KATEDRA INFORMATIKY
UNIVERZITA PALACKÉHO V OLOMOUCI
www.inf.upol.cz

Jan Laštovička
https://www.inf.upol.cz/lide/jan-lastovicka
jan.lastovicka@upol.cz
17. listopadu 12, 771 46 Olomouc

Databáze ⋄ poznámky k přednášce

5. Relační algebra

verze z 15. října 2023

1 Operace s relacemi

Nejprve si zopakujeme operace s relacemi a ukážeme si vztahy mezi nimi.

1.1 Množinové operace

Vezměme dvě relace $\mathcal{D}_1$ a $\mathcal{D}_2$ nad relačním schématem R.

Sjednocení $\mathcal{D}_1$ a $\mathcal{D}_2$ je množina

$$\mathcal{D}_1 \cup \mathcal{D}_2 = \{r \mid r \in \mathcal{D}_1 \text{ nebo } r \in \mathcal{D}_2\}.$$

Věta 1. *Sjednocení $\mathcal{D}_1 \cup \mathcal{D}_2$ je relace nad R.*

Důkaz. Konečnost $\mathcal{D}_1 \cup \mathcal{D}_2$ plyne z konečnosti $\mathcal{D}_1$ a $\mathcal{D}_2$. Dále musíme ukázat, že každý prvek $r \in \mathcal{D}_1 \cup \mathcal{D}_2$ je n-tice nad R. Vezměme libovolný $r \in \mathcal{D}_1 \cup \mathcal{D}_2$. Víme, že $r \in \mathcal{D}_1$ nebo $r \in \mathcal{D}_2$. Předpokládejme, že $r \in \mathcal{D}_1$. Protože $\mathcal{D}_1$ je relace nad R musí být $r \in \mathcal{D}_1$ n-ticí nad R. Podobně pro $r \in \mathcal{D}_2$. □

Musí být sjednocení dvou relací vždy relace?

Věta 2. *Pokud je $\mathcal{D}$ relace nad R, pak libovolná její podmnožina je opět relace nad R.*

Důkaz. Nechť $\mathcal{D}_2 \subseteq \mathcal{D}$. Podmnožina konečné množiny musí být konečná. Tedy $\mathcal{D}_2$ je konečná množina. Zvolme libovolné $r \in \mathcal{D}_2$. Protože $\mathcal{D}_2 \subseteq \mathcal{D}$, platí $r \in \mathcal{D}$. Tedy r je n-tice nad R. Ukázali jsme, že každý prvek $\mathcal{D}_2$ je n-tice nad R. □

Průnik $\mathcal{D}_1$ a $\mathcal{D}_2$ je množina

$$\mathcal{D}_1 \cap \mathcal{D}_2 = \{t \mid t \in \mathcal{D}_1 \text{ a } t \in \mathcal{D}_2\}.$$

Věta 3. *Množina $\mathcal{D}_1 \cap \mathcal{D}_2$ je relace nad R.*

Důkaz. Tvrzení plyne z Věty 2 a $\mathcal{D}_1 \cap \mathcal{D}_2 \subseteq \mathcal{D}_1$ (a také z $\mathcal{D}_1 \cap \mathcal{D}_2 \subseteq \mathcal{D}_2$). □

Co je výsledkem průniku dvou relací nad různými schématy?

Rozdíl $\mathcal{D}_1$ a $\mathcal{D}_2$ je množina

$$\mathcal{D}_1 - \mathcal{D}_2 = \{r \mid r \in \mathcal{D}_1 \text{ a } r \notin \mathcal{D}_2\}.$$

Věta 4. *Rozdíl $\mathcal{D}_1 - \mathcal{D}_2$ je relace nad R.*

Důkaz. Opět plyne z Věty 2 a $\mathcal{D}_1 - \mathcal{D}_2 \subseteq \mathcal{D}_1$. □

Co vznikne rozdílem dvou relací nad různými schématy?

Je pro relaci $\mathcal{D}$ nad R množina $\mathrm{Tupl}(R) - \mathcal{D}$ vždy relace nad R?

Věta 5. *Pro množiny A, B platí $A - (A - B) = A \cap B$.*

Důkaz. Zvolme libovolné x. Máme $x \in A - (A - B)$, právě když $x \in A$ a $x \notin A - B$, právě když $x \in A$ a ($x \notin A$ nebo $x \in B$), právě když ($x \in A$ a $x \notin A$) nebo ($x \in A$ a $x \in B$), právě když $x \in A$ a $x \in B$, právě když $x \in A \cap B$. □

Vidíme, že průnik relací můžeme vyjádřit pomocí rozdílu.

1.2 Restrikce relace

Je dáno schéma R. *Podmínka* θ nad R je definována následovně.

- jsou-li $y \in R$ a $v \in D_y$, pak $(y = v)$,$(v = y)$ jsou podmínky nad R,
- jsou-li$y_1, y_2 \in R$ a $D_{y_1} = D_{y_2}$, pak $(y_1 = y_2)$ je podmínka nad R,
- jsou-li θ_1 a θ_2 podmínky nad R, pak $(\theta_1 \wedge \theta_2)$, $(\theta_1 \vee \theta_2)$, $\neg\theta_1$ jsou také podmínky nad R.

Nejvíce vnější závorky kolem podmínek budeme vynechávat.

n-tice t nad R *splňuje* podmínku θ nad R, jestliže:

- θ je tvaru $(y = v)$ nebo $(v = y)$ a přitom platí $t(y) = v$,
- θ je tvaru $(y_1 = y_2)$ a $t(y_1) = t(y_2)$,
- θ je tvaru $(\theta_1 \wedge \theta_2)$ a přitom t splňuje obě podmínky θ_1 a θ_2,
- θ je tvaru $(\theta_1 \vee \theta_2)$ a přitom t splňuje θ_1 nebo θ_2,
- θ je tvaru $\neg\theta_1$ a přitom t nesplňuje podmínku θ_1.

Restrikce realce $\mathcal{D}$ nad schématem R podle podmínky θ nad R je množina

$$\sigma_\theta(\mathcal{D}) = \{t \in \mathcal{D} \mid t \text{ splňuje podmínku } \theta\}.$$

Jistě $\sigma_\theta(\mathcal{D}) \subseteq \mathcal{D}$ a proto podle Věty 2 je $\sigma_\theta(\mathcal{D})$ relací na R.

Věta 6. *Pro relaci $\mathcal{D}$ nad schématem R a podmínky θ_1 a θ_2 nad R platí:*

1. $\sigma_{\theta_1 \wedge \theta_2}(\mathcal{D}) = \sigma_{\theta_1}(\mathcal{D}) \cap \sigma_{\theta_2}(\mathcal{D})$,
2. $\sigma_{\theta_1 \vee \theta_2}(\mathcal{D}) = \sigma_{\theta_1}(\mathcal{D}) \cup \sigma_{\theta_2}(\mathcal{D})$,
3. $\sigma_{\neg\theta_1}(\mathcal{D}) = \mathcal{D} - \sigma_{\theta_1}(\mathcal{D})$.

Důkaz. Zvolme libovolnou n-tici t nad R.

1. $r \in \sigma_{\theta_1 \wedge \theta_2}(\mathcal{D})$, právě když $r \in \mathcal{D}$ a r splňuje $\theta_1 \wedge \theta_2$, právě když $r \in \mathcal{D}$ a r splňuje θ_1 a r splňuje θ_2, právě když $r \in \sigma_{\theta_1}(\mathcal{D})$ a $r \in \sigma_{\theta_2}(\mathcal{D})$, právě když $r \in \sigma_{\theta_1}(\mathcal{D}) \cap \sigma_{\theta_2}(\mathcal{D})$,
2. podobně $r \in \sigma_{\theta_1 \vee \theta_2}(\mathcal{D})$, právě když $r \in \mathcal{D}$ a (r splňuje θ_1 nebo r splňuje θ_2), právě když $r \in \sigma_{\theta_1}(\mathcal{D}) \cup \sigma_{\theta_2}(\mathcal{D})$,
3. nakonec $r \in \sigma_{\neg\theta_1}(\mathcal{D})$, právě když $r \in \mathcal{D}$ a r nesplňuje θ_1, právě když $r \in \mathcal{D} - \sigma_{\theta_1}(\mathcal{D})$.

□

1.3 Projekce relace

Pro n-tici r nad schématem R a $S \subseteq R$ se n-tice $r(S) = \{\langle y, r(y)\rangle \mid y \in S\}$ nad schématem S nazývá *projekce* n-tice r na S.

Vezměme relaci $\mathcal{D}$ na schématu R a zvolme $S \subseteq R$, pak se množina

$$\pi_S(\mathcal{D}) = \{r(S) \mid r \in \mathcal{D}\}$$

nazývá se *projekce* relace $\mathcal{D}$ na S.

Věta 7. *Projece $\pi_S(\mathcal{D})$ relace $\mathcal{D}$ na S je relací na S.*

Důkaz. $\pi_S(\mathcal{D})$ je tvořena projekcemi n-tic $r \in \mathcal{D}$ na S, které jsou n-ticemi nad S. Konečnost $\pi_S(\mathcal{D})$ plyne z konečnosti $\mathcal{D}$. □

1.4 Spojení relací

Pro relaci $\mathcal{D}_1$ nad schématem R a relaci $\mathcal{D}_2$ nad schématem S se množina

$$\mathcal{D}_1 \bowtie \mathcal{D}_2 = \{t \in \mathrm{Tupl}(R \cup S) \mid t(R) \in \mathcal{D}_1 \text{ a } t(S) \in \mathcal{D}_2\}$$

nazývá (*přirozené*) *spojení* relací $\mathcal{D}_1$ a $\mathcal{D}_2$.

Věta 8. *Spojení $\mathcal{D}_1 \bowtie \mathcal{D}_2$ relací $\mathcal{D}_1$ a $\mathcal{D}_2$ je relace nad $R \cup S$.*

Důkaz. Z konstrukce vyplývá, že $\mathcal{D}_1 \bowtie \mathcal{D}_2$ tvoří n-tice nad $R \cup S$. Konečnost $\mathcal{D}_1 \bowtie \mathcal{D}_2$ je důsledkem konečností $\mathcal{D}_1$ i $\mathcal{D}_2$. □

Vezměme dvě n-tice $r \in \mathrm{Tupl}(R)$ a $s \in \mathrm{Tupl}(S)$. Pokud $r(R \cap S) = s(R \cap S)$, pak n-tice r a s nazýváme *spojitelné* a $r \cup s$ je n-tice nad $R \cup S$ nazývaná *spojení* n-tic r a s, které značíme rs.

Věta 9. *Platí* $\mathcal{D}_1 \bowtie \mathcal{D}_2 = \{rs \mid r \in \mathcal{D}_1 \text{ a } s \in \mathcal{D}_2 \text{ a } r \text{ a } s \text{ jsou spojitelné}\}$

Důkaz. Vezměme libovolnou n-tici t nad $R \cup S$. Dostáváme, že
$t \in \mathcal{D}_1 \bowtie \mathcal{D}_2 = \{t \in \mathrm{Tupl}(R \cup S) \mid t(R) \in \mathcal{D}_1 \text{ a } t(S) \in \mathcal{D}_2\}$,
přávě když $t \in \mathrm{Tupl}(R \cup S)$ a $t(R) \in \mathcal{D}_1$ a $t(S) \in \mathcal{D}_2$,
právě když $t \in \mathrm{Tupl}(R \cup S)$ a $t(R) \in \mathcal{D}_1$ a $t(S) \in \mathcal{D}_2$ a existují r, s tak, že $r = t(R)$ a $s = t(R)$ a $rs = t$ a $r(R \cap S) = s(R \cap S)$,
právě když existují $r \in \mathcal{D}_1$ a $s \in \mathcal{D}_2$ tak, že $t = rs$ a r a s jsou spojitelné,
právě když $t \in \{rs \mid r \in \mathcal{D}_1 \text{ a } s \in \mathcal{D}_2 \text{ a } r \text{ a } s \text{ jsou spojitelné}\}$ □

Spojení relací, kde R a S jsou disjunktní, se nazývá (*kartézským*) *součinem* a značíme jej též $\mathcal{D}_1 \times \mathcal{D}_2$.

Věta 10. *Pokud* $R = S$, *pak* $\mathcal{D}_1 \bowtie \mathcal{D}_2 = \mathcal{D}_1 \cap \mathcal{D}_2$.

Důkaz. Z $R = S$ plyne, že $\mathcal{D}_2$ je relace nad R a i $\mathcal{D}_1 \bowtie \mathcal{D}_2$ je relace nad R.

Pro libovolnou n-tici r nad R máme $r \in \mathcal{D}_1 \bowtie \mathcal{D}_2$, právě když $r(R) \in \mathcal{D}_1$ a $t(R) \in \mathcal{D}_2$, právě když, $r \in \mathcal{D}_1$ a $r \in \mathcal{D}_2$, právě když $r \in \mathcal{D}_1 \cap \mathcal{D}_2$. □

1.5 Přejmenování atributů

Uvažujme bijektivní zobrazení $h\colon R \to S$ mezi schématy R, S takové, že $D_y = D_{h(y)}$ pro každé $y \in R$. Zobrazení h nazýváme *přejmenování atributů.*

Pro n-tici r nad R se n-tice

$$\rho_h(r) = \{\langle h(y), v\rangle \mid \langle y, v\rangle \in r\}$$

nad S nazývá *přejmenování* n-tice t podle h.

Pro relaci $\mathcal{D}$ nad R se množina

$$\rho_h(\mathcal{D}) = \{\rho_h(t) \mid t \in \mathcal{D}\}$$

nazývá *přejmenování* $\mathcal{D}$ podle h.

Věta 11. *Přejmenování* $\rho_h(\mathcal{D})$ *relace* $\mathcal{D}$ *podle* h *je relace nad* S.

Důkaz. Kardinalita množiny $\rho_h(\mathcal{D})$ je stejná jako $\mathcal{D}$. Pro každé $r \in \mathcal{D}$ je $\rho_h(r)$ n-tice nad S. □

Předpokládejme, že $R = \{y_1, \ldots, y_n, s_1, \ldots, s_m\}$, kde $h(s_i) = s_i$ pro každé $1 \leq i \leq m$. Pak $\rho_h(\mathcal{D})$ můžeme značit $\rho_{h(y_1) \leftarrow y_1, \ldots, h(y_n) \leftarrow y_n}$.

2 Relační algebra

Relační algebra je tvořena operacemi:

1. sjednocení ($\cup$)
2. průnik ($\cap$)
3. rozdíl ($-$)
4. restrikce (σ)
5. projekce (π)
6. spojení ($\bowtie$)
7. přejmenování atributů (ρ)

Relační model dat představil E. F. Codd v roce 1970. Navrhl relační algebru jako základ dotazovacích jazyků. V originální relační algebře chybí přejmenování atributů (atributy v relaci měly pevně dané pořadí) a navíc zde byl kartézský součin a relační dělení. Víme, že kartézský součin je speciální případ spojení. Relačnímu dělení je věnována část níže.

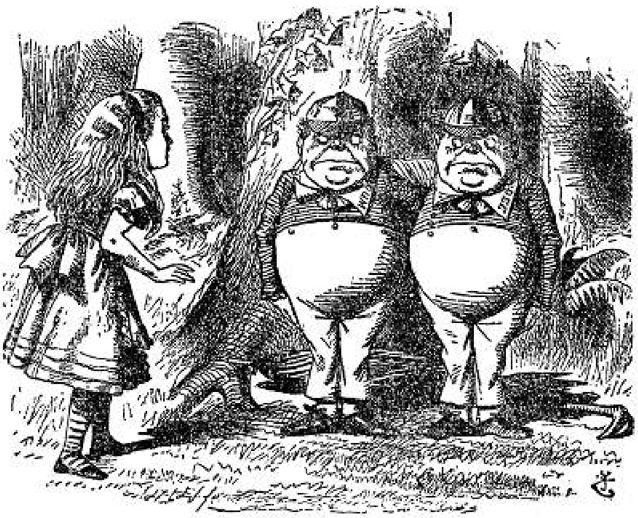
SQL vychází z relačního modelu, ale některé jeho principy porušuje. Například neumožňuje pracovat s relacemi s nad prázdným schématem. Jak víme tabulky přinášení závažnější prohřešky proti relačnímu modelu (můžou obsahovat null hodnoty, duplicitní řádky a názvy sloupců nemusí být jedinečné).

Christopher J. Date a Hugh Darwen navrhli v třetím manifestu (The Third Manifesto) publikovaném v roce 1995 požadavky na jazyk respektující relační model. Jejich specifikace jazyka se nazývá D. Manifest konkrétněji popisuje jazyk Tutorial D, který specifikacím D vyhovuje. Známá implementace jazyka Tutorial D se jmenuje Rel.

Relační kalkul je dotazovací jazyk, který vychází z predikátové logiky. Dotaz formulujeme pomocí relačních symbolů (odpovídají relačním proměnným), logických spojek (disjunkce, implikace,...) a kvantifikátorů (existenční a obecný). Relační kalkul pracuje s vnitřními strukturami relací. Dělíme jej na dva typy podle možných hodnot objektových proměnných. Za prvé *n-ticový relační kalkul*, kde objektové proměnné nabývají hodnot n-tic. Za druhé *doménový relační kalkul*. Zde hodnoty objektových proměnných jsou přímo domén atributů. Oba relační kalkuly a relační algebra mají stejnou vyjadřovací sílu. Tím se rozumí, že libovolný dotaz v jednom jazyku jsme schopni přeformulovat do ostatních jazyků tak, že výsledky všech dotazů jsou vždy stejné.

3 Relace nad prázdným schématem

Prázdná množina atributů $\emptyset$ je také relační schéma. Existuje jen jediná n-tice $t_0 \in \mathrm{Tupl}(\emptyset)$ nad $\emptyset$ a to prázdná množina $\emptyset$. Existují dvě relace nad schématem $\emptyset$:



Obrázek 1: Tweedledum a Tweedledee (česky dvojčata Tydliták a Tydlitek) z knihy *Through the Looking-Glass* (česky *Za zrcadlem a co tam Alenka našla*) od Lewise Carrolla.

prázdná relace $\emptyset$ a množina obsahující pouze n-tici t_0 tedy množina $\{t_0\} = \{\emptyset\}$. První se jmenuje `DUM` a druhá `DEE`. Jména relací vychází z anglických jmen postav Tweedledum a Tweedledee (česky dvojčata Tydliták a Tydlitek) z knihy *Za zrcadlem a co tam Alenka našla* od Lewise Carrolla. Postavy jsou zachyceny na Obrázku 1. Relace `DUM` reprezentuje *nepravdu* a relace `DEE` *pravdu*.

Uvažujme například relaci $\mathcal{D}$ nad R a podmínku θ nad R. Chceme zjistit, zda existuje n-tice $r \in \mathcal{D}$, která splňuje podmínku θ. Můžeme nejprve provést restrikci relace $\mathcal{D}$ vzhledem k θ a poté udělat projekci na prázdnou množinu atributů, tedy zjistit hodnotu výrazu $\pi_\emptyset(\sigma_\theta(\mathcal{D}))$. Pokud výsledkem je relace `DEE`, pak odpověď je ano a pokud `DUM`, pak je odpověď ne.

V SQL neexistují relace nad prázdným schématem. Zde tedy nenajdeme ani relaci `DUM` ani relaci `DEE`. Musíme se spokojit s tím, že každá prázdná relace reprezentuje nepravdu a každá neprázdná relace reprezentuje pravdu.

4 Relační dělení

Uvažujme dvě schémata R a S, která jsou disjunktní. Pro relaci $\mathcal{D}_1$ na $R \cup S$ a relaci $\mathcal{D}_2$ na S se relace

$$\mathcal{D}_1 \div \mathcal{D}_2 = \{r \in \pi_R(\mathcal{D}_1) \mid \text{pro každé } s \in \mathcal{D}_2 \text{ platí, že } rs \in \mathcal{D}_1\} \tag{1}$$

nad R nazývá *podíl* $\mathcal{D}_1$ a $\mathcal{D}_2$.

Věta 12. *Platí, že* $\mathcal{D}_1 \div \mathcal{D}_2 = \pi_R(D_1) - \pi_R((\pi_R(\mathcal{D}_1) \times \mathcal{D}_2) - \mathcal{D}_1)$.

Důkaz. Zvolme libovolné $r \in \mathrm{Tupl}(R)$. Máme
$r \in \pi_R(D_1) - \pi_R((\pi_R(\mathcal{D}_1) \times \mathcal{D}_2) - \mathcal{D}_1)$,

právě když $r \in \pi_R(D_1)$ a $r \notin \pi_R((\pi_R(\mathcal{D}_1) \times \mathcal{D}_2) - \mathcal{D}_1)$,
právě když $r \in \pi_R(D_1)$ a pro každé $s \in \mathrm{Tupl}(S)$ platí, že $rs \notin (\pi_R(\mathcal{D}_1) \times \mathcal{D}_2) - \mathcal{D}_1$,
právě když $r \in \pi_R(D_1)$ a pro každé $s \in \mathrm{Tupl}(S)$ platí, že $rs \notin \pi_R(\mathcal{D}_1) \times \mathcal{D}_2$ nebo $rs \in \mathcal{D}_1$,
právě když $r \in \pi_R(D_1)$ a pro každé $s \in \mathrm{Tupl}(S)$ platí, že $r \notin \pi_R(\mathcal{D}_1)$ nebo $s \notin \mathcal{D}_2$ nebo $rs \in \mathcal{D}_1$,
právě když $r \in \pi_R(D_1)$ a pro každé $s \in \mathrm{Tupl}(S)$ platí, že $r \in \pi_R(\mathcal{D}_1)$ a $s \in \mathcal{D}_2$ implikuje $rs \in \mathcal{D}_1$,
právě když $r \in \pi_R(D_1)$ a pro každé $s \in \mathrm{Tupl}(S)$ platí, že $s \in \mathcal{D}_2$ implikuje $rs \in \mathcal{D}_1$
právě když $r \in \pi_R(D_1)$ a pro každé $s \in \mathcal{D}_2$ platí, že $rs \in \mathcal{D}_1$
právě když $r \in \{r \in \pi_R(\mathcal{D}_1) \mid \text{pro každé } s \in \mathcal{D}_2 \text{ platí, že } rs \in \mathcal{D}_1\}$. □

Relační dělení umíme vyjádřit v SQL. Uvažujme například relační proměnné

```
# TABLE liked;
 person |    movie
--------+-------------
 Anna   | Blue Velvet
 Anna   | Eraserhead
 Bert   | Blue Velvet
 Bert   | The Matrix
 Cyril  | Blue Velvet
 Cyril  | Eraserhead
 Cyril  | The Matrix
(7 rows)

# TABLE lynch_movies;
    movie
-------------
 Blue Velvet
 Eraserhead
(2 rows)
```

Relační proměnná `liked` má charakteristickou vlastnost „Osoba jménem `person` má ráda film `movie`“. Omezujeme se na určitou množinu lidí a filmů a pro jednoduchost předpokládáme, že každý film je jednoznačně určen jménem. Relační proměnná `lynch_movies` má charakteristickou vlastnost „Film `movie` režíroval David Lynch.“ Relačním podílem `liked` a `lynch_movies` získáme relaci:

```
# ( SELECT DISTINCT person FROM liked )
    EXCEPT
  ( SELECT   DISTINCT person
    FROM ( ( SELECT DISTINCT liked.person, lynch_movies.*
             FROM  liked, lynch_movies )
             EXCEPT
           ( TABLE liked ) )  );
```

```
 person
--------
 Anna
 Cyril
(2 rows)
```

která má charakteristickou vlastnost „Osoba `person` má ráda všechny Lynchovy filmy.“

5 Klauzule SELECT výrazu

Příkazy v SQL se skládají z částí, které se nazývají *klauzule*. Klauzule se většinou jmenují podle klíčového slova, které klauzuli uvozuje. Například výraz SELECT může mít tuto formu:

```
( SELECT DISTINCT attributes FROM relations WHERE condition )
```

a skládá se z klauzulí: SELECT, DISTINCT, FROM a WHERE.

Klauzule SELECT má tvar:

```
SELECT attributes
```

Klauzule DISTINCT se vkládá do klauzule SELECT.

Klauzule FROM má tvar

```
FROM relations
```

Klauzule WHERE má tvar

```
WHERE condition
```

6 SELECT jako relační výraz

Uvažujme obecný SELECT výraz:

```
( SELECT DISTINCT y1 AS z1, ..., yn AS zn
  FROM   expr1 AS relation1, ..., exprm AS relationm
  WHERE  condition )
```

Kde pro $1 \leq i \leq m$ jsou

- `expri` relační výrazy,
- `relationi` jsou po dvou různé dočasné relační proměnné,
- R_i je pro typ `expri` s prefixem `relationi`,
- $R = R_1 \cup \ldots \cup R_m$,
- `condition` je podmínka nad R,
- $y_1, \ldots, y_n \in R$,
- $z_1, \ldots, z_n$ jsou po dvou různé atributy
- $D_{y_j} = D_{z_j}$ pro každé $1 \leq j \leq n$.

Hodnotu výrazu můžeme spočítat pomocí relačních operací následovně:

1. Získáme hodnoty výrazů `expr1`, ..., `exprm`, které si označíme $\mathcal{D}_1, \ldots, \mathcal{D}_m$.
2. Přejmenujeme každý atribut y relace $\mathcal{D}_i$ na `relationi.y`. Tím získáme relace $\mathcal{D}_1^R = \rho_{h_1}(\mathcal{D}_1), \ldots \mathcal{D}_m^R = \rho_{h_m}(\mathcal{D}_m)$, kde $h_1, \ldots, h_m$ označují jednotlivá přejmenování.
3. Spočítáme kartézský součin $\mathcal{D}_1^R, \ldots, \mathcal{D}_m^R$. Získáme relaci $\mathcal{D}_1^S = \mathcal{D}_1^R \times \cdots \times \mathcal{D}_m^R$.
4. Dále se provede restrikce relace $\mathcal{D}_2^S$ vzhledem k podmínce `condition`. Jako výsledek obdržíme relaci $\mathcal{D}_3^S = \sigma_\theta(\mathcal{D}_2^S)$, kde θ je zkrásený zápis podmínky `condition`.
5. Následuje projekce relace $\mathcal{D}_S^3$ na $\{y_1, \ldots, y_n\} = R$. Obdržíme relaci $\mathcal{D}_4^S = \pi_R(\mathcal{D}_3^S)$.
6. Nakonec se provede přejmenování $\mathcal{D}_4^S$ podle h, kde $h(y_1) = z_1, \ldots, h(y_n) = z_n$. Získáme výstupní relaci $\mathcal{D}_5^S = \rho_h(\mathcal{D}_4^S)$ nad S, kde $S = \{y_1, \ldots, y_n\}$.

Celý výpočet (kromě prvního kroku) můžeme zapsat výrazem:

$$\rho_h(\pi_R(\sigma_\theta(\rho_{h_1}(\mathcal{D}_1) \times \cdots \times \rho_{h_m}(\mathcal{D}_m))))$$

7 Konstantní relace

Pro atribut y a prvek $d \in \mathcal{D}_y$ je

$$[y : d] = \{\{\langle y, d\rangle\}\}$$

relací nad $\{y\}$ nazývanou *singleton*.

Jak následující věta ukazuje, lze restrikci relace $\mathcal{D}$ nad R podle podmínky tvaru $y = d$ vyjádřit pomocí restricke podle podmínky $y = y_2$, kde $y_2 \notin R$.

Věta 13. *Platí* $\sigma_{y=d}(\mathcal{D}) = \pi_R(\sigma_{y=y_2}(\mathcal{D} \times [y_2 : d]))$.

Důkaz. Pro libovolné $r \in \mathrm{Tupl}(R)$ dostáváme, že
$r \in \pi_R(\sigma_{y=y_2}(\mathcal{D} \times [y_2 : d]))$,
právě když existuje $r_2 \in \mathrm{Tupl}(R \cup \{y_2\})$ tak, že $r_2(R) = r$ a $r_2 \in \mathcal{D} \times [y_2 : d]$ a $r_2(y) = r_2(y_2)$,
právě když existuje $r_2 \in \mathrm{Tupl}(R \cup \{y_2\})$ tak, že $r_2(R) = r$ a $r_2(R) \in \mathcal{D}$ a $r_2(y_2) = d$ a $r_2(y) = r_2(y_2)$,
právě když $r \in \mathcal{D}$ a $r(y) = d$,
právě když $r \in \sigma_{y=d}(\mathcal{D})$. □

Z vět 6 a 13 vyplývá, že se lze omezit na restrikce podle podmínek tvaru $y_1 = y_2$.

Nechť $R = \{r_1, \dots, r_m\}$ je relace nad $R = \{y_1, \dots, y_n\}$

Následující výraz lze použít k určení relace $\mathcal{D}$ jménem ***relation*** ve FROM klauzuli SELECT výrazu.

```
( VALUES ( r₁(y₁), ..., r₁(yₙ) ),

         ( rₘ(y₁), ..., rₘ(yₙ) )
  AS relation ( y₁, ..., yₙ )
```

Například:

```
# SELECT *
  FROM ( VALUES ( 'Anna', 'Blue Velvet' ),
                ( 'Anna', 'Eraserhead' ),
                ( 'Bert', 'Blue Velvet' ),
                ( 'Bert', 'The Matrix' ),
                ( 'Cyril', 'Blue Velvet' ),
                ( 'Cyril', 'Eraserhead' ),
                ( 'Cyril', 'The Matrix' ) )
         AS liked ( person, movie );

 person |    movie
--------+-------------
 Anna   | Blue Velvet
 Anna   | Eraserhead
 Bert   | Blue Velvet
 Bert   | The Matrix
 Cyril  | Blue Velvet
 Cyril  | Eraserhead
 Cyril  | The Matrix
```

Uvažujme relační proměnnou `likend`

```
# TABLE liked;

 person |    movie
--------+-------------
 Anna   | Blue Velvet
 Anna   | Eraserhead
 Bert   | Blue Velvet
 Bert   | The Matrix
 Cyril  | Blue Velvet
 Cyril  | Eraserhead
 Cyril  | The Matrix
(7 rows)
```

s charakteristickou vlastností „Osoba `person` má ráda film `movie`“. Pak relace

```
# SELECT *
  FROM   liked,
       ( VALUES ( 'Blue Velvet' ) )
        AS const ( movie_blue_velvet );

 person |    movie    | movie_blue_velvet
--------+-------------+-------------------
 Anna   | Blue Velvet | Blue Velvet
 Anna   | Eraserhead  | Blue Velvet
 Bert   | Blue Velvet | Blue Velvet
 Bert   | The Matrix  | Blue Velvet
 Cyril  | Blue Velvet | Blue Velvet
 Cyril  | Eraserhead  | Blue Velvet
 Cyril  | The Matrix  | Blue Velvet
(7 rows)
```

má charakteristickou vlastnost „Osoba `person` má ráda film `movie` a `movie_blue_velvet` je film Blue Velvet.“ Následující dvě relace jsou podle věty 13 totožné.

```
# SELECT person
  FROM   liked,
       ( VALUES ( 'Blue Velvet' ) )
         AS const ( movie_blue_velvet )
  WHERE movie = movie_blue_velvet;

 person
--------
 Anna
 Bert
 Cyril
(3 rows)
```

```
# SELECT person
  FROM   liked
  WHERE  movie = 'Blue Velvet';

 person
--------
 Anna
 Bert
 Cyril
(3 rows)
```