

1. Relace

verze z 27. září 2024

1 Relace a relační proměnné

Představme si, že chceme uchovávat informace o filmech, které vlastníme. Naše právě vznikající kolekce obsahuje pouhé tři filmy: The Avengers (1998), The Avengers (2012) a The Matrix (1999). Číslo v závorce uvedené za názvem filmu je rokem jeho vydání.

Vhodné řetězce písmen anglické abecedy a podtržítka budeme nazývat **atributy**. Například `title`, `year` nebo `movie_id` jsou atributy. Množinu všech atributů označíme Y .

Každému atributu $y \in Y$ je přiřazena spočetná množina D_y nazývaná jeho **doména**. Například D_{title} je množina všech řetězců nad anglickou abecedou a D_{year} množina všech přirozených čísel.

Konečná podmnožina množiny atributů Y se nazývá (**relační**) **schéma**. Například $\{\text{title}, \text{year}\}$ je relační schéma.

Dvojici $\langle y, d \rangle$, kde y je atribut a $d \in D_y$ prvek z jeho domény nazveme **komponentou** atributu y s hodnotou d . Například $\langle \text{title}, \text{'The Matrix'} \rangle$ je komponenta atributu `title` s hodnotou `'The Matrix'`.

Mějme relační schéma S . Množinu komponent t , kde

1. pro každý atribut y z S existuje právě jedna komponenta $\langle y, d_y \rangle$ z t
2. a pro každou komponentu $\langle y, d_y \rangle$ z t platí, že $y \in S$,

nazýváme **n -tice** nad S . Například $\{\langle \text{title}, \text{'The Matrix'} \rangle, \langle \text{year}, 1999 \rangle\}$ je n -tice nad $\{\text{title}, \text{year}\}$. Všimněme si, že t je n -tice nad R , právě když $t: S \rightarrow \bigcup_{y \in S} D_y$, kde pro každé $y \in S$ je $t(y) \in D_y$. Množinu všech n -tic nad S značíme $\text{Tupl}(S)$.

Nechť S je schéma. Pak výroková formu $V(t)$, kde t je proměnná s oborem hodnot $\text{Tupl}(S)$, se nazývá **vlastnost** (**n -tic**) nad S . Například $V_1(t) = \text{„Vlastním film } t(\text{title}) \text{ vydaný v roce } t(\text{year}).\text{“}$ je vlastnost nad $\{\text{title}, \text{year}\}$ a n -tice

$$t_1 = \{\langle \text{title}, \text{'The Matrix'} \rangle, \langle \text{year}, 1999 \rangle\}$$

má vlastnost $V_1(t)$, protože výrok $V_1(t_1) = \text{„Vlastním film The Matrix vydaný v roce 1999.“}$ je pravdivý, ale n -tice

$$t_2 = \{\langle \text{title}, \text{'The Matrix'} \rangle, \langle \text{year}, 2000 \rangle\}$$

vlastnost $V_1(t)$ nemá, neboť výrok $V_1(t_2) =$ „Vlastním film The Matrix vydaný v roce 2000.“ pravdivý není.

Konečná podmnožina množiny $\text{Tupl}(S)$ se nazývá **relace** nad S . Například

$$\begin{aligned} & \{ \langle \text{title}, 'The Avengers' \rangle, \langle \text{year}, 1998 \rangle \}, \\ & \{ \langle \text{title}, 'The Avengers' \rangle, \langle \text{year}, 2012 \rangle \}, \\ & \{ \langle \text{title}, 'The Matrix' \rangle, \langle \text{year}, 1999 \rangle \} \end{aligned}$$

je relace nad $\{\text{title}, \text{year}\}$.

Relaci můžeme přehledně zadat tabulkou. Například předchozí relaci můžeme zadat tabulkou:

title	year
The Avengers	1998
The Avengers	2012
The Matrix	1999

Tabulka oproti relaci navíc určuje pořadí n -tic a atributů. Tedy i tato tabulka určuje stejnou relaci:

year	title
1999	The Matrix
1998	The Avengers
2012	The Avengers

Prázdnou relací nad R rozumíme prázdnou množinu \emptyset .

Nechť R je relace nad S a $V(t)$ je vlastnost nad R . Pak $V(t)$ se nazývá **charakteristická vlastnost** R (zkráceně jen **vlastnost určující** R), jestliže platí $t \in R$, právě když $V(t)$ je pravdivé. Platí, že $V(t)$ určuje R , právě když $R = \{t \in \text{Tupl}(S) \mid V(t)\}$.

Například vlastnost „Vlastním film $t(\text{title})$ vytvořený roku $t(\text{year})$.“ je charakteristickou vlastností relace:

title	year
The Avengers	1998
The Avengers	2012
The Matrix	1999

Charakteristické vlastnosti relací budeme zjednodušovat vynecháním proměnné t . Tedy předchozí vlastnost lze zkrátit na: „Vlastním film **title** vytvořený roku **year**.“

Databáze vždy zachycuje jen omezenou část reálného světa. Například udržujeme informace jen o vlastněných filmech. Pokud jasně určíme část světa, která nás zajímá, můžeme charakteristické vlastnosti relací zjednodušit.

Například místo:

„Vlastním film `title` vydaný roku `year`.“

uvedeme jen:

„Film `title` byl vydaný roku `year`.“

K označování objektů reálného světa používáme **identifikátory** (zkráceně **id**). Přičemž identifikátor musí objekt jednoznačně určovat. Jako identifikátory objektů se často používají přirozená čísla. Atributy určující identifikátory končí na `id`. Například identifikátor filmu může být určený atributem `movie_id`. Objekt reálného světa ztotožňujeme s jeho identifikátorem.

Například vlastnost „Film `movie_id` má název `title` a byl vydaný roku `year`.“ určuje relaci:

<code>movie_id</code>	<code>title</code>	<code>year</code>
1	The Avengers	1998
2	The Avengers	2012
3	The Matrix	1999

Podívejme se na chvíli na jinou část reálného světa, kterou chceme modelovat. Konkrétně chceme databázi zachytit herce, kteří nás zajímají a vystupující v námi vlastněných filmech. Předpokládejme, že se zajímáme o herce Keanu Reeves, Laurence Fishburne, Gary Oldman a Anthony Hopkins a že vlastníme filmy The Matrix (1999), Dracula (1992), The Elephant Man (1980) a Amarcord (1973). Vymezenou část reálného světa zachytíme relací určenou vlastností „Zajímá mě herec `actor_name`, který hrál ve mnou vlastněném filmu `movie_title` z roku `movie_year`.“ Díky vymezení části modelovaného světa, můžeme vlastnost zkrátit na „Herec `actor_name` hrál ve filmu `movie_title` z roku `movie_year`.“ Určenou relaci zobrazuje tabulka:

<code>actor_name</code>	<code>movie_title</code>	<code>movie_year</code>
Keanu Reeves	The Matrix	1999
Keanu Reeves	Dracula	1992
Laurence Fishburne	The Matrix	1999
Gary Oldman	Dracula	1992
Anthony Hopkins	The Elephant Man	1980

Všimněme si, že v relaci není uvedený film Amarcord a to z toho důvodu, že neexistuje herec, který by nás zajímal a současně by ve filmu hrál.

Relační proměnná je proměnná, jejíž hodnota je relace. **Typem** relační proměnné je relační schéma. Hodnotou relační proměnné musí být relace nad jejím schématem. K relační proměnné přiřazujeme její **charakteristickou vlastnost**, kterou je výroková forma. Charakteristická vlastnost proměnné je vždy charakteristickou vlastností její hodnoty. **Základní relace** je relační proměnná, jejíž hodnotu lze přímo měnit.

Například základní relace `movie` nad `{title, year}` určená vlastností: „Film `title` je z roku `year`.“ má hodnotu:

title	year
The Avengers	1998
The Avengers	2012
The Matrix	1999

Pokud bychom prodali film The Matrix, musíme změnit hodnotu proměnné na:

title	year
The Avengers	1998
The Avengers	2012

aby charakteristická vlastnost proměnné byla stále charakteristickou vlastností její hodnoty.

2 Relace v SQL

SQL tabulky jsou podobné relacím, ale liší se od nich tím, že určují pořadí sloupců i řádků, názvy sloupců nemusí být jedinečné, mohou obsahovat duplicitní řádky a buňka tabulky nemusí obsahovat hodnotu.

Jako relaci můžeme chápat SQL tabulku, kde

- názvy sloupců jsou jedinečné,
- nejsou duplicitní řádky,
- se nespolehneme na pořadí řádků,
- všechny buňky mají hodnotu.

Jiné tabulky používat nebudeme.

Skalární typ dává jméno množině, která může být doménou atributu. Například `integer` je skalárním typem, který dává jméno množině celých čísel od $-2\,147\,483\,648$ do $2\,147\,483\,647$. Skalární typ `text` pojmenovává množinu všech řetězců. Skalární typ `text` není ve standardu SQL, ale zavádí jej databázový systém PostgreSQL, s kterým budeme pracovat.

Atribut (sloupec) SQL tabulky má doménu určenou skalárním typem.

Přijmeme následující omezení. Stejné atributy v různých SQL tabulkách musí být stejného skalárního typu.

Skalární typ atributů můžeme zobrazit v tabulce:

title text	year integer
The Avengers	1998
The Avengers	2012
The Matrix	1999

Budeme používat databázový systém PostgreSQL aspoň ve verzi 16, který si můžete stáhnout z adresy <https://www.postgresql.org/download/>. Pro přístup k systému můžete použít konzolový nástroj `psql` (SQL shell), který je součástí instalace systému nebo editor Visual Studio Code (<https://code.visualstudio.com>) s rozšířením Database Client (id rozšíření je `cweijan.vscode-database-client2`). Skripty pro systém mají příponu `sql`.

Příkaz

```
CREATE TABLE relation (  
    attribute1 scalar_type1,  
    attribute2 scalar_type2,  
    :  
);
```

deklaruje základní relaci *relation* nad $\{attribute1, attribute2, \dots\}$. Doména atributu *attribute_i* je dána skalárním typem *scalar_type_i*. Hodnotou je prázdná relace.

Příklad:

```
CREATE TABLE movie (  
    title text,  
    year integer,  
);
```

Příkaz

```
TABLE relation;
```

vrátí hodnotu relační proměnné *relation*. Příklad:

```
# TABLE movie;  
title | year  
-----+-----  
(0 rows)
```

Následující příkaz slouží k přidání n -tic do relační proměnné. Necht *relation* je relační proměnná nad $S = \{y_1, \dots, y_n\}$ a hodnotou R a t_1, \dots, t_m jsou n -tice nad S takové, že množiny R a $\{t_1, \dots, t_m\}$ jsou disjunktní. Předpokládáme pořadí atributů y_1, \dots, y_n dané při deklaraci proměnné.

Příkaz

```
INSERT INTO relation VALUES
  ( $t_1(y_1), \dots, t_1(y_n)$ ),
  :
  ( $t_m(y_1), \dots, t_m(y_n)$ );
```

nastaví hodnotu proměnné *relation* na $R \cup \{t_1, \dots, t_m\}$.

Například po vykonání

```
# INSERT INTO movie VALUES
  ( 'The Matrix', 1999 ),
  ( 'The Avengers', 2012 ),
  ( 'The Avengers', 1998 );
```

se změní hodnota proměnné na

```
# TABLE movie;
  title      | year
-----+-----
The Matrix   | 1999
The Avengers | 2012
The Avengers | 1998
(3 rows)
```

Pokud si pořídíme film A Space Odyssey z roku 1968, musíme změnit hodnotu proměnné příkazem:

```
INSERT INTO movie VALUES ( 'A Space Odyssey', 1968 );
```

Nová hodnota proměnné:

```
# TABLE movie;
  title      | year
-----+-----
The Matrix   | 1999
The Avengers | 2012
The Avengers | 1998
A Space Odyssey | 1968
(4 rows)
```

Příkaz

```
DROP TABLE relation;
```

zruší základní relaci *relation*.

Příklad:

```
DROP TABLE movie;
```

Otázky a úkoly na cvičení

1. Předpokládejme v následujících úkolech, že doména atributů *year* a *length* je množina přirozených čísel a *title* množina řetězců nad anglickou abecedou (s mezerou). Rozhodněte, zda jsou následující dvojice komponentami. Své rozhodnutí zdůvodněte.
 - (a) $\langle \text{title}, \text{'apple'} \rangle$
 - (b) $\langle \text{'The Last Laugh'}, \text{title} \rangle$
 - (c) $\langle \text{year}, \text{'The Last Laugh'} \rangle$
2. Rozhodněte, které z následujících množin jsou n -tice nad $\{\text{title}, \text{year}\}$.
 - (a) $\{\langle \text{title}, \text{'The Last Laugh'} \rangle\}$
 - (b) $\{\langle \text{year}, 1924 \rangle, \langle \text{title}, \text{'The Last Laugh'} \rangle\}$
 - (c) $\{\langle \text{year}, 1924 \rangle, \langle \text{year}, 1924 \rangle, \langle \text{title}, \text{'The Last Laugh'} \rangle\}$
 - (d) $\{\langle \text{year}, 1924 \rangle, \langle \text{year}, 1923 \rangle, \langle \text{title}, \text{'The Last Laugh'} \rangle\}$
 - (e) $\{\langle \text{'The Last Laugh'}, \text{title} \rangle, \langle 1924, \text{year} \rangle\}$
 - (f) $\{\langle \text{year}, 1924 \rangle, \langle \text{title}, \text{'The Last Laugh'} \rangle, \langle \text{length}, 90 \rangle\}$
3. Je následující množina relace nad $\{\text{title}, \text{year}\}$? Kolik má prvků?
$$\{\{\langle \text{title}, \text{'The Last Laugh'} \rangle, \langle \text{year}, 1924 \rangle\}, \{\langle \text{year}, 1924 \rangle, \langle \text{title}, \text{'The Last Laugh'} \rangle\}\}$$
4. Zakreslete následující relaci nad $\{\text{title}, \text{year}\}$ tabulkou.
$$\{\{\langle \text{title}, \text{'The Last Laugh'} \rangle, \langle \text{year}, 1924 \rangle\}, \{\langle \text{title}, \text{'The Burning Soil'} \rangle, \langle \text{year}, 1922 \rangle\}\}$$
5. Zapište relaci zakreslenou následující tabulkou.

title	length
Faust	106
The Last Laugh	90

6. Vytvořte v SQL relační proměnnou danou vlastností „Režisér `director` natočil film `movie` dlouhý `length` minut.“ Zaujímáme se jen o filmy, které vlastníme. Vlastníme dva filmy od F. W. Murnaua. Jmenovitě film Faust dlouhý 106 minut a film City Girl dlouhý 89 minut. Dále vlastníme 109 minut dlouhý film Psycho od Alfreda Hitchcocka.