
5. Formalizace návrhu databáze

5.1. Úvod do teorie závislostí	2
5.1.1. Funkční závislost	2
5.1.2. Vícehodnotová závislost (multizávislost)	7
5.1.3. Závislosti na spojení	9
5.2. Využití teorie závislostí při návrhu databáze	12
5.2.1. Normalizace	13
5.2.2. Normální formy	18
5.2.3. Obecný postup odstranění částečných a tranzitivních závislostí	27
Literatura	28

- druhá významná teoretická podpora relačních databází

5.1. Úvod do teorie závislostí

5.1.1. Funkční závislost

- hodnota atributu relace určuje jednoznačně hodnotu jiného atributu téže relace
- zapisujeme $X \rightarrow Y$
- vyplývá z významu atributů, představuje integritní omezení

Př) Klient(r_číslo, jméno, ulice, město)

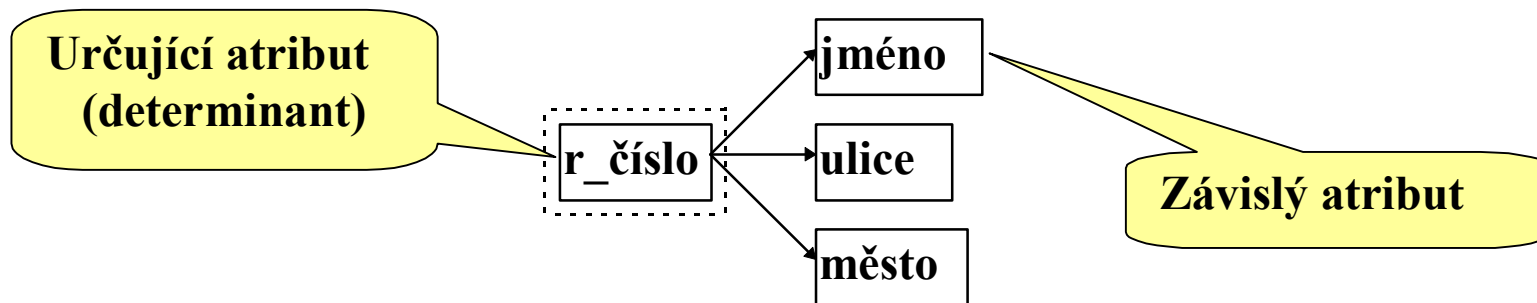
Hodnota rodného čísla jednoznačně určuje hodnoty ostatních atributů

r_číslo \rightarrow jméno
 \rightarrow ulice
 \rightarrow město

r_číslo \rightarrow (jméno, ulice, město)
jméno $\not\rightarrow$ (ulice, město)

Nechť X a Y jsou atributy relace R . Řekneme, že Y **funkčně závisí na X** , zapisujeme $X \rightarrow Y$, právě když pro libovolné dvě n -tice t_1 a t_2 každého přípustného stavu relace R platí, že je-li x_1 , resp. y_1 hodnota atributu X , resp. Y v n -tici t_1 a x_2 , resp. y_2 hodnota atributu X , resp. Y v n -tici t_2 a $x_1 = x_2$, potom i $y_1 = y_2$.

- Diagram funkční závislosti



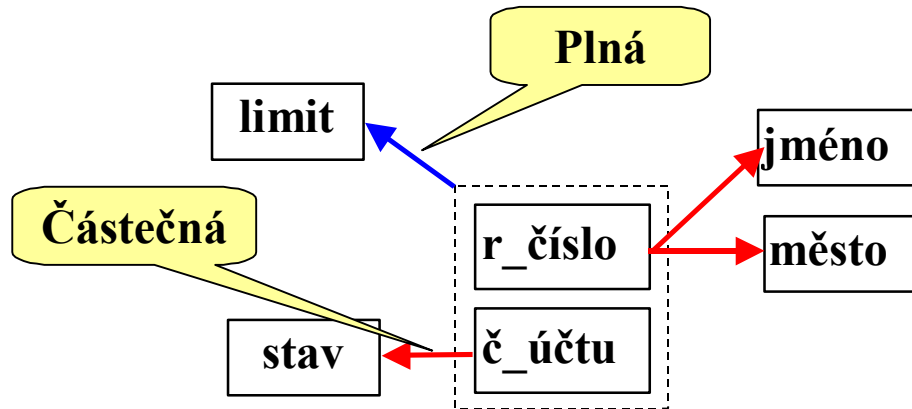
Praktický důsledek (integritní omezení): Opakuje-li se v relaci stejná hodnota determinantu, musí se opakovat i odpovídající stejné hodnoty závislého atributu.

- Triviální funkční závislost

$X \rightarrow Y$ platí pro každý atribut $Y \subseteq X$.

- Plná funkční závislost
 - atribut je funkčně závislý na celém složeném atributu a není funkčně závislý jen na některé jeho části

Př) Disponuje(r_číslo, jméno, město, č_úctu, stav, limit)



Nechť X a Y jsou atributy relace R . Řekneme, že atribut Y je **plně funkčně závislý** na atributu X , právě když je funkčně závislý na X a není funkčně závislý na žádném atributu $Z \subset X$.

Praktický důsledek: je-li kandidátní klíč relace složený, stejné hodnoty složek se mohou opakovat \Rightarrow musí se opakovat i hodnoty atributů, které jsou částečně (ne plně) závislé.

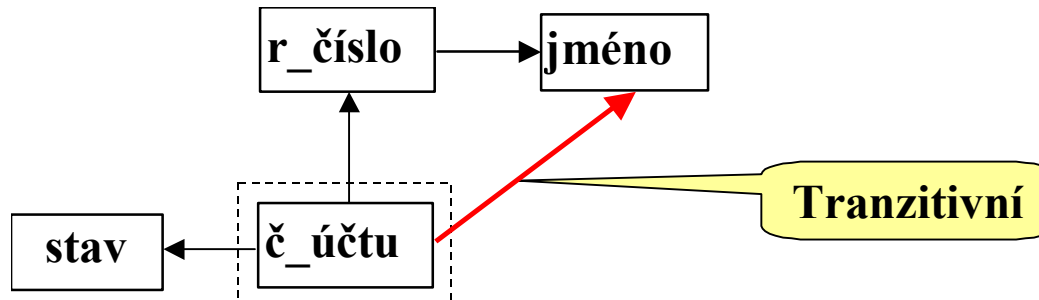
Př)

č_úctu	r_číslo	stav	jméno	limit	město
100	600528/0275	100000	Novák	10000	Praha
100	581015/9327	100000	Malá	3000	Brno
130	600528/0275	50000	Novák	5000	Praha
150	450205/3419	150000	Veselý	5000	Ostrava

- Tranzitivní závislost

- atribut je funkčně závislý na jiném funkčně závislém atributu

Př) Účet(č_úctu, stav, r_číslo, jméno)



Nechť X a Y jsou atributy relace R a necht' platí $X \rightarrow Y$, avšak neplatí $Y \rightarrow X$, a necht' existuje atribut Z relace R , který není v X , ani Y , a platí $Y \rightarrow Z$. Potom říkáme, že Z je **tranzitivně závislý** na X .

Praktický důsledek: existuje-li funkční závislost na atributu, který není kandidátním klíčem, hodnota se může opakovat ⇒ musí se opakovat i hodnoty závislého atributu.

Př)

č_úctu	stav	r_číslo	jméno
100	100000	600528/0275	Novák
120	135000	581015/9327	Malá
130	50000	600528/0275	Novák
150	150000	450205/3419	Veselý

5.1.2. Vícehodnotová závislost (multizávislost)

- hodnota atributu relace určuje jednoznačně množinu hodnot jiného atributu téže relace nezávisle na hodnotách ostatních atributů
- zapisujeme $X \twoheadrightarrow Y$

Př) Účet – r_číslo značí vlastníka, klient může mít více adres

- nejsou funkční závislosti, ale informace se opakuje

č_úctu	r_číslo	město
100	600528/0275	Praha
100	600528/0275	Brno
130	620726/1234	Praha
150	450205/3419	Ostrava

Nechť A je množina atributů relace R a X, Y, Z jsou atributy R takové, že $Z = A - (X \cup Y)$. Řekneme, že relace R *splňuje vícehodnotovou závislost* Y na X (zapisujeme $X \twoheadrightarrow Y$), právě když množina hodnot atributu Y odpovídající libovolné dvojici (x, z) , kde x je hodnota atributu X a z je hodnota atributu Z , závisí pouze na hodnotě x a nezávisí na z .

- **Věta (integritní omezení plynoucí z vícehodnotové závislosti):**

Relace R splňuje vícehodnotovou závislost $X \twoheadrightarrow Y$, právě když jsou-li $\{x, y, z\}$ a $\{x, y', z\}$ n-tice relace R, potom i $\{x, y', z\}$ a $\{x, y, z\}$ jsou n-tice R.

Př) Vložení informace o novém účtu klienta s RČ 600528/0275

- **Triviální vícehodnotová závislost**

$X \twoheadrightarrow Y$ platí pro každý atribut Y takový, že $Y \subseteq X$ nebo $X \cup Y = A$

- **Pravidlo doplňku**

V relaci R s množinou atributů A platí vícehodnotová závislost $X \twoheadrightarrow Y$, právě když platí i závislost $X \twoheadrightarrow A - Y$.

5.1.3. Závislosti na spojení

- relaci lze získat spojením relací, jejichž schéma je dekompozicí schématu původní relace
- zapisujeme $*(R_1, R_2, \dots, R_n)$, kde R_1, R_2, \dots, R_n tvoří dekompozici R

Př) Dodavatel – Produkt – Projekt

Sémantika: Dodává-li dodavatel S_x produkt P_y a P_y je používán v projektu G_z a dodavatel P_x dodává pro projekt G_z , pak S_x dodává P_y pro G_z (tzv. 3D omezení).

dodavatel	produkt	projekt
S1	P1	G2
S1	P2	G1
S2	P1	G1
S1	P1	G1

Relace R *splňuje závislost na spojení* $*(X, Y, \dots, Z)$, právě když je relace R rovna spojení projekcí na X, Y, \dots, Z , kde X, Y, \dots, Z jsou podmnožiny atributů relace R .

- Triviální závislost na spojení

Závislost na spojení (X, Y, \dots, Z) platí vždy, je-li některá z podmnožin atributů X, Y, \dots, Z rovna množině atributů A relace R .

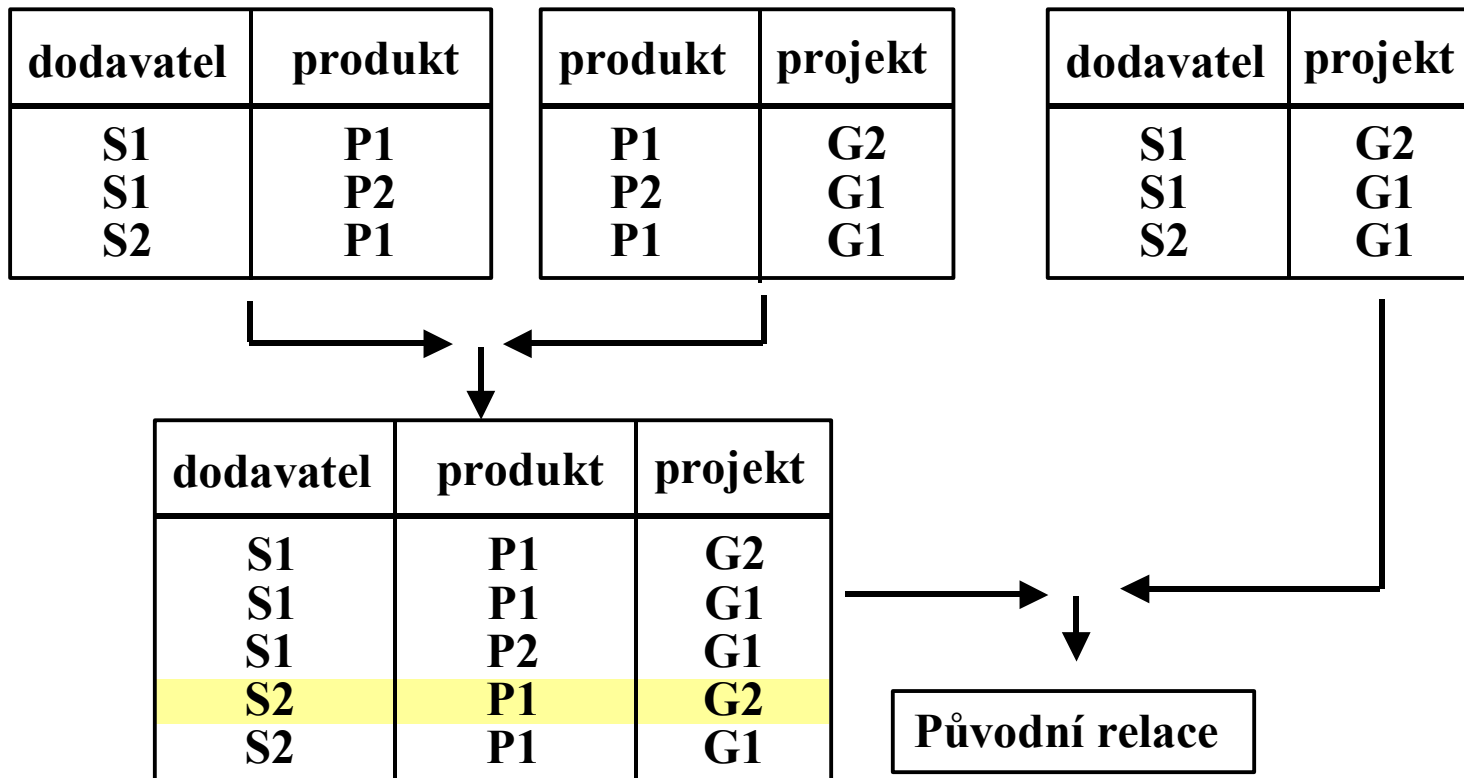
- závislost na spojení opět vede na integritní omezení

Př) Problémy aktualizace:

vložení (S2, P3, G2) \Rightarrow nutnost vložení i (S2, P1, G2)

zrušení (S1, P1, G1) \Rightarrow nutnost zrušit i některou ze zbývajících tří n-tic

dodavatel	produkt	projekt
S1	P1	G2
S1	P2	G1
S2	P1	G1
S1	P1	G1



5.2. Využití teorie závislostí při návrhu databáze

- proces návrhu založený na teorii závislostí se nazývá **normalizace**.

Postup návrhu: seznam atributů (univerzální relace) → postupná dekompozice na schéma v dostatečně vysoké **normální formě**

Praktický postup: datový model (ER diagram) → transformace na schéma relační databáze → zjemnění využitím normalizace resp. normalizace ER modelu před transformací.

- Hlavní problémy špatného návrhu:
 - opakující se informace (redundance),
 - nemožnost reprezentovat určitou informaci,
 - ztráta informace,
 - složitá kontrola integritních omezení.

Př) Účet1(č_úctu, r_číslo, stav, pobočka, jmění)

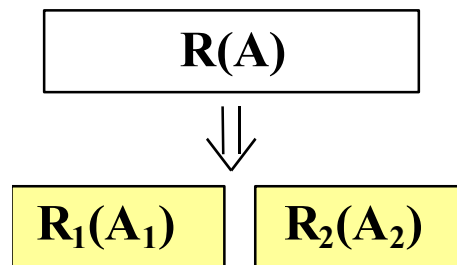
Předpokládejme, že s účtem může disponovat více osob.

č_úctu	r_číslo	stav	pobočka	jmění
100	600528/0275	100000	Jánská	10000000
100	581015/9327	100000	Jánská	10000000
130	600528/0275	50000	Palackého	5000000
150	450205/3419	150000	Palackého	5000000

5.2.1. Normalizace

- postupná transformace tabulky do vhodnějšího tvaru (postupná dekompozice)

Nechť $R(A)$ je schéma relace R . Množina schémat $\{R_1(A_1), R_2(A_2), \dots, R_n(A_n)\}$ je **dekompozicí schématu** $R(A)$, jestliže
$$A = A_1 \cup A_2 \cup \dots \cup A_n$$



- Žádoucí vlastnosti dekompozice:
 - bezeztrátovost při zpětném spojení
 - zachování závislostí
 - odstranění opakování informace (redundance)
- **Bezeztrátová dekompozice (Lossless-Join/Nonloss decomp.)**
 - spojení tabulek, které vzniknou dekompozicí musí dát přesně původní tabulku

Př) Účet1(č_úctu, r_číslo, stav, pobočka, jmění) –viz předchozí

č_úctu	r_číslo	Stav	pobočka	jmění
100	600528/0275	100000	Jánská	10000000
100	581015/9327	100000	Jánská	10000000
130	600528/0275	50000	Palackého	5000000
150	450205/3419	150000	Palackého	5000000

Účet_v_pob(č_úctu, r_číslo, pobočka, jmění),

Stav(r_číslo, stav)

č_úctu	r_číslo	pobočka	jmění
100	600528/0275	Jánská	10000000
100	581015/9327	Jánská	10000000
130	600528/0275	Palackého	5000000
150	450205/3419	Palackého	5000000

r_číslo	stav
600528/0275	100000
581015/9327	100000
600528/0275	50000
450205/3419	150000

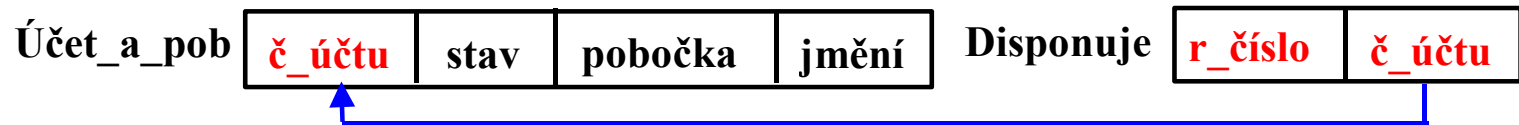
č_úctu	r_číslo	stav	pobočka	jmění
100	600528/0275	100000	Jánská	10000000
100	600528/0275	50000	Jánská	10000000
100	581015/9327	100000	Jánská	10000000
130	600528/0275	50000	Palackého	5000000
130	600528/0275	100000	Palackého	5000000
150	450205/3419	50000	Palackého	5000000

Podmínka bezeztrátové dekompozice:

Pro relacemi se schématy $R1(A1)$ a $R2(A2)$:

$A1 \cap A2 \rightarrow A1$ nebo $A1 \cap A2 \rightarrow A2$, tj. společný atribut musí být kandidátním klíčem alespoň jedné z tabulek.

Př)



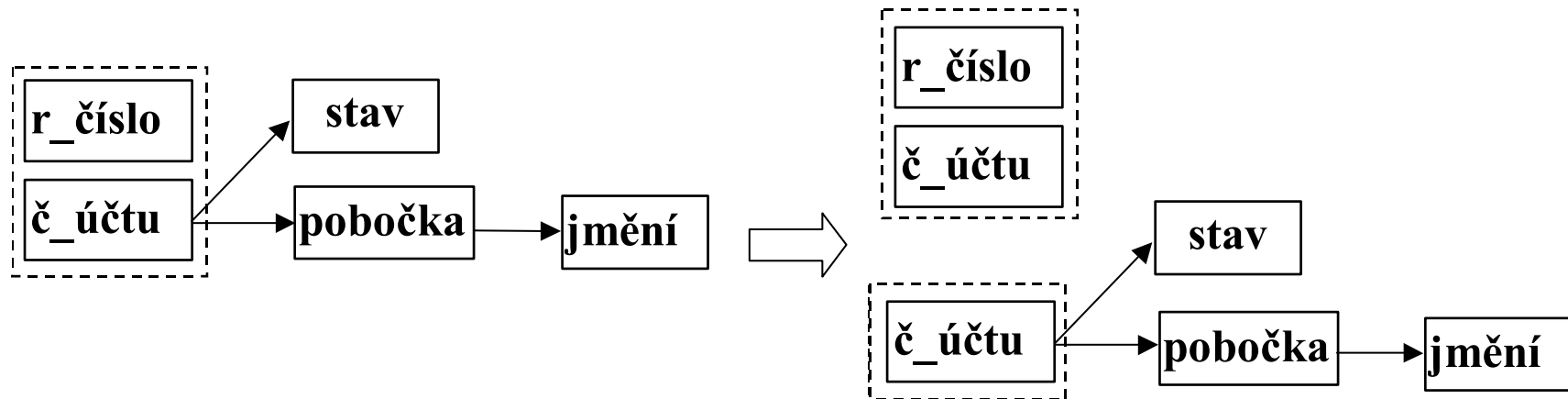
č_úctu	Stav	pobočka	jmění
100	100000	Jánská	10000000
130	50000	Palackého	5000000
150	150000	Palackého	5000000

č_úctu	r_číslo
100	600528/0275
100	581015/9327
130	600528/0275
150	450205/3419

- **Zachování závislostí**

- všechny původní závislosti musí být zachovány a snadno kontrolovatelné (v rámci jedné tabulky)

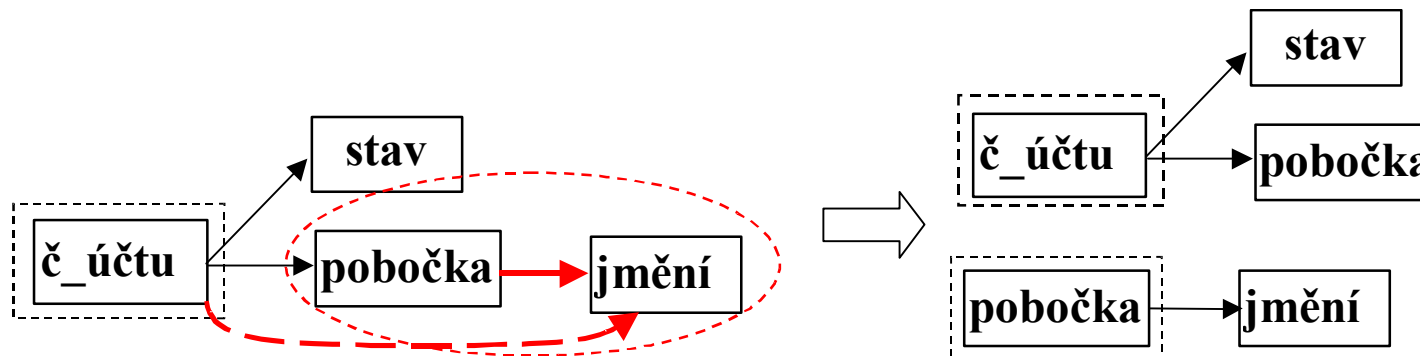
Př)



- **Odstranění opakování informace**

Př)

č_úctu	stav	pobočka	jmění
100	100000	Jánská	10000000
130	50000	Palackého	5000000
150	150000	Palackého	5000000



Účet

č_úctu	stav	pobočka
100	100000	Jánská
130	50000	Palackého
150	150000	Palackého

Pobočka

pobočka	jmění
Jánská	1000000
Palackého	500000

- **Boyce-Coddova normální forma (BCNF)**
 - odstraňuje opakování informace
 - všechny netriviální funkční závislosti jsou dány závislostí na kandidátních klíčích
 - ne každá dekompozice do BCNF zachovává závislosti \Rightarrow potom stačí 3NF

5.2.2. Normální formy

- definuje požadavek na vlastnosti schématu relace z pohledu závislostí mezi atributy
- hierarchie normálních forem (Codd: 1NF až 3NF, BCNF, 4NF, 5NF), tj. n-tá normální forma musí splňovat podmínky (n-1) normální formy a něco navíc.
- Požadavky na návrh založený na normalizaci:
 - bezeztrátovost dekompozice
 - zachování závislostí
 - dosažení minimálně BCNF, resp. 3NF
- **První normální forma (1NF)**

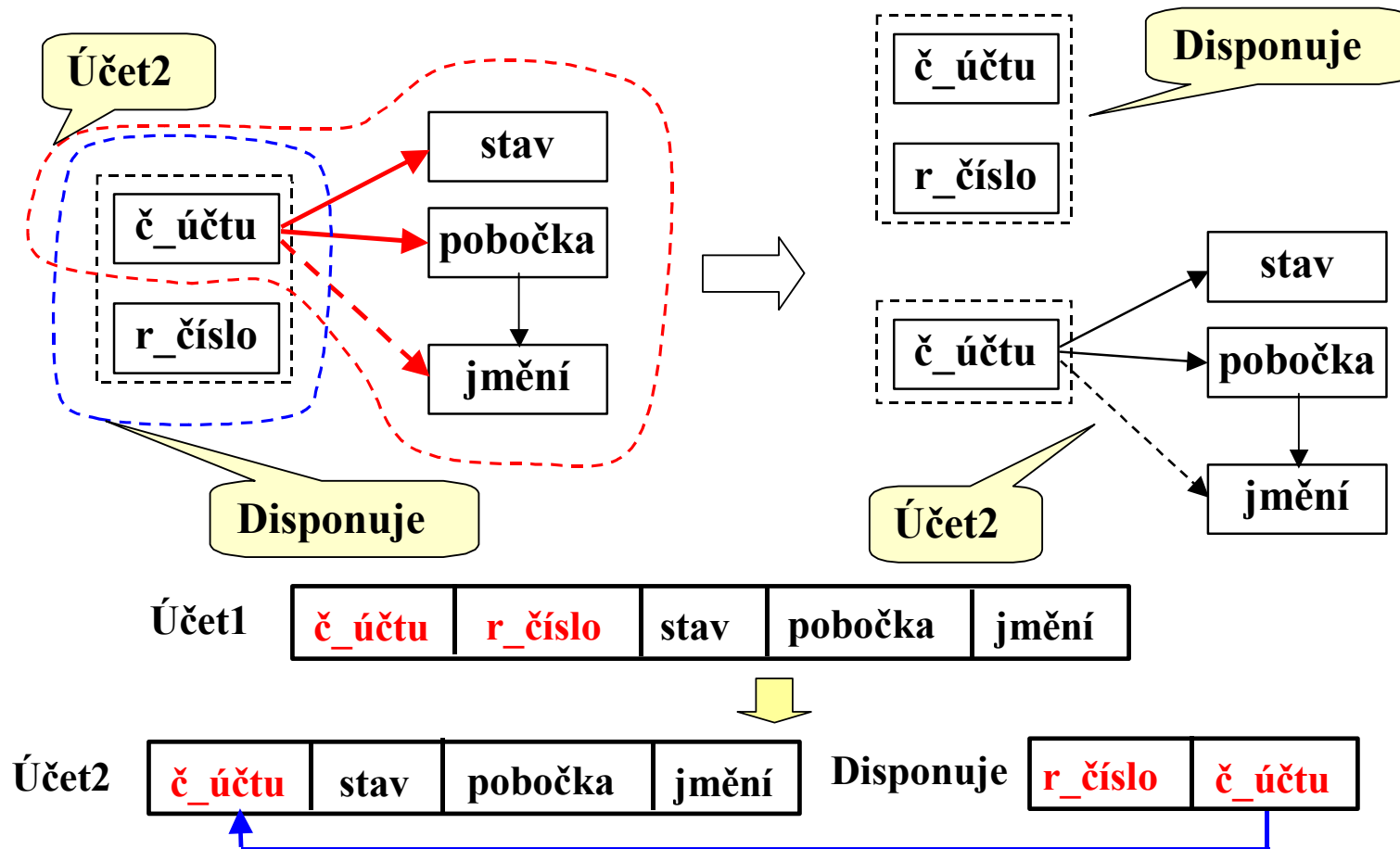
Relace je v *první normální formě*, právě když všechny její jednoduché domény obsahují pouze atomické hodnoty.

Př) Účet1(č_úctu, r_číslo, stav, pobočka, jmění)

- **Druhá normální forma (2NF)**

Relace je ve **druhé normální formě**, právě když je v 1NF a každý její neklíčový atribut, je plně funkčně závislý na každém kandidátním klíči.

Př)



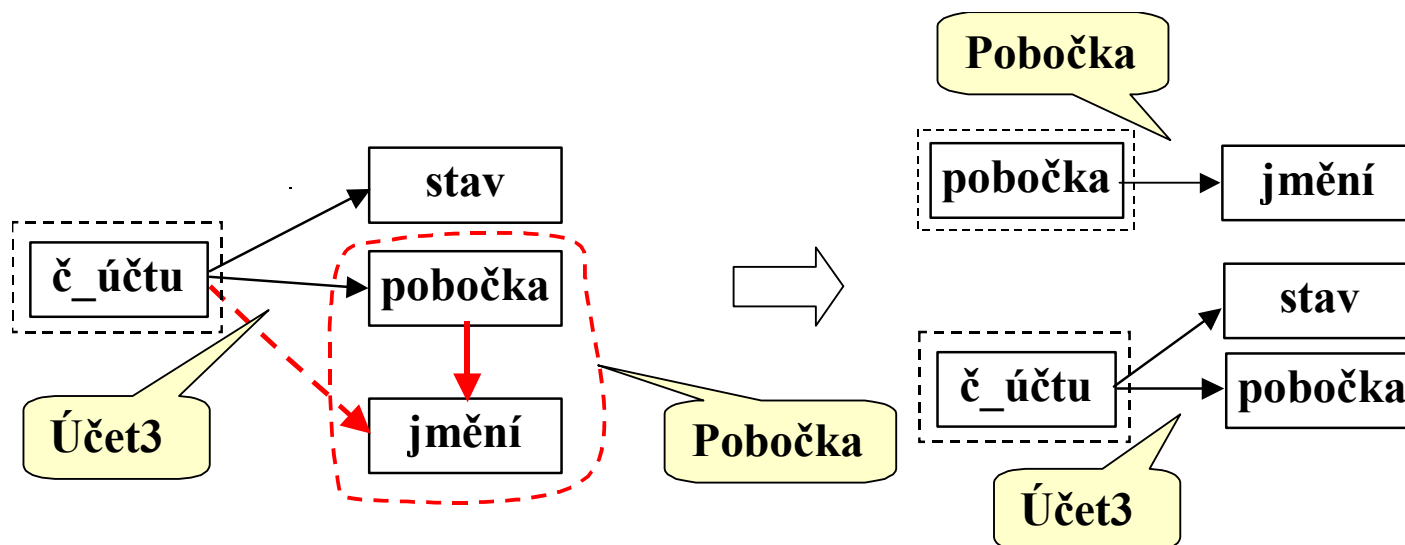
č_úctu	stav	pobočka	jmění
100	100000	Jánská	10000000
130	50000	Palackého	5000000
150	150000	Palackého	5000000

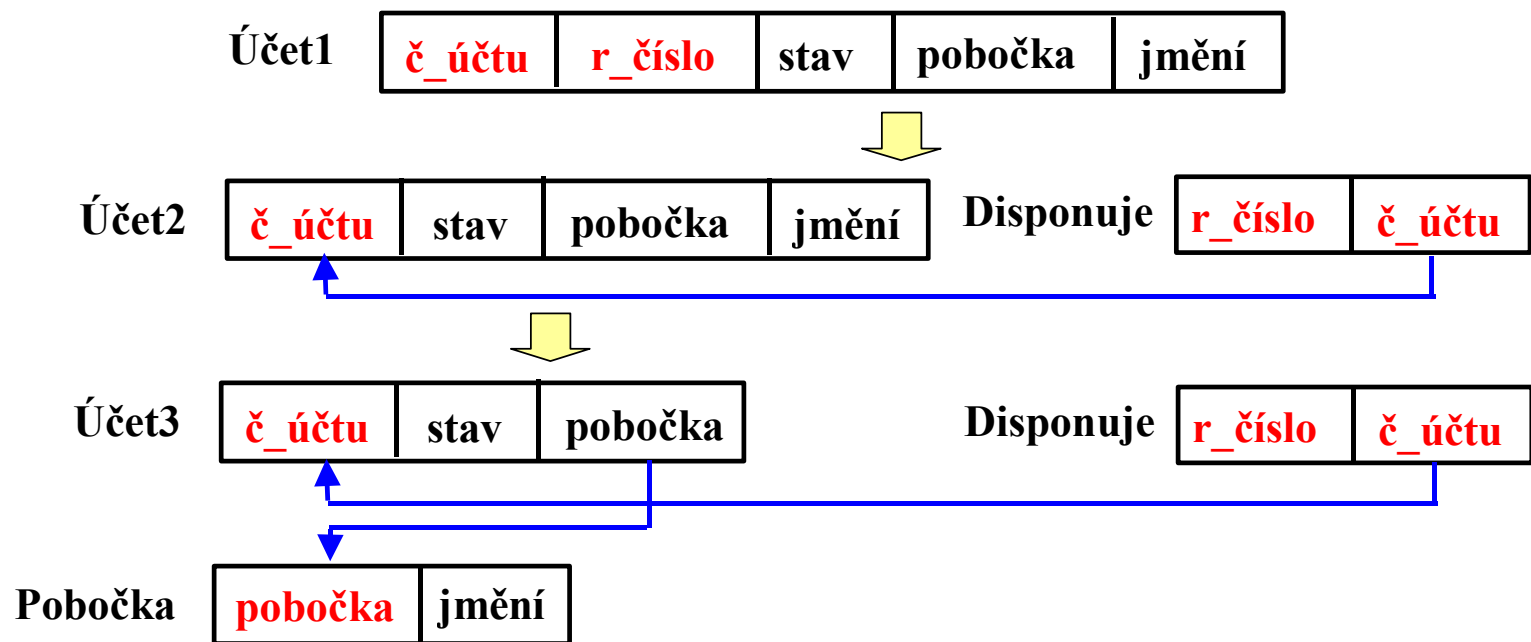
r_číslo	č_úctu
600528/0275	100
581015/9327	100
600528/0275	130
450205/3419	150

- **Třetí normální forma (3NF)**

Relace je ve *třetí normální formě*, právě když je ve 2NF a neexistuje žádný neklíčový atribut, který je tranzitivně závislý na některém kandidátním klíči.

Př)



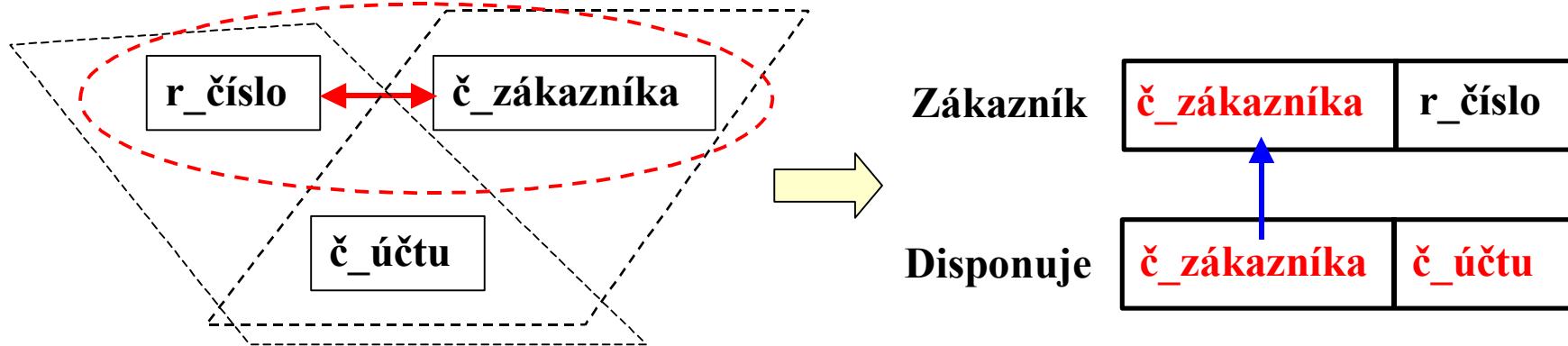


pobočka	jmění
Jánská	1000000
Palackého	500000

č_úctu	stav	pobočka
1035853	10000	Jánská
1348427	5000	Palackého
1529054	15000	Palackého

- **Boyce - Coddova normální forma (BCNF)**

Př) Disponuje(r_číslo, č_klienta, č_účtu)



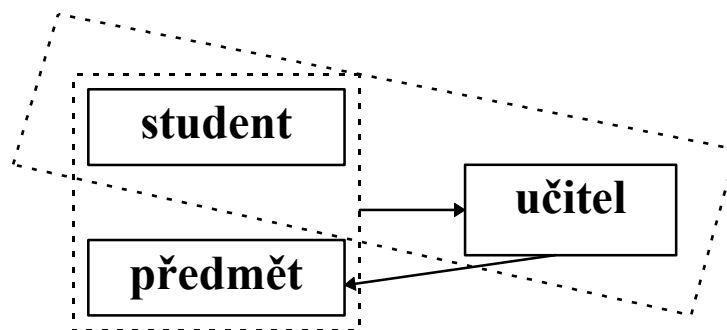
- může existovat několik kandidátních klíčů,
- kandidátní klíče mohou být složené,
- kandidátní klíče se mohou překrývat
- 3NF připouští tranzitivní závislosti mezi klíčovými atributy

Relace je v **Boyce-Codově normální formě**, jestliže pro každou netriviální funkční závislost $X \rightarrow Y$ je X superklíčem.

Př) Student_předmět_učitel(student, předmět, učitel)

Sémantika: - každého studenta učí z daného předmětu jen jeden učitel, každý učitel učí jen jeden předmět,
- jeden předmět může učit několik učitelů

student	předmět	učitel
Novák	matematika	Prof.Adam
Novák	fyzika	Doc.Kovář
Veselý	matematika	Prof.Adam
Veselý	fyzika	Doc.Zelený



- dekompozice Učí_předmět(učitel, předmět) a Učí_studenta(student, učitel) nezachovává závislosti v rámci relací (relace nejsou nezávislé)
→ zachování závislostí a dosažení BCNF není vždy splnitelné

Př) Zkoušky (student, předmět, pozice)

Sémantika: Dva studenti nemohou být na stejné pozici v jednom předmětu.

student	předmět	pozice
Novák	matematika	10
Veselý	matematika	7
Novák	fyzika	5
Veselý	fyzika	2

→ překrývající se kandidátní klíče ještě nemusí způsobovat problémy

- Čtvrtá normální forma (4NF)

Př) Účet – r_číslo je r.česuo vlastníka, klient může mít více účtů i adres.

Relace je v BCNF, ale obsahuje redundanci.

č_úctu	r_číslo	město
100	600528/0275	Praha
100	600528/0275	Brno
130	620726/1234	Praha
150	450205/3419	Ostrava

Relace je *ve čtvrté normální formě*, jestliže pro každou netriviální vícehodnotovou závislost $X \twoheadrightarrow Y$ je X superklíčem v R .

Př) Účet(č_úctu, r_číslo, město) \Rightarrow

Účet4(r_číslo, č_úctu), Adresa(r_číslo, město)

- je ve BCNF a všechny vícehodnotové závislosti jsou ve skutečnosti závislostmi funkčními.

Př) Učebnice pro předmět je předepsaná, neurčuje učitel

předmět	učitel	učebnice
fyzika	Doc.Kovář	Fyzika I
fyzika	Doc.Kovář	Sbírka úloh z fyziky
fyzika	Doc.Zelený	Fyzika I
fyzika	Doc.Zelený	Sbírka úloh z fyziky

- **Pátá normální forma (5NF, PJ/NF)**

Relace je **v páté normální formě**, jestliže pro každou netriviální závislost na spojení (R_1, R_2, \dots, R_n) je každá množina atributů R_i ($i=1, 2, \dots, n$) superklíčem v R .

- Normální forma schématu databáze

Návrh databáze je v n -té normální formě, je-li každá jeho relace (schéma) alespoň v n -té normální formě.

5.2.3. Obecný postup odstranění částečných a tranzitivních závislostí

- Převod do 2NF

$R(A, B, C, D, E)$

PRIMARY KEY (A,B)

$A \rightarrow C$

$A \rightarrow D$

$A \rightarrow E$

\Rightarrow

$R1(A, C, D, E)$

PRIMARY KEY (A)

$R2(A, B)$

PRIMARY KEY(A,B)

FOREIGN KEY(A) REFERENCES R1

- Převod do 3NF

$R(A, B, C, D)$

PRIMARY KEY (A)

$C \rightarrow D$

\Rightarrow

$R1(C, D)$

PRIMARY KEY (C)

$R2(A, B, C)$

PRIMARY KEY(A)

FOREIGN KEY(C) REFERENCES R1

Literatura

1. Silberschatz, A., Korth H.F, Sudarshan, S.: Database System Concepts. Fourth Edition. McGRAW-HILL. 2001, str. 79 – 131.
2. Pokorný, J.: Dotazovací jazyky. Science, Veletiny, 1994, str. 21 – 46.
3. Pokorný, J.: Databazová abeceda. Science, Veletiny, 1998, str. 35 – 38, 53 – 57, 201 – 204.