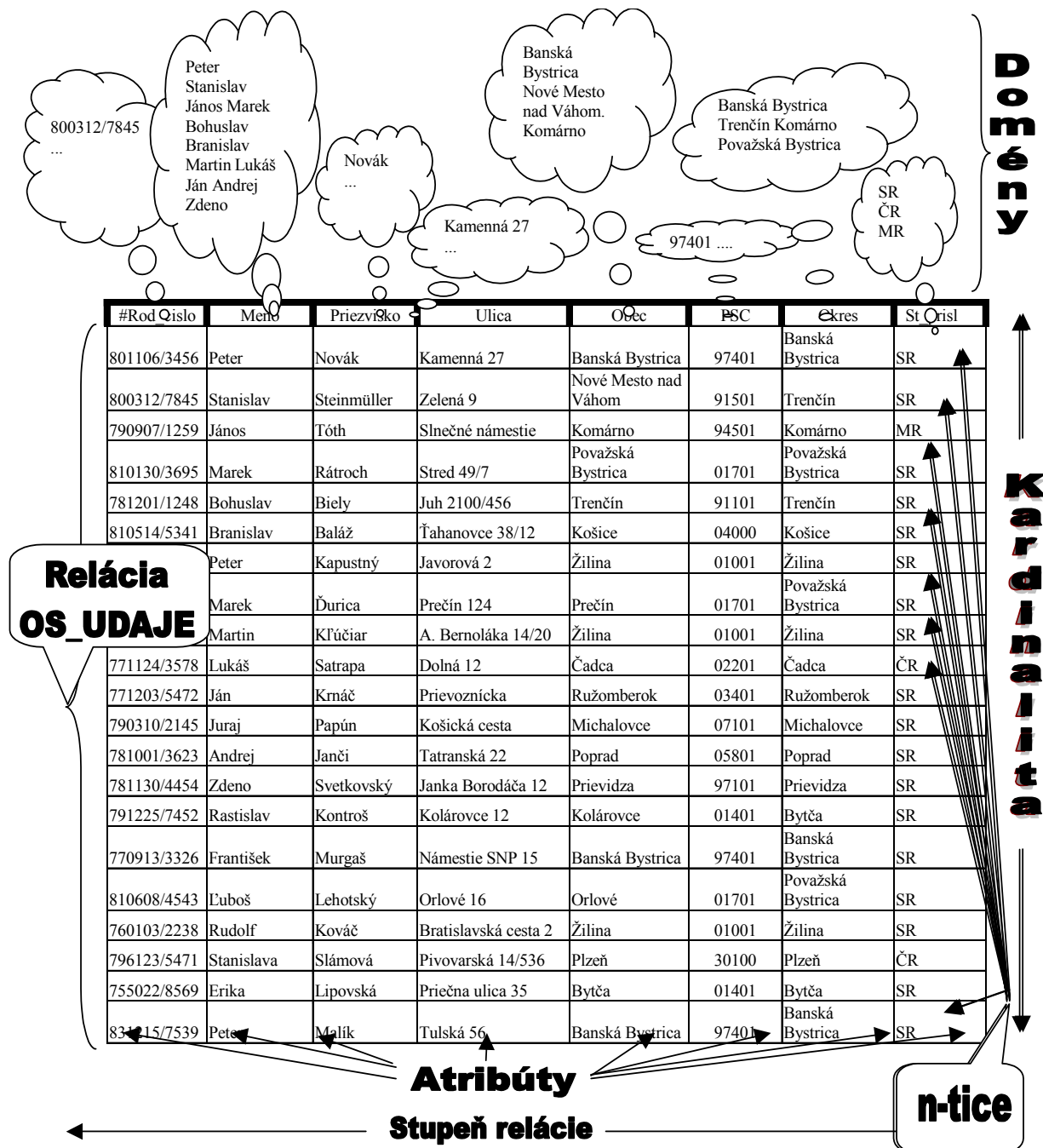


5. Relačný databázový systém

5.1 Úvod



br. 5.1 Relácia OS_UDAJE

5.2 Relačný dátový model

Relačný dátový model podľa pôvodnej definície vychádzal z nasledovných požiadaviek, ktoré neboli dosiahnuté v HDBS a ani v SDBS:

- Zabezpečiť vysoký stupeň dátovej nezávislosti.
- Zabezpečiť minimálnu redundanciu dát spolu s konzistenciou dát s podporou sémantiky jazyka.
- Sprístupniť databázu pomocou množinovo orientovaného neprocedurálneho jazyka.
- Umožniť jednoduchým spôsobom reštrukturalizáciu a rast dátového modelu.

Relačný dátový model je teda v podstate:

- založený na teórii relácií, množinovom pojme relácií a relačnej algebre
- všetky vzťahy reprezentuje vo forme tabuliek, pretože dvojrozmerné tabuľky stačia na modelovanie reálnych vzťahov
- kľúčovým pojmom používaným pri relačnom modelovaní je relácia

Sú to:

- **relácia** - zodpovedá tabuľke
- **n-tica** - zodpovedá riadku tabuľky
- **atribút** - zodpovedá stĺpcu tabuľky
- **primárny kľúč** - *jednoznačný* identifikátor v tabuľke, ktorý je reprezentovaný stĺpcom alebo skupinu stĺpcov
- **doména** - množina hodnôt, ktoré môže nadobúdať atribút

<i>Relačný pojem</i>	<i>Reprezentácia</i>	<i>Súborová analógia</i>
relácia	tabuľka	súbor
n-tica	riadok	záznam
atribút	stĺpec	položka

Obr. 5.2 Analógia pojmov

5.2.1 Atribút

Definícia- Atribút

Atribút relácie je stĺpec relácie označený unikátnym názvom a doménou.



Poznámka

Definícia atribútu je pojem, ktorý je v rôznej literatúre popisovaný rôznymi definíciami a často môže byť zamieňaný s pojmom atribútu, ktorý je definovaný v rámci ERA-modelovania. Je potrebné si ale uvedomiť, že ide o dve rôzne definície.

5.2.2 Domény

Od dátovej hodnoty je požadované, aby bola **atomická** tzn. ďalej nedeliteľná. C. J. Date [Date95] ju nazval **skalárom**, čo je termín pomerne zriedkavo sa vyskytujúci v databázovej literatúre, ale veľmi vhodný pre budúce pochopenia relačných databáz.

Definícia- Skalár

Skalár je hodnota, ktorá reprezentuje najmenšiu sémantickú jednotku dát.

Definícia- Doména

Doména je pomenovaná množina skalárnych hodnôt rovnakého typu.

Reprezentácia domén:

Z hľadiska praktickej implementácie doména môže byť definovaná ako:

- Interval
- Množina

Typy domén:

- Jednoduchá - sa týka jednoduchého atribútu a reprezentuje len hodnoty tvoriace prípustnú množinu hodnôt tohto atribútu
- Kompozitná - je definovaná ako kartézsky súčin hodnôt jednoduchých domén atribútov tvoriacich kompozitnú doménu

☐ **Označovanie domén:**

Doména, týkajúca sa atribútu *A*, sa často označuje ako **dom(A)**.

5.2.2.1 Význam domén

5.2.2.2 Definícia domén

Doménu môžeme popísať nasledovným príkazom:

CREATE DOMAIN meno_domény **AS** dátový_typ;

Kde

meno_domény	popisuje meno domény
dátový_typ	určuje prípustný dátový typ (Např. CHAR(n), NUMERIC, DATE, SMALLINT, apod.)

5.2.2.3 Pravidlá použitia domén

Pre správny návrh dátového modelu je žiaduce vykonať definíciu domén v dátovom modeli tak, aby boli dodržané tieto pravidlá:

Určíte množinu **D**, t. j. množinu všetkých domén danej databázy. Táto množina je uzavretou množinou, pričom do nej patria aj domény pre hodnoty, ktoré sú výsledkom predpokladaných aritmetických operácií.

1. Určite pre každú doménu $D_i \in D$, unárne operátory, ktoré je možné aplikovať na každý prvok $d_i \in D_i$
2. Pre každú dvojicu domén $D_i \in D$, $D_j \in D$ určite, ktoré binárne operátory je možné použiť medzi dvojicami prvkov $d_i \in D_i$ a $d_j \in D_j$ uvedených domén
3. Určite pre každý výraz, používajúci prvky domén, doménu všetkých možných výsledkov daného výrazu

Definovanie domén je proces, ktorý je súčasťou definovania dátového modelu a dodržanie vyššie uvedených pravidiel umožní predísť chybám, ktoré sa môžu vyskytnúť pri riešení praktických dotazov súvisiacich s operáciami relačnej algebry, alebo pri zabezpečovaní integrity a konzistencie databáz.

5.2.3 Relácia

Relačný model dát má len jediný konštrukt a tým je databázová relácia. Od matematickej relácie sa líši tým, že :

- Obsahuje schému relácie
- Prvky domén, z ktoré sú atomickými hodnotami

5.2.4 Pojmy

Definícia- Relačná schéma [Pokorný92]

Relačná schéma relácie R je $\Omega = \{A_1:D_1, A_2:D_2, \dots, A_n:D_n\}$, t. j. konečná množina dvojíc atribút : doména

Definícia- Relácia (podľa [Codd70])

Nech existuje množina domén $D = \{D_1, D_2, \dots, D_n\}$ nad množinou Ω , potom relácia je podmnožina karteziánskeho súčinu domén $D_1 \times D_2 \times \dots \times D_n$.

□ Poznámka

Určenie domény celej relácie

Ak existuje schéma relácie $\Omega = \{A, B\}$, tak potom pre doménu celej relácie môžeme určiť množinu prípustných hodnôt celej relácie nasledovne :

$$\text{dom}(\Omega) = \text{dom}(A) \times \text{dom}(B)$$

Definícia- Relácia (podľa [Date96])

Relácia R nad množinou domén D_1, D_2, \dots, D_n (nie nevyhnutne odlišnými) sa skladá z hlavičky a tela :

Hlavička – stála (nemenná) množina atribútov A_1, A_2, \dots, A_n takých, že každému atribútu A_i zodpovedá práve jedna z domén D_i kde $i=1, 2, \dots, n$ ($A_i:D_i$)

Telo – v čase sa meniaci množina n-tíc, kde každú n-ticu tvorí množina dvojíc

$$(A_i: v_{ij}), \text{ kde } i=1, 2, \dots, n \text{ a } j=1, 2, \dots, m$$

tzn. existuje jedna dvojica atribút : hodnota ($A_i: v_{ij}$) pre každý atribút A_i z hlavičky.

Pre ktorúkoľvek danú dvojicu atribút : hodnota ($A_i: v_{ij}$) platí, že v_{ij} je jednou z hodnôt danej domény D_i prislúchajúcej atribútu A_i .

□ Označovanie schém relácií:

Schéma relácie sa často označuje aj $R = \{ A_1, A_2, \dots, A_n \}$, $R(\Omega)$ alebo aj Ω , ak nám nezáleží na mene relácie.

Definícia- Kardinalita relácie

Kardinalita relácie je počet n-tíc relácie.

□ **Označovanie kardinality:**

Kardinalita relácie R sa často označuje **card(R)**, v niektorej literatúre je možné sa stretnúť aj s označením $|R|$

Definícia-Stupeň relácie

Ak existuje relačná schéma relácie $R = \{ A_1, A_2, \dots, A_n \}$, tak počet atribútov n udáva stupeň relácie.

□ **Označovanie stupňa relácie:**

Stupeň relácie R budeme označovať **degree(R)**, v niektorej literatúre je možné sa stretnúť aj s označením **arity(R)**

5.2.5 Vlastnosti relácie

Relácia má štyri základné vlastnosti :

1. **Neobsahuje duplicitné n-tice** - táto vlastnosť vychádza z faktu, že telo relácie je matematická množina (množina n-tíc), a podľa definície neobsahuje duplicitné prvky. Základným predpokladom pre splnenie tohto bodu je nutná existencia primárneho kľúča v každej relácii (pozri definíciu primárneho kľúča).
2. **n-tice sú neusporiadané** (zhora dolu) – n-tice tvoria matematickú množinu (nie je usporiadaná), hoci z praktického hľadiska je často vhodné udržiavať reláciu ako usporiadanú množinu. Z druhej strany neusporiadanosť umožňuje zjednodušiť a zrýchliť algoritmy pre prácu nad reláciou.
3. **m-tice atribútov sú neusporiadané** (zl'ava doprava) – m-tice tvoria matematickú množinu (nie je usporiadaná). Táto vlastnosť hovorí o tom, že k jednotlivým stĺpcom alebo k hodnotám atribútu pristupuje prostredníctvom identifikátoru (meno atribútu) a nie podľa pozície v stĺpci. To umožní tvorbu programov nezávislých od dát.
4. **hodnoty atribútov sú atomické** – túto vlastnosť môžeme vyjadriť aj inak (menej formálne): každému atribútu je priradená vždy len jedna hodnota a nie množina hodnôt. Je potrebné, aby všetky atribúty boli atomické, čím dosiahneme to, že v relácii sa nemôžu vyskytovať tzv. opakujúce sa skupiny atribútov, ktoré sa vyskytujú v nenormalizovaných záznamoch. Hodnota atribútu je vždy skalár a nie množina. "Relácia neobsahuje opakujúce sa skupiny". Relácii bez opakujúcich sa skupín hovoríme, že je **normalizovaná**.

Na začiatku je stav relácií nasledovný:

Vyucba

#os_cislo	predmet_vysledok	
	#cis_predmet	vysledok
S1	P1	3
	P2	2
	P3	
	P5	1
	P6	1
S2	P1	3
	P2	
S3	P2	2
S4	P2	2
	P3	3
	P4	

Zap_predmety

#os_cislo	#cis_predmet	vysledok
S1	P1	3
S1	P2	2
S1	P3	
S1	P5	1
S1	P6	1
S2	P1	3
S2	P2	
S3	P2	2
S4	P2	2
S4	P3	3
S4	P4	

Relácia Vyucba je reláciou druhého stupňa, ale prvok relácie predmet_vysledok, je vlastne sám binárnou reláciou. V relácii Zap_predmety je táto anomália odstránená pomocou procesu normalizácie, čím splňujeme štvrtú z požadovaných vlastností relácií.

Aplikujme na tieto relácie dve transakcie:

- Transakcia T_1 vkladá študentovi S5 výsledok s hodnotou 3 z predmetu P5 (operácia INSERT)
- Transakcia T_2 vkladá študentovi S4 výsledok s hodnotou 2 z predmetu P5 (operácia INSERT)

Prvá transakcia pridáva do oboch relácií jeden záznam (riadok) a zdanlivo sú tieto operácie ekvivalentné. Je však potrebné si uvedomiť, že v relácii Vyucba je potrebné najskôr zistiť, či existuje študent S5, či má zapísaný predmet P5 a vykonať vloženie výsledku. Možnosti, ktoré sa objavujú je niekoľko, ale v našom prípade ide o vloženie nového záznamu s neexistujúcim študentom S5.

V relácii Zap_predmety je situácia jednoduchšia, pretože sa pokúšame vložiť jediný riadok a našou úlohou je zabezpečiť prvú vlastnosť relácií, t. j. aby neobsahovala duplicitné n-tice.

Stav relácií po transakcii T₁:**Výucba**

#os_cislo	predmet_vysledok	
	#cis_predmet	vysledok
S1	P1	3
	P2	2
	P3	
	P5	1
	P6	1
S2	P1	3
	P2	
S3	P2	2
S4	P2	2
	P3	3
	P4	
S5	P5	3

Zap_predmety

#os_cislo	#cis_predmet	vysledok
S1	P1	3
S1	P2	2
S1	P3	
S1	P5	1
S1	P6	1
S2	P1	3
S2	P2	
S3	P2	2
S4	P2	2
S4	P3	3
S4	P4	
S5	P5	3

V druhej transakcii je situácia zložitejšia z hľadiska relácie Vyucba, pretože nejde o operáciu vloženia, ale o opravu záznamu týkajúceho sa študenta S4. V relácii Zap_predmety je situácia rovnaká ako v prvej transakcii.

Podobnú anomáliu z hľadiska použitia algoritmu pre vkladania dát objavíme aj pri pokuse vymazať niektoré záznamy, opravovať, alebo vypísať (operácie DELETE, UPDATE, SELECT)

Stav po transakcii T₂:**Výucba**

#os_cislo	predmet_vysledok	
	#cis_predmet	vysledok
S1	P1	3
	P2	2
	P3	
	P5	1
	P6	1
S2	P1	3
	P2	
S3	P2	2
S4	P2	2
	P3	3
	P4	
	P5	2
S5	P5	3

Zap_predmety

#os_cislo	#cis_predmet	vysledok
S1	P1	3
S1	P2	2
S1	P3	
S1	P5	1
S1	P6	1
S2	P1	3
S2	P2	
S3	P2	2
S4	P2	2
S4	P3	3
S4	P4	
S5	P5	3
S4	P5	2

5.2.6 Schémy relačnej databázy a relačná databáza

Definícia-Relačná schéma databázy

Relačná schéma databázy je potom definovaná dvojicou (R, I) , kde R je množina schém relácií a I je množina integritných obmedzení (IO).

Poznámka

Podrobnejšie by to bolo možné vyjadriť aj v tvare :

$$R = ((R_1, IO_1), (R_2, IO_2), \dots, (R_n, IO_n))$$

ak chceme zdôrazniť, že pre každú reláciu R_i v databáze R existuje vlastná množina integritných obmedzení.

Definícia-Prípustná relačná schéma databázy

Prípustnou relačnou databázou so schémou (R, I) nazývame množinu relácií R_1, R_2, \dots, R_n takých, že ich prvky (n -tice) vyhovujú I .

Definícia-Relačná databáza

Relačná databáza je množina v čase sa meniacich, normalizovaných relácií.

5.2.7 Klasifikácia relácií

1. **Podľa pôvodu** – charakterizujeme reláciu buď ako základnú, ktorá je súčasťou dátového modelu, alebo odvodenú, ktorá vzniká z dôvodu potreby špecifikovanej aplikáciou

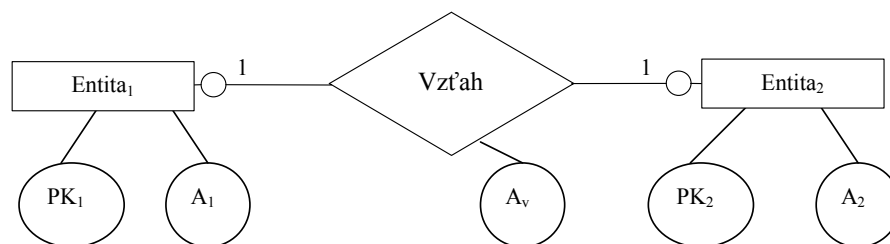
- a) **základné** – CREATE TABLE
- b) **odvodené**
 - *pohľad* – CREATE VIEW
 - *snímka* - CREATE SNAPSHOT
 - *dočasné tabuľky*

2. **Podľa spôsobu uloženia**

- a) **perzistentné** – základné, snímky
- b) **dočasné** – medzivýsledky, pohľady

3. **Podľa mena**

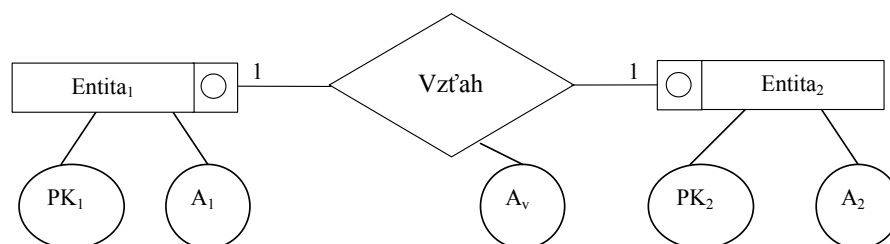
- a) **pomenované** – základné, pohľady, snímky
- b) **nepomenované** – dotaz, dočasné (medzivýsledok)

Transformácie Entitného diagramu na Dátový diagram**Vzťah 1:1**
 $R_1(\#PK_1, A_1)$
 $R_2(\#PK_2, A_2, PK_1, A_v)$

alebo

 $R_1(\#PK_1, A_1, PK_2, A_v)$
 $R_2(\#PK_2, A_2)$

alebo

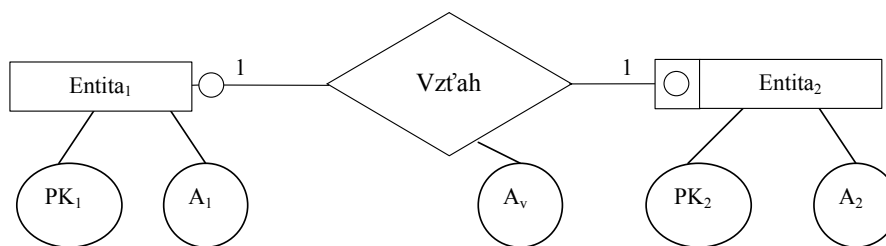
 $R(\#PK_1, A_1, PK_2, A_v)$

 $R_1(\#PK_1, A_1)$
 $R_2(\#PK_2, A_2, PK_1, A_v)$

alebo

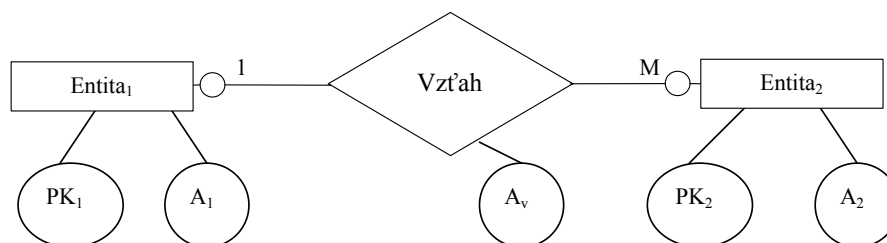
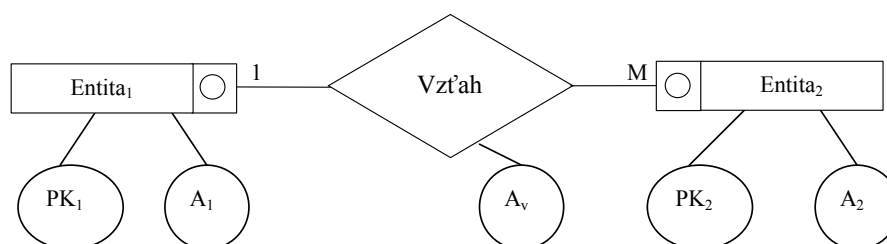
 $R_1(\#PK_1, A_1, PK_2, A_v)$
 $R_2(\#PK_2, A_2)$

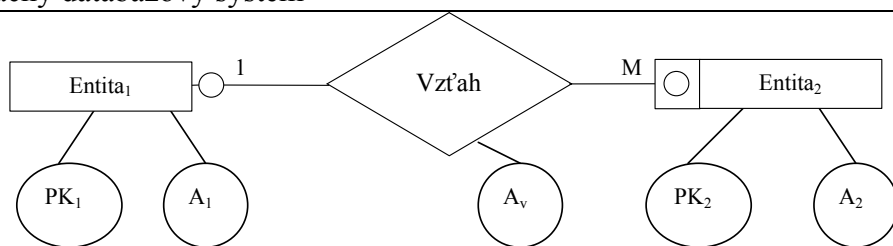
alebo

 $R(\#PK_1, A_1, PK_2, A_v)$

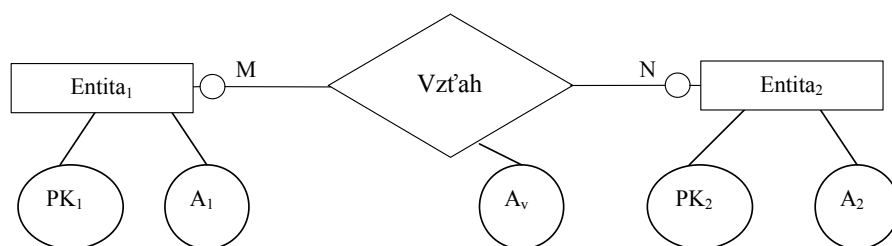

 $R_1(\#PK_1, A_1)$
 $R_2(\#PK_2, A_2, PK_1, A_v)$

alebo

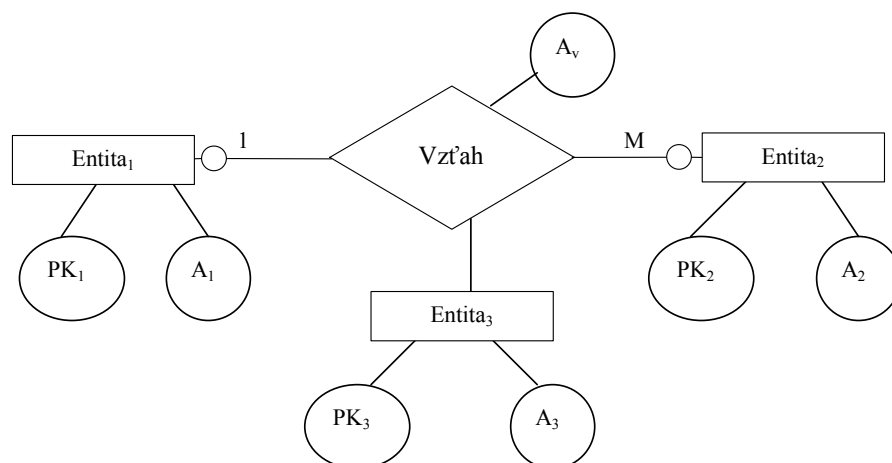
 $R(\#PK_1, A_1, PK_2, A_v)$
Vzťah 1:M
 $R_1(\#PK_1, A_1)$
 $R_2(\#PK_2, A_2, PK_1, A_v)$

 $R_1(\#PK_1, A_1)$
 $R_2(\#PK_2, A_2, PK_1, A_v)$

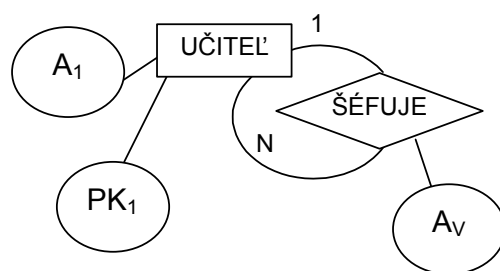

 $R_1(\#PK_1, A_1)$
 $R_2(\#PK_2, A_2, PK_1, A_v)$

Vzťah M:N


 $R_1(\#PK_1, A_1)$
 $R_2(\#PK_2, A_2)$
 $R_3(\#PK_1, \#PK_2, A_v)$

Ternárny vzťah


 $R_1(\#PK_1, A_1)$
 $R_2(\#PK_2, A_2)$
 $R_3(\#PK_3, A_3)$
 $R_4(\#PK_1, \#PK_2, \#PK_3, A_v)$

Rekurzívny vzťah

Učiteľ(#PK₁, A₁, UC_PK₁, A_v)