одговарајуће релације увек садржати све оне информације које би садржала полазна релација
- Потпуност се формализује преко ***реконструктибилности***
- Кажемо да је декомпозиција *потпуна* акко полазна релација може да се реконструише спајањем резултујућих релација

Очување зависности

- Потпуност декомпозиције може да се сведе на очување функционалних зависности
- Декомпозиција ***чува функционалне зависности*** ако се из функционалних зависности на декомпонованим релацијама могу реконструисати све функционалне зависности на полазној релацији
- Теорема: **Ако декомпозиција чува функционалне зависности, онда је она потпуна.**
  - Представља један од најважнијих теоријских елемената релационог модела и основ за пречишћавање схеме и елеминацију редундантности из модела

Функц. зависности на релацијама (2)

- **Теорема**: Нека је $Attr(R)$ скуп атрибута релације R и нека за непразне покскупове атрибута $X,Y \subseteq Attr(R)$ важи ФЗ $X \to Y$, тада важи једнакост $R = R[XY] * R[XZ]$, где је $Z=Attr(R) \setminus XY$.
  - т.ј. ако из релације издвојимо нову релацију тако да моделира једну ФЗ, а при томе у релацији задржимо атрибуте домена зависности, онда је декомпозиција потпуна
  - (формалан доказ је у књизи ГПЛ)
  - идеја је да на основу нове релације можемо да реконструишемо недостајуће атрибуте

Да ли се проблеми услед редундантности виде у концептуалном моделу?

- Неке зависности нису очигледне у концептуалном моделу
- Последица је да након превођења може да се добије логички модел у коме постоји редундантност
- На пример, можемо да добијемо релације:
  - *Студент(индекс, име и презиме, ниво студија)*
  - *Смер(ознака, назив, трајање)*
- које су у односу:
  - *Студент—(*)---студира---(*)--Смер*
- Све зависности у релацијама *Студент* и *Смер* су исправне...

…

- Редундантност се јасније уочава тек када се у модел уведу и кључеви везаних релација (или се посматра међусобна зависност односа и атрибута):
  - *Студент(индекс, име и презиме, ниво студија,* ***ознака смера****)*
  - *Смер(* ***ознака****, назив, трајање)*
- Сада је очигледно да постоји редундантност атрибута *ниво студија* у релацији *Студент*
- Исправно решење би било:
  - *Студент(индекс, име и презиме, ознака смера)*
  - *Смер(ознака, назив,* ***ниво студија****, трајање)*

…

- Да би се редундантности виделе у концептуалном моделу, модел мора да се критички посматра и стално део по део „преводи“ у релациони (тј. логички) модел
  - то није оствариво пре интегрисања погледа
    - превођење једног погледа може да да непотпун “логички” модел без довољно информација да би се уочиле редундантности
  - ретко се ради и након интегрисања погледа
    - нема много смисла да се више пута прави логички модел и поправља концептуални чак и када редундантности не представљају проблем у концептуалном моделу

[PM13]

Увод у РБП

Саша Малков

Тема 5.4

Логичко пројектовање - Основне нормалне форме -

Нормалне форме

- *Нормалне форме* су специфични облици релација који задовољавају одређена правила у вези функционалних зависности
  - који зато *гарантују* да у релацији неће бити редундантности одређеног типа

Нормалне форме

- У свом првом раду о релационом моделу (1970), Код је дефинисао само
  - 1. нормалну форму (*1NF*)
- Годину дана касније (1971) је дефинисао
  - 2. нормалну форму (*2NF*)
  - 3. нормалну форму (*3NF*)
- Заједно са Рејмондом Бојсом је дефинисао (1974)
  - Бојс-Кодову нормалну форму (*BCNF*)

- Касније су формулисане још и
  - 4. нормална форма (*4NF*) (1977)
  - 5. нормална форма (*5NF*) (1979)
  - Нормална форма домена и кључа (*DKNF*) (1981)
  - Нормална форма елементарног кључа (*EKNF*) (1982)
  - 6. нормална форма (*6NF*) (2003)
  - Нормална форма есенцијалних торки (*ETNF*) (2012)
  - Нормална форма суперкључева (*SKNF*)
  - Нормална форма без редунданси (*RFNF*)

Нормалне форме (2)

- Већина нормалних форми (све осим 6*NF*) могу да се поређају тако да наредна подразумева све претходне:
  - 1. нормална форма (*1NF*)
  - 2. нормална форма (2*NF*)
  - 3. нормална форма (3*NF*)
  - Нормална форма елементарног кључа (*EKNF*)
  - Бојс-Кодова нормална форма (*BCNF*)
  - 4. нормална форма (4*NF*)
  - Нормална форма есенцијалних торки (*ETNF*)
  - Нормална форма без редунданси (*RFNF*)
  - Нормална форма суперкључева (*SKNF*)
  - 5. нормална форма (5*NF*)
  - Нормална форма домена и кључа (*DKNF*)

- 6. нормална форма (6*NF*) представља алтернативу за *EKNF* и не може да се непосредно упореди са осталим нормалним формама

---

1. нормална форма

- Релација је у првој нормалној форми ако сваки атрибут може да има само атомичне вредности.

  - Релациони модел претпоставља атомичне вредности
  - ЕР омогућава и сложене вредности атрибута

---

1. нормална форма (пример)

- Нека један предмет може да држи више наставника
- *Неатомични* модел би омогућавао да у релацији ***предмет*** постоји атрибут ***наставници*** са именима више наставника
- *1.НФ* то забрањује
- Једно могуће решење је да се атрибут ***наставници*** замени већим бројем атрибута ***наставник1***, ***наставник2***, ***наставник3***
  - такво решење није добро ни ако има мање ни ако има више од 3 наставника за неки предмет
- Боље решење је да се уведе нова релација која моделира однос предмета и наставника

2. нормална форма

- Релација је у другој нормалној форми
  - ако је у првој нормалној форми и
  - ако ниједан некључан атрибут (тј. који не припада ниједном кандитату кључу) није функционално зависан од неког правог подскупа атрибута неког кандидат кључа.
- Суштина је да у случају сложеног кључа сви остали атрибути зависе од *целог* кључа а не од неког његовог дела.
  - т.ј. кандидати кључеви су минимални не само у односу на скуп свих атрибута него и у односу на појединачне атрибуте

---

2. нормална форма (пример)

- Нека следећа релација описује предмете које су уписали студенти:
  - *УписанПредмет{ **индекс**, **шифпред**, име, презиме, називпред, којипут }*
  - Релација је у 1.НФ
- Релација није у 2.НФ
  - *име* и *презиме* зависе од атрибута *индекс*
  - *називпред* зависи од атрибута *шифпред*
- Да би се задовољила 2.НФ потребно је да се релација подели на више релација
- 3 релације:
  - *УписанПредмет{ **индекс**, **шифпред**, којипут }*
  - *Студент{ **индекс**, име, презиме }*
  - *Предмет{**шифпред**, називпред }*

---

2. нормална форма (пример)

- Релацију
  - *УписанПредмет{ **индекс**, **шифпред**, име, презиме, називпред, којипут }*
- Можемо да поделимо на 3 релације:
  - *УписанПредмет{ **индекс**, **шифпред**, којипут }*
  - *Студент{ **индекс**, име, презиме }*
  - *Предмет{**шифпред**, називпред }*
- Према теореми о декомпозицији, декомпозиција је потпуна
  - Показује се за једну по једну декомпозицију, најпре:
    - *УписанПредмет{ **индекс**, **шифпред**, називпред, којипут }*
    - *Студент{ **индекс**, име, презиме }*
  - а затим и
    - *УписанПредмет{ **индекс**, **шифпред**, којипут }*
    - *Предмет{**шифпред**, називпред }*

---

2. нормална форма (дискусија)

- Нарушавање 2.НФ указује на неисправно груписање атрибута у релацији
  - релација моделира више различитих ФЗ
  - то води редундантности
- Постизање 2.НФ је кључно за позиционирање атрибута у *одговарајуће* релације
- Ипак, 2.НФ допушта да у релацији постоје два кључа кандидата, као и транзитивне зависности
- Увек је могућа потпуна декомпозиција

3. нормална форма

- Ако је R релација и X неки подскуп њених атрибута и A неки атрибут релације R, онда је R у 3.НФ акко за сваку ФЗ $X \rightarrow A$ на релацији R важи једно од:
  - $A \in X$, тј. A припада скупу X (тзв. тривијална зависност)
  - X је наткључ (тј. кључ или садржи кључ)
  - A је део неког кључа релације
- Оригинална дефиниција (Код, 1971)
  - Релација је у 3.НФ акко је
    - у 2.НФ и
    - сваки некључан атрибут је нетранзитивно зависан од сваког кључа.

---

3. нормална форма (2)

- У основи налаже да сваки некључни атрибут може да представља само додатни опис неког кључа
- Тј. нема транзитивних зависности међу некључним атрибутима (некључни атрибути не представљаују описе некључних атрибута)

- Могућа нарушавања 3.НФ:
  - X је подскуп неког кључа K
    - тј. имамо зависност од дела кључа
      - нарушена је и 2.НФ
  - X није ни подскуп ни надскуп ниједног кључа
    - тј. имамо транзитивну зависност

---

3. нормална форма (пример)

- Нека следећа релација описује уписане студенте:
  - *Студент*{ ***индекс***, *име*, *презиме*, *студпрограм*, *ниво*, *трајање* }
- Релација јесте у 2.НФ али није у 3.НФ
  - *ниво* и *трајање* зависе од некључног атрибута *студпрограм*
- Да би се задовољила 3.НФ потребно је да се релација подели на 2 релације:
  - *Студент*{ ***индекс***, *име*, *презиме*, *студпрограм* }
  - *Програм*{ ***студпрограм***, *ниво*, *трајање* }

---

3. нормална форма (дискусија)

- Нарушавање 3.НФ указује да је у релацији моделирано више релативно независних ФЗ
- Увек је могућа потпуна декомпозиција

Нормална форма елементарног кључа

- Релација је у *НФЕК* ако је у 3.НФ акко за сваку елементарну ФЗ $X \rightarrow Y$ на релацији R важи једно од:
  - X је кључ или
  - Y је елементарни кључ или део елементарног кључа
- При томе:
  - ФЗ $X \rightarrow Y$ је *елементарна* акко не постоји ФЗ $Z \rightarrow Y$, ни за који прави подскуп атрибута $Z \subset X$
  - Кључ је *елементаран* ако је полазиште бар једне елементарне функционалне зависности
- Ова форма стоји између *3.НФ* на *БКНФ*
- Пример и мотивација ће бити објашњени након *БКНФ*

Бојс-Кодова нормална форма

- Ако је R релација и X неки подскуп њених атрибута и A неки атрибут релације R, онда је R у Бојс-Кодовој нормалној форми акко за сваку ФЗ $X \rightarrow A$ на релацији R важи једно од:
  - $A \in X$, тј. A припада скупу X, тзв. тривијална зависност
  - X је наткључ
- Тј.
  - сваки атрибут релације је или део кључа или зависи од *целог* кључа
  - не постоје зависности међу некључним атрибутима ни транзитивне зависности

Бојс-Кодова нормална форма, дискусија

- Релације које су у 3.НФ најчешће задовољавају и БКНФ
  - Практично увек када постоји само један кључ или нема преклапања међу кључевима
  - Већина примера који се могу пронаћи у литератури заправо нису исправни, тј. јесу у 3.НФ иако аутори тврде супротно
- Штавише, тешко је пронаћи добар пример који
  - јесте у 3.НФ
  - није у БКНФ
  - може да се преведе у БКНФ

БКНФ, пример 1.1

- (https://en.wikipedia.org/wiki/Boyce-Codd_normal_form)
- Посматрамо релацију
  - *Резервација{ **терен**, **од**, до, тарифа }*
    - постоје два терена, један је травнати и један бетонски
    - резервације су дефинисане тереном и периодом
    - постоје 4 тарифе – по две за сваки од терена зависно да ли је термин предвиђен за чланове или не
- Релација јесте у 2.НФ и 3.НФ (нема атрибута који су ван кључева)
  - постоји већи број кључева
    - *{ **терен**, **од**, до, тарифа }*
    - *{ **терен**, од, **до**, тарифа }*
    - *{ терен, **од**, до, **тарифа** }*
    - *{ терен, од, **до**, **тарифа** }*
- Релација ***није*** у БКНФ
  - проблем је у зависности *тарифа* -> *терен*
    - *терен* не припада скупу *{ тарифа }*
    - *тарифа* није наткључ

БКНФ, пример 1.2

- Један начин да се релација преведе у БКНФ је додавање новог атрибута *за_чланове* и подела на две релације:
    - *Резервација{ **терен**, **за_чланове**, **од**, до }*
    - *Тарифа{ **тарифа**, терен, за_чланове }*
- Испоставља се да је проблем у *скривеним* ФЗ које постају очигледне и решиве када се додају нови атрибути

БКНФ, пример 2.1

- Посматрамо релацију
    - *Распоред{ **радник**, **тим**, руководилац }*
- и зависности
    - *радник, тим -> руководилац*
        - сваки радник може да буде у више тимова
        - у сваком тиму радник има руководиоца
        - у једном тиму може да буде више руководилаца
            - али су одговарајући скупови подређених радника дисјунктни
    - *руководилац -> тим*
        - руководилац може да руководи само у једном тиму

БКНФ, пример 2.2

- Релација *Распоред* задовољава 3НФ али не и БКНФ
- Практичан проблем је да може доћи до *аномалије мењања:*
    - ако неко промени податак о *тиму* у коме је неки радник, а при томе не промени *руководиоца*...
    - ...онда се добија да руководилац потенцијално руководи у различитим тимовима...
    - ...што није у складу са функционалним зависностима

БКНФ, пример 2.3

- (*Teorey, 2011*) наводи овај пример:
    - “релација јесте у 3.НФ а није у БКНФ”
    - наводи и аномалију брисања:
        - ако сви радници код неког руководиоца напусте тим, онда ће се изгубити информација о томе да тај руководилац има статус руководиоца у тиму
    - предлаже да се проблем реши тако да се додатно уведе релација *Руководилац{ **руководилац**, тим }* која би описивала другу зависност
        - као и да се у првој направи страни кључ у односу на другу
        - чиме би се гарантовао интегритет
    - приметимо да полазна релација остаје у 3НФ а не и БКНФ

БКНФ, пример 2.4

- Приметимо да не бисмо поправили ствар ако бисмо направили декомпозицију:
  - { **радник**, **руководилац** }, { **руководилац**, тим }
- Тако бисмо увели аномалију мењања у односу на промену тима:
  - ако руководилац пређе у други тим, имплицитно ће у други тим прећи и сви радници којима је руководио

БКНФ, пример 2.5

- Заправо, претходни проблем би могао да се реши тако што уочимо да имамо још једну зависност која одређује кључ-кандидат:
  - *радник, руководилац -> тим*
  - Она можда не изгледа *природно*, али је последица ограничења да (1) сваки руководилац руководи највише једним тимом и (2) да сваки радник има руководиоца у сваком тиму у који је распоређен
    - уосталом, и последица је постојеће зависности и аксиоме прошириvости
- Ако уочимо да та ФЗ одређује кључ-кандидат, онда видимо и да није задовољена чак ни 2.НФ:
  - *радник, руководилац -> руководилац -> тим*
- А раније уочена зависност *радник, тим -> руководилац* постаје последица

БКНФ, пример 2.6

- Да бисмо задовољили 2.НФ морамо да направимо две релације:
  - *Распоред{ **радник**, **руководилац** }*
  - *Руководилац{ **руководилац**, тим }*
- Сада су задовољене и 2.НФ и 3.НФ и БКНФ
- ...а видимо да претходни пример заправо није задовољавао 2.НФ (а тиме ни 3.НФ) зато што нису довољно пажљиво разматране функционалне зависности
- Међутим, тако правимо нови проблем у виду аномалије мењања
  - ако руководилац пређе у други тим, имплицитно ће у други тим прећи и сви радници којима је руководио
  - тј. изгубили смо зависност { радник, тим } -> { руководилац }

БКНФ и НФЕК

- Нека имамо релацију:
  - *ПријаваИспита(**студент**, **испитни рок**, предмет)*
  - кључ је *{ студент, испитни рок }*
- и зависности:
  - *{ студент, испитни рок }* → *{ предмет }*
    - тј. студент у једном исп.року може да пријави испит из највише једног предмета
  - *{ предмет }* → *{ испитни рок }*
    - тј. сваки предмет се полаже само у по једном испитном року
- Тада:
  - релација *ПријаваИспита* није у БКНФ:
    - испитни рок зависи од предмета, а предмет није део кључа
  - ако направимо декомпозицију, нарушићемо прву зависност...
  - ако променимо кључ (*студент*, *предмет*), проблем опстаје

БКНФ и НФЕК (2)

- Али:
  - релација *ПријаваИспита* ***јесте*** у НФЕК
  - прва зависност је од целог кључа
  - друга је према делу кључа
  - обе су елементарне
- Значај НФЕК је у томе што неке релације не могу да пређу пут од 3НФ до БКНФ а да се очувају све зависности, али могу да дођу бар до НФЕК
- Да би се у моделу сачувале све зависности, потребно је да се постојећој релацији дода још једна која описује другу зависност:
  - *ПредметУРоку (* ***предмет****, испитни рок)*
  - Уводи се редундантност, али ФЗ су такве...

БКНФ и НФЕК (3)

- Приметимо да овде проблем не може да се реши као у претходном примеру, увођењем алтернативних зависности, зато што постоји ограничење:
  - у једном испитном року студент може да пријави испит из највише једног предмета
- Због тога овај пример испуњава и 2.НФ и 3.НФ и НФЕК, али не и БКНФ

БКНФ

- Приметимо да претходно представљени примери имају сличну форму:
  - Релација { A, B, C }
  - Зависности:
    - { A, B } -> { C }
    - { C } -> { A }
- Као што смо видели у примерима, у зависности од контекста, тај проблем некада може делимично да се реши, али никада потпуно

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.*
- Watt, Eng, **Database Design**, *2.ed, Open Edition, 2014.*