Bazy danych 2024

27 lutego 2024

Algebra relacji

- Operatory: $\pi, \sigma, \rho, \times, \cup, \setminus, \bowtie,$

Algebra relacji

- Operatory: $\pi, \sigma, \rho, \times, \cup, \setminus, \bowtie$,
- operacyjny język (mówimy w jaki sposób obliczyć wynik zapytania),

Algebra relacji

- Operatory: $\pi, \sigma, \rho, \times, \cup, \setminus, \bowtie$,
- operacyjny język (mówimy w jaki sposób obliczyć wynik zapytania),
- domyślnie brak grupowania i agregacji (γ) - kalkulator

Algebra relacji

- Operatory: $\pi, \sigma, \rho, \times, \cup, \setminus, \bowtie$,
- operacyjny język (mówimy w jaki sposób obliczyć wynik zapytania),
- domyślnie brak grupowania i agregacji (γ) - kalkulator
- ale i tak dajemy radę:

Algebra relacji

- Operatory: $\pi, \sigma, \rho, \times, \cup, \setminus, \bowtie$,
- operacyjny język (mówimy w jaki sposób obliczyć wynik zapytania),
- domyślnie brak grupowania i agregacji (γ) - kalkulator
- ale i tak dajemy radę:
- ```
  movies -
  pi movies.id, movies.name, movies.year, movies.rank
  (sigma movies.rank < movies2.rank (movies × rho movies2 (movies)))
  ```
  ```

Algebra relacji

- Operatory: $\pi, \sigma, \rho, \times, \cup, \setminus, \bowtie$,
- operacyjny język (mówimy w jaki sposób obliczyć wynik zapytania),
- domyślnie brak grupowania i agregacji (γ) - kalkulator
- ale i tak dajemy radę:
-
```
movies -
pi movies.id, movies.name, movies.year, movies.rank
(sigma movies.rank < movies2.rank (movies × rho movies2 (movies)))
```

movies.id	movies.name	movies.year	movies.rank
92616	'Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb'	1964	8.7
267038	'Pulp Fiction'	1994	8.7

-

Algebra relacji

```
EXPLAIN
  SELECT m1.* FROM movies m1
              JOIN movies m2 ON (m1.rank > m2.rank)
              JOIN movies_genre mg ON (m1.id = mg.movies_id )
              WHERE m1.year>1960;
```

Algebra relacji

```
EXPLAIN
  SELECT m1.* FROM movies m1
              JOIN movies m2 ON (m1.rank > m2.rank)
              JOIN movies_genre mg ON (m1.id = mg.movies_id )
              WHERE m1.year>1960;
              QUERY PLAN
---------------------------------------------------------------------------------------
Merge Join
  Merge Cond: (mg.movies_id = m1.id)
  ->  Sort (Sort Key: mg.movies_id)
        ->  Seq Scan on movies_genre mg
        ->  Sort (Sort Key: m1.id)
              ->  Nested Loop
                    Join Filter: (m1.rank > m2.rank)
                    ->  Seq Scan on movies m2
                          ->  Seq Scan on movies m1
                                Filter: (year > 1960)
```

Rachunki relacyjne

- logika I rzędu (first-order logic)

Rachunki relacyjne

- logika I rzędu (first-order logic) przykłady

Rachunki relacyjne

- logika I rzędu (first-order logic) przykłady
- język deklaratywny (co chcę mieć w wyniku, a nie jak to policzyć)

Rachunki relacyjne

- logika I rzędu (first-order logic) przykłady
- język deklaratywny (co chcę mieć w wyniku, a nie jak to policzyć)
- dobry benchmark, dużo wiadomo np. o zawieraniu zapytań, itp.

Rachunki relacyjne

- logika I rzędu (first-order logic) przykłady
- język deklaratywny (co chcę mieć w wyniku, a nie jak to policzyć)
- dobry benchmark, dużo wiadomo np. o zawieraniu zapytań, itp.
- łatwy?

Rachunki relacyjne

- logika I rzędu (first-order logic) przykłady
- język deklaratywny (co chcę mieć w wyniku, a nie jak to policzyć)
- dobry benchmark, dużo wiadomo np. o zawieraniu zapytań, itp.
- łatwy?
- dwa warianty - relacyjny rachunek dziedzin i krotek

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

- $x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')$

Znaczenie zapytań

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

- $x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')$

Znaczenie zapytań

- x - student, który dostał *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

- $x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')$

Znaczenie zapytań

- *x* - student, który dostał *5.0* z *BD*.
- Wszyscy studenci, którzy dostali *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

- $x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')$
- $\{x \mid x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')\}$

Znaczenie zapytań

- *x* - student, który dostał *5.0* z *BD*.
- Wszyscy studenci, którzy dostali *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

- $x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')$
- $\{x \mid x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')\}$

Znaczenie zapytań

- *x* - student, który dostał *5.0* z *BD*.
- Wszyscy studenci, którzy dostali *5.0* z *BD*.
- Nazwiska i indeksy studentów, którzy dostali *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok), \; P = (\underline{nazwa}, typ), \; O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

- $x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')$
- $\{x \mid x \in S \wedge (\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD')\}$
- $\{z^{[nazwisko,indeks]} \mid$
  $(\exists x)(x \in S \wedge z.indeks = x.indeks \wedge z.nazwisko = x.nazwisko \wedge$
  $(\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5 \wedge y.przed =' BD'))\}$

Znaczenie zapytań

- x - student, który dostał *5.0* z *BD*.
- Wszyscy studenci, którzy dostali *5.0* z *BD*.
- Nazwiska i indeksy studentów, którzy dostali *5.0* z *BD*.

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

Znaczenie zapytań

2a. Studenci, którzy nie dostali żadnej piątki.

2b. Studenci, którzy mają tylko piątki.

2c. Studenci, którzy mają najlepszą ocenę z Baz danych.

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

2a. $\{x \mid x \in S \land \neg(\exists y)(y \in O \land y.indeks = x.indeks \land y.stop = 5)\}$

Znaczenie zapytań

2a. Studenci, którzy nie dostali żadnej piątki.

2b. Studenci, którzy mają tylko piątki.

2c. Studenci, którzy mają najlepszą ocenę z Baz danych.

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

2a. $\{x \mid x \in S \wedge \neg(\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop = 5)\}$

2b. $\{x \mid x \in S \wedge \neg(\exists y)(y \in O \wedge y.indeks = x.indeks \wedge y.stop \neq 5)\}$

Znaczenie zapytań

2a. Studenci, którzy nie dostali żadnej piątki.

2b. Studenci, którzy mają tylko piątki.

2c. Studenci, którzy mają najlepszą ocenę z Baz danych.

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ \ P = (\underline{nazwa}, typ),\ \ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

2a. $\{x \mid x \in S \land \neg(\exists y)(y \in O \land y.indeks = x.indeks \land y.stop = 5)\}$

2b. $\{x \mid x \in S \land \neg(\exists y)(y \in O \land y.indeks = x.indeks \land y.stop \neq 5)\}$

2c. $\{x \mid x \in S \land (\exists y)(y \in O \land y.indeks = x.indeks \land y.przed =' BD' \land$
$\neg(\exists y_1)(y_1 \in O \land y_1.indeks \neq x.indeks \land y_1.przed =' BD' \land y_1.stop > y.stop)\}$

Znaczenie zapytań

2a. Studenci, którzy nie dostali żadnej piątki.

2b. Studenci, którzy mają tylko piątki.

2c. Studenci, którzy mają najlepszą ocenę z Baz danych.

Zapytanie relacyjnego rachunku krotek

Formuła rrk — opisuje własności krotek

Formuła atomowa:

- $R(t)$ lub $t \in R$, gdzie R to relacja z bazy danych, a t to zmienna (krotkowa);
- $t.a = c$, gdzie a jest atrybutem t; równość można zastąpić przez: $\neq, <, \leq, >, \geq$, a c jest stałą lub atrybutem innej zmiennej krotkowej;

Formuła:

- formuła atomowa,
- (ϕ), $\neg(\phi)$, $\phi \vee \psi$, $\phi \wedge \psi$, gdzie ϕ i ψ są formułami;
- $(\exists t)(\phi(t))$ lub $(\forall t)(\phi(t))$, gdzie ϕ jest formułą, a t jej zmienną wolną.

Zapytanie rrk — wybiera krotki mające daną własność

$$\{x \mid \phi(x)\} \qquad \{x^{[A_1, A_2, \ldots, A_k]} \mid \phi(x)\},$$

gdzie x jest zmienną krotkową, a ϕ jest formułą relacyjnego rachunku krotek, w której x jest jedyną zmienną wolną;

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

Znaczenie zapytań

1. x - student, który dostał *5.0* z *BD*.
2. Wszyscy studenci, którzy dostali *5.0* z *BD*.
3. Nazwiska i indeksy studentów, którzy dostali *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

1. $S(x_1, x_2, x_3) \wedge (\exists y_1, y_2, y_3, y_4)((O(y_1, y_2, y_3, y_4) \wedge y_1 = x_1 \wedge y_2 =' BD' \wedge y_4 = 5)$

Znaczenie zapytań

1. x - student, który dostał *5.0* z *BD*.
2. Wszyscy studenci, którzy dostali *5.0* z *BD*.
3. Nazwiska i indeksy studentów, którzy dostali *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

1. $S(x_1, x_2, x_3) \wedge (\exists y_1, y_2, y_3, y_4)((O(y_1, y_2, y_3, y_4) \wedge y_1 = x_1 \wedge y_2 =' BD' \wedge y_4 = 5)$

1a. $S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5))$

Znaczenie zapytań

1. *x* - student, który dostał *5.0* z *BD*.
2. Wszyscy studenci, którzy dostali *5.0* z *BD*.
3. Nazwiska i indeksy studentów, którzy dostali *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

1. $S(x_1, x_2, x_3) \wedge (\exists y_1, y_2, y_3, y_4)((O(y_1, y_2, y_3, y_4) \wedge y_1 = x_1 \wedge y_2 =' BD' \wedge y_4 = 5)$

1a. $S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5))$

2. $\{x_1, x_2, x_3 \mid S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5))\}$

Znaczenie zapytań

1. *x* - student, który dostał *5.0* z *BD*.
2. Wszyscy studenci, którzy dostali *5.0* z *BD*.
3. Nazwiska i indeksy studentów, którzy dostali *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

1. $S(x_1, x_2, x_3) \wedge (\exists y_1, y_2, y_3, y_4)((O(y_1, y_2, y_3, y_4) \wedge y_1 = x_1 \wedge y_2 =' BD' \wedge y_4 = 5)$

1a. $S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5))$

2. $\{x_1, x_2, x_3 \mid S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5))\}$
3. $\{x_1, x_2 \mid (\exists x_3)(S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5)))\}$

Znaczenie zapytań

1. *x* - student, który dostał *5.0* z *BD*.
2. Wszyscy studenci, którzy dostali *5.0* z *BD*.
3. Nazwiska i indeksy studentów, którzy dostali *5.0* z *BD*.

Przykład 1

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

1. $S(x_1, x_2, x_3) \wedge (\exists y_1, y_2, y_3, y_4)((O(y_1, y_2, y_3, y_4) \wedge y_1 = x_1 \wedge y_2 =' BD' \wedge y_4 = 5)$

1a. $S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5))$

2. $\{x_1, x_2, x_3 \mid S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5))\}$

3. $\{x_1, x_2 \mid (\exists x_3)(S(x_1, x_2, x_3) \wedge (\exists y_3)(O(x_1,' BD', y_3, 5)))\}$

3a. $\{ind, naz \mid (\exists rok)(S(ind, naz, rok) \wedge (\exists dat)(O(ind,' BD', dat, 5)))\}$

Znaczenie zapytań

1. x - student, który dostał *5.0* z *BD*.
2. Wszyscy studenci, którzy dostali *5.0* z *BD*.
3. Nazwiska i indeksy studentów, którzy dostali *5.0* z *BD*.

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok)$, $P = (\underline{nazwa}, typ)$, $O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

Znaczenie zapytań

2a. Studenci, którzy nie dostali żadnej piątki.

2b. Studenci, którzy mają tylko piątki.

2c. Studenci, którzy mają najlepszą ocenę z Baz danych.

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

2a. $\{(i, n, r) \mid S(i, n, r) \wedge \neg(\exists p, d)(O(i, p, d, 5))\}$

Znaczenie zapytań

2a. Studenci, którzy nie dostali żadnej piątki.

2b. Studenci, którzy mają tylko piątki.

2c. Studenci, którzy mają najlepszą ocenę z Baz danych.

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

2a. $\{(i, n, r) \mid S(i, n, r) \wedge \neg(\exists p, d)(O(i, p, d, 5))\}$

2b. $\{(i, n, r) \mid S(i, n, r) \wedge \neg(\exists p, d, s)(O(i, p, d, s) \wedge s \neq 5)\}$

Znaczenie zapytań

2a. Studenci, którzy nie dostali żadnej piątki.

2b. Studenci, którzy mają tylko piątki.

2c. Studenci, którzy mają najlepszą ocenę z Baz danych.

Przykład 2

Baza danych

$S = (\underline{indeks}, nazwisko, rok),\ P = (\underline{nazwa}, typ),\ O = (\underline{indeks}, \underline{przed}, \underline{data}, stop)$

2a. $\{(i,n,r) \mid S(i,n,r) \land \neg(\exists p,d)(O(i,p,d,5))\}$

2b. $\{(i,n,r) \mid S(i,n,r) \land \neg(\exists p,d,s)(O(i,p,d,s) \land s \neq 5)\}$

2c. $\{(i,n,r) \mid S(i,n,r) \land (\exists d,s)(O(i,'BD',d,s) \land \neg(\exists i_1,d_1,s_1)(O(i_1,'BD',d_1,s_1) \land i \neq i_1 \land s_1 > s))\}$

Znaczenie zapytań

2a. Studenci, którzy nie dostali żadnej piątki.

2b. Studenci, którzy mają tylko piątki.

2c. Studenci, którzy mają najlepszą ocenę z Baz danych.

Zapytanie relacyjnego rachunku dziedzin

Formuła rrd — opisuje własności wektorów zmiennych dziedzinowych

Formuła atomowa:

- $R(x_1, x_2, \ldots, x_k)$ lub $(x_1, x_2, \ldots, x_k) \in R$, gdzie R to relacja o arności k, $x_1, x_2, \ldots, x_k$ to zmienne lub stałe (dziedzinowe);
- $x = c$, gdzie x jest zmienną; równość można zastąpić przez: $\neq, <, \leq, >, \geq$, a c jest stałą lub zmienną;

Formuła:

- formuła atomowa,
- (ϕ), $\neg(\phi)$, $\phi \vee \psi$, $\phi \wedge \psi$, gdzie ϕ i ψ są formułami;
- $(\exists t_1, t_2, \ldots, t_\ell)(\phi(t_1, t_2, \ldots, t_\ell)$ lub $(\forall t_1, t_2, \ldots, t_\ell)(\phi(t_1, t_2, \ldots, t_\ell)$, gdzie ϕ jest formułą, a $t_1, t_2, \ldots, t_\ell$ jej zmiennymi wolnymi.

Zapytanie rrd — wybiera wektory wartości zmiennych mające daną własność

$$\{x_1, x_2, \ldots, x_\ell \mid \phi(y_1, y_2, \ldots, y_k)\},$$

gdzie ϕ jest formułą relacyjnego rachunku krotek, a $x_1, x_2, \ldots, x_\ell$ to zmienne dziedzinowe i stałe, wśród których występują wszystkie zmienne wolne ϕ: $y_1, y_2, \ldots, y_k$ i tylko takie zmienne;

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x|\neg movies(x)\}$?

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x | \neg movies(x)\}$?
- A na $\{x, y | movies(x, 'SomeTitle', 2022, 5) \vee movies(y, 'OtherTitle', 2023, 5)\}$?

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x | \neg \mathit{movies}(x)\}$?
- A na $\{x, y | \mathit{movies}(x, '\mathrm{SomeTitle}', 2022, 5) \vee \mathit{movies}(y, '\mathrm{OtherTitle}', 2023, 5)\}$?
- Co jest nie tak?

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x | \neg movies(x)\}$?
- A na $\{x, y | movies(x, '\mathrm{SomeTitle}', 2022, 5) \vee movies(y, '\mathrm{OtherTitle}', 2023, 5)\}$?
- Co jest nie tak?
- W obu przypadkach wynik zależy od przyjętej dziedziny (zbioru dopuszczalnych wartości w kolumnie!)

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x|\neg movies(x)\}$?
- A na $\{x, y|movies(x, '\mathrm{SomeTitle}', 2022, 5) \vee movies(y, '\mathrm{OtherTitle}', 2023, 5)\}$?
- Co jest nie tak?
- W obu przypadkach wynik zależy od przyjętej dziedziny (zbioru dopuszczalnych wartości w kolumnie!)
- AAAAAAAAAA! Przecież to może być zbiór nieskończony!

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x|\neg movies(x)\}$?
- A na $\{x, y|movies(x, '\mathrm{SomeTitle}', 2022, 5) \vee movies(y, '\mathrm{OtherTitle}', 2023, 5)\}$?
- Co jest nie tak?
- W obu przypadkach wynik zależy od przyjętej dziedziny (zbioru dopuszczalnych wartości w kolumnie!)
- AAAAAAAAAA! Przecież to może być zbiór nieskończony!
- Nawet gdyby dziedzina była skończona to i tak kto pisząc zapytanie pamięta ją całą?

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x | \neg movies(x)\}$?
- A na $\{x, y | movies(x, '\mathrm{SomeTitle}', 2022, 5) \vee movies(y, '\mathrm{OtherTitle}', 2023, 5)\}$?
- Co jest nie tak?
- W obu przypadkach wynik zależy od przyjętej dziedziny (zbioru dopuszczalnych wartości w kolumnie!)
- AAAAAAAAAA! Przecież to może być zbiór nieskończony!
- Nawet gdyby dziedzina była skończona to i tak kto pisząc zapytanie pamięta ją całą?
- Formuły, które mogą zwracać nieskończony wynik nie są bezpieczne!

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x | \neg movies(x)\}$?
- A na $\{x, y | movies(x, '\mathrm{SomeTitle}', 2022, 5) \vee movies(y, '\mathrm{OtherTitle}', 2023, 5)\}$?
- Co jest nie tak?
- W obu przypadkach wynik zależy od przyjętej dziedziny (zbioru dopuszczalnych wartości w kolumnie!)
- AAAAAAAAAA! Przecież to może być zbiór nieskończony!
- Nawet gdyby dziedzina była skończona to i tak kto pisząc zapytanie pamięta ją całą?
- Formuły, które mogą zwracać nieskończony wynik nie są bezpieczne!
- Matematycy radzą sobie pamiętając osobno dziedzinę - model czyli matematyczna baza danych to dziedzina, sygnatura (=schemat) i interpretacja ($\approx$ stan bazy)

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x | \neg movies(x)\}$?
- A na $\{x, y | movies(x, '\mathrm{SomeTitle}', 2022, 5) \lor movies(y, '\mathrm{OtherTitle}', 2023, 5)\}$?
- Co jest nie tak?
- W obu przypadkach wynik zależy od przyjętej dziedziny (zbioru dopuszczalnych wartości w kolumnie!)
- AAAAAAAAAA! Przecież to może być zbiór nieskończony!
- Nawet gdyby dziedzina była skończona to i tak kto pisząc zapytanie pamięta ją całą?
- Formuły, które mogą zwracać nieskończony wynik nie są bezpieczne!
- Matematycy radzą sobie pamiętając osobno dziedzinę - model czyli matematyczna baza danych to dziedzina, sygnatura (=schemat) i interpretacja ($\approx$ stan bazy)
- I niczego się tu nie boją! Dlaczego?

Mały problem

- Jaka powinna być odpowiedź na zapytanie: $\{x|\neg movies(x)\}$?
- A na $\{x, y|movies(x, '\mathrm{SomeTitle}', 2022, 5) \vee movies(y, '\mathrm{OtherTitle}', 2023, 5)\}$?
- Co jest nie tak?
- W obu przypadkach wynik zależy od przyjętej dziedziny (zbioru dopuszczalnych wartości w kolumnie!)
- AAAAAAAAAAA! Przecież to może być zbiór nieskończony!
- Nawet gdyby dziedzina była skończona to i tak kto pisząc zapytanie pamięta ją całą?
- Formuły, które mogą zwracać nieskończony wynik nie są bezpieczne!
- Matematycy radzą sobie pamiętając osobno dziedzinę - model czyli matematyczna baza danych to dziedzina, sygnatura (=schemat) i interpretacja ($\approx$ stan bazy)
- I niczego się tu nie boją! Dlaczego?
- Inne przykłady: $\{x|R(y)\}, \{x|R(x) \vee R(y)\}, \{x|R(x) \vee \neg R(x)\}$

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

$\{a, b \mid (\forall d)((\exists c)(E(c, d)) \Rightarrow E(a, d) \vee E(b, d))\}$

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

$\{a, b \mid (\forall d)((\exists c)(E(c, d)) \Rightarrow E(a, d) \vee E(b, d))\}$

Co jest nie tak?

- Szukamy takich par (a, b), że dla każdego tematu d (czyli zmiennej występującej w drugiej kolumnie E) a lub b jest ekspertem w tym temacie (występuje w parze z d w relacji E).

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

$$\{a, b \mid (\forall d)((\exists c)(E(c, d)) \Rightarrow E(a, d) \vee E(b, d))\}$$

Co jest nie tak?

- Szukamy takich par (a, b), że dla każdego tematu d (czyli zmiennej występującej w drugiej kolumnie E) a lub b jest ekspertem w tym temacie (występuje w parze z d w relacji E).
- Jeśli nie ma tematów (E jest pusta), to każda para wartości $(?, ?)$ spełnia formułę zapytania.

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

$$\{a, b \mid (\forall d)((\exists c)(E(c, d)) \Rightarrow E(a, d) \vee E(b, d))\}$$

Co jest nie tak?

- Szukamy takich par (a, b), że dla każdego tematu d (czyli zmiennej występującej w drugiej kolumnie E) a lub b jest ekspertem w tym temacie (występuje w parze z d w relacji E).
- Jeśli nie ma tematów (E jest pusta), to każda para wartości $(?, ?)$ spełnia formułę zapytania.
- Jeśli jest jeden ekspert *Wszystkowiedzący* znający się na wszystkim, to każda para $(\text{'Wszystkowiedzący'}, ?)$ spełnia formułę zapytania.

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

$\{a, b \mid (\forall d)((\exists c)(E(c, d)) \Rightarrow E(a, d) \vee E(b, d))\}$

$\{x \mid P(x) \wedge (\forall y) S(y)\}$

Co jest nie tak?

- Szukamy takich par (a, b), że dla każdego tematu d (czyli zmiennej występującej w drugiej kolumnie E) a lub b jest ekspertem w tym temacie (występuje w parze z d w relacji E).
- Jeśli nie ma tematów (E jest pusta), to każda para wartości $(?, ?)$ spełnia formułę zapytania.
- Jeśli jest jeden ekspert *Wszystkowiedzący* znający się na wszystkim, to każda para $(\text{'Wszystkowiedzący'}, ?)$ spełnia formułę zapytania.

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

$\{a, b \mid (\forall d)((\exists c)(E(c, d)) \Rightarrow E(a, d) \vee E(b, d))\}$

$\{x \mid P(x) \wedge (\forall y)S(y)\}$ Skończony! Mamy rozwiązanie?

Co jest nie tak?

- Szukamy takich par (a, b), że dla każdego tematu d (czyli zmiennej występującej w drugiej kolumnie E) a lub b jest ekspertem w tym temacie (występuje w parze z d w relacji E).
- Jeśli nie ma tematów (E jest pusta), to każda para wartości $(?, ?)$ spełnia formułę zapytania.
- Jeśli jest jeden ekspert *Wszystkowiedzący* znający się na wszystkim, to każda para $(\text{'Wszystkowiedzący'}, ?)$ spełnia formułę zapytania.

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

$\{a, b \mid (\forall d)((\exists c)(E(c, d)) \Rightarrow E(a, d) \vee E(b, d))\}$

$\{x \mid P(x) \wedge (\forall y)S(y)\}$ Skończony! Mamy rozwiązanie?

Co jest nie tak?

- Szukamy takich par (a, b), że dla każdego tematu d (czyli zmiennej występującej w drugiej kolumnie E) a lub b jest ekspertem w tym temacie (występuje w parze z d w relacji E).
- Jeśli nie ma tematów (E jest pusta), to każda para wartości $(?, ?)$ spełnia formułę zapytania.
- Jeśli jest jeden ekspert *Wszystkowiedzący* znający się na wszystkim, to każda para $(\text{'Wszystkowiedzący'}, ?)$ spełnia formułę zapytania.
- Zapytania mogą zależeć od dziedziny nawet jak są skończone!

Mały problem

Bardziej skomplikowany przykład

$E(osoba, temat)$ to baza ekspertów z określonych tematów. Poszukujemy pary ekspertów, którzy razem tworzą zespół znający się na wszystkich tematach.

$\{a, b \mid (\forall d)((\exists c)(E(c, d)) \Rightarrow E(a, d) \vee E(b, d))\}$

$\{x \mid P(x) \wedge (\forall y)S(y)\}$ Skończony! Mamy rozwiązanie?

$\{x \mid P(x) \wedge (\forall y)L(x, y)\}$?

Co jest nie tak?

- Szukamy takich par (a, b), że dla każdego tematu d (czyli zmiennej występującej w drugiej kolumnie E) a lub b jest ekspertem w tym temacie (występuje w parze z d w relacji E).
- Jeśli nie ma tematów (E jest pusta), to każda para wartości $(?, ?)$ spełnia formułę zapytania.
- Jeśli jest jeden ekspert *Wszystkowiedzący* znający się na wszystkim, to każda para $(\text{'Wszystkowiedzący'}, ?)$ spełnia formułę zapytania.
- Zapytania mogą zależeć od dziedziny nawet jak są skończone!

Niezależność od dziedziny (domain independence)

Niech $\phi_B(\mathcal{A})$ oznacza zbiór krotek będący wynikiem zapytania ϕ obliczonego na bazie $\mathcal{A}$ przy założeniu, że dziedzina (zbiór możliwych wartości zmiennych) to B.

Niezależność od dziedziny (domain independence)

Niech $\phi_B(\mathcal{A})$ oznacza zbiór krotek będący wynikiem zapytania ϕ obliczonego na bazie $\mathcal{A}$ przy założeniu, że dziedzina (zbiór możliwych wartości zmiennych) to B.

Dziedzina aktywna formuły

Zbiór D nazwiemy **dziedziną aktywną** formuły ϕ i bazy $\mathcal{A}$, gdy jest to zbiór wszystkich wartości występujących we wszystkich kolumnach wszystkich relacji z $\mathcal{A}$ oraz wszystkich stałych występujących jawnie w ϕ.

Niezależność od dziedziny (domain independence)

Niech $\phi_B(\mathcal{A})$ oznacza zbiór krotek będący wynikiem zapytania ϕ obliczonego na bazie $\mathcal{A}$ przy założeniu, że dziedzina (zbiór możliwych wartości zmiennych) to B.

Dziedzina aktywna formuły

Zbiór D nazwiemy **dziedziną aktywną** formuły ϕ i bazy $\mathcal{A}$, gdy jest to zbiór wszystkich wartości występujących we wszystkich kolumnach wszystkich relacji z $\mathcal{A}$ oraz wszystkich stałych występujących jawnie w ϕ.

Zapytanie ϕ jest *dziedzinowo niezależne* wtedy i tylko wtedy gdy nie istnieje:

- taka baza danych $\mathcal{A}$ oraz
- dwie dziedziny D_1 i D_2 zawierające dziedzinę aktywną ϕ i $\mathcal{A}$ takie, że $\phi_{D_1}(\mathcal{A}) \neq \phi_{D_2}(\mathcal{A})$.

Niezależność od dziedziny (domain independence)

Niech $\phi_B(\mathcal{A})$ oznacza zbiór krotek będący wynikiem zapytania ϕ obliczonego na bazie $\mathcal{A}$ przy założeniu, że dziedzina (zbiór możliwych wartości zmiennych) to B.

Dziedzina aktywna formuły

Zbiór D nazwiemy **dziedziną aktywną** formuły ϕ i bazy $\mathcal{A}$, gdy jest to zbiór wszystkich wartości występujących we wszystkich kolumnach wszystkich relacji z $\mathcal{A}$ oraz wszystkich stałych występujących jawnie w ϕ.

Zapytanie ϕ jest *dziedzinowo niezależne* wtedy i tylko wtedy gdy nie istnieje:

- taka baza danych $\mathcal{A}$ oraz
- dwie dziedziny D_1 i D_2 zawierające dziedzinę aktywną ϕ i $\mathcal{A}$ takie, że $\phi_{D_1}(\mathcal{A}) \neq \phi_{D_2}(\mathcal{A})$.

Zapytania dziedzinowo niezależne będziemy wyliczać na dziedzinie aktywnej i będziemy nazywać je bezpiecznymi

Niezależność od dziedziny (domain independence)

Niech $\phi_B(\mathcal{A})$ oznacza zbiór krotek będący wynikiem zapytania ϕ obliczonego na bazie $\mathcal{A}$ przy założeniu, że dziedzina (zbiór możliwych wartości zmiennych) to B.

Dziedzina aktywna formuły

Zbiór D nazwiemy **dziedziną aktywną** formuły ϕ i bazy $\mathcal{A}$, gdy jest to zbiór wszystkich wartości występujących we wszystkich kolumnach wszystkich relacji z $\mathcal{A}$ oraz wszystkich stałych występujących jawnie w ϕ.

Zapytanie ϕ jest *dziedzinowo niezależne* wtedy i tylko wtedy gdy nie istnieje:

- taka baza danych $\mathcal{A}$ oraz
- dwie dziedziny D_1 i D_2 zawierające dziedzinę aktywną ϕ i $\mathcal{A}$ takie, że $\phi_{D_1}(\mathcal{A}) \neq \phi_{D_2}(\mathcal{A})$.

Zapytania dziedzinowo niezależne będziemy wyliczać na dziedzinie aktywnej i będziemy nazywać je bezpiecznymi
Czy może zdarzyć się, że dostaniemy inny wynik niż w semantyce standardowej?

Niezależność od dziedziny (domain independence)

W informatyce nie lubimy jak formuła zależy od dziedziny.

Niezależność od dziedziny (domain independence)

W informatyce nie lubimy jak formuła zależy od dziedziny.

Sprawdzenie czy formuła jest dziedzinowo niezależna jest nierozstrzygalne!

Niezależność od dziedziny (domain independence)

W informatyce nie lubimy jak formuła zależy od dziedziny.

Sprawdzenie czy formuła jest dziedzinowo niezależna jest nierozstrzygalne!

Istnieją (jakieś) syntaktyczne ograniczenia zapewniające bezpieczeństwo.

Niezależność od dziedziny (domain independence)

W informatyce nie lubimy jak formuła zależy od dziedziny.

Sprawdzenie czy formuła jest dziedzinowo niezależna jest nierozstrzygalne!

Istnieją (jakieś) syntaktyczne ograniczenia zapewniające bezpieczeństwo.

Uwaga! Na formułę, która zwraca nieskończony wynik mówi się, że jest niebezpieczna

Niezależność od dziedziny (domain independence)

W informatyce nie lubimy jak formuła zależy od dziedziny.

Sprawdzenie czy formuła jest dziedzinowo niezależna jest nierozstrzygalne!

Istnieją (jakieś) syntaktyczne ograniczenia zapewniające bezpieczeństwo.

Uwaga! Na formułę, która zwraca nieskończony wynik mówi się, że jest niebezpieczna
Czy istnieje formuła, które nie jest bezpieczna ani niebezpieczna?

Niezależność od dziedziny (domain independence)

W informatyce nie lubimy jak formuła zależy od dziedziny.

Sprawdzenie czy formuła jest dziedzinowo niezależna jest nierozstrzygalne!

Istnieją (jakieś) syntaktyczne ograniczenia zapewniające bezpieczeństwo.

Uwaga! Na formułę, która zwraca nieskończony wynik mówi się, że jest niebezpieczna
Czy istnieje formuła, które nie jest bezpieczna ani niebezpieczna?
Praca: ON SAFETY, DOMAIN INDEPENDENCE, AND CAPTURABILITY OF DATABASE QUERIES

Niezależność od dziedziny (domain independence)

W informatyce nie lubimy jak formuła zależy od dziedziny.

Sprawdzenie czy formuła jest dziedzinowo niezależna jest nierozstrzygalne!

Istnieją (jakieś) syntaktyczne ograniczenia zapewniające bezpieczeństwo.

Uwaga! Na formułę, która zwraca nieskończony wynik mówi się, że jest niebezpieczna
Czy istnieje formuła, które nie jest bezpieczna ani niebezpieczna?
Praca: ON SAFETY, DOMAIN INDEPENDENCE, AND CAPTURABILITY OF DATABASE QUERIES

Czy zapytania algebry relacji są dziedzinowo niezależne?

Twierdzenie

Języki zapytań dla modelu relacyjnego:

- *algebra relacji,*
- *relacyjny rachunek krotek i relacyjny rachunek dziedzin*

Twierdzenie

Języki zapytań dla modelu relacyjnego:

- *algebra relacji,*
- *relacyjny rachunek krotek i relacyjny rachunek dziedzin*

nie są sobie równoważne.

Twierdzenie

Języki zapytań dla modelu relacyjnego:

- *algebra relacji,*
- *relacyjny rachunek krotek i relacyjny rachunek dziedzin*

nie są sobie równoważne.

Każde wyrażenie algebry relacji zwraca zbiór skończony!

Twierdzenie (Twierdzenie)

Języki zapytań dla modelu relacyjnego:

- *algebra relacji,*
- *relacyjny rachunek krotek ograniczony do formuł bezpiecznych i*
- *relacyjny rachunek dziedzin ograniczony do formuł bezpiecznych*

są sobie równoważne.

Twierdzenie (Twierdzenie)

Języki zapytań dla modelu relacyjnego:

- *algebra relacji,*
- *relacyjny rachunek krotek ograniczony do formuł bezpiecznych i*
- *relacyjny rachunek dziedzin ograniczony do formuł bezpiecznych*

są sobie równoważne.

Proste ćwiczenie 1: Dla każdego wyrażenia algebry relacji istnieje równoważna mu bezpieczna formuła w relacyjnym rachunku krotek.

Proste ćwiczenie 2: Dla każdej bezpiecznej formuły w relacyjnym rachunku krotek istnieje równoważna mu bezpieczna formuła w relacyjnym rachunku dziedzin.

Twierdzenie 1: Dla każdej bezpiecznej formuły w relacyjnym rachunku dziedzin istnieje równoważne mu wyrażenie algebry relacji.

Twierdzenie (Twierdzenie)

Języki zapytań dla modelu relacyjnego:

- *algebra relacji,*
- *relacyjny rachunek krotek ograniczony do formuł bezpiecznych i*
- *relacyjny rachunek dziedzin ograniczony do formuł bezpiecznych*

są sobie równoważne.

Proste ćwiczenie 1: Dla każdego wyrażenia algebry relacji istnieje równoważna mu bezpieczna formuła w relacyjnym rachunku krotek. przykłady

Proste ćwiczenie 2: Dla każdej bezpiecznej formuły w relacyjnym rachunku krotek istnieje równoważna mu bezpieczna formuła w relacyjnym rachunku dziedzin.

Twierdzenie 1: Dla każdej bezpiecznej formuły w relacyjnym rachunku dziedzin istnieje równoważne mu wyrażenie algebry relacji.

Pokażemy, że dla każdego $\phi(x_1, \ldots, x_k)$ — bezpiecznej formuły relacyjnego rachunku dziedzin o zmiennych wolnych $x_1, x_2, \ldots, x_k$ istnieje wyrażenie algebry relacji W_ϕ, którego wartością jest $\{x_1, x_2, \ldots, x_k \mid \phi(x_1, x_2, \ldots, x_k)\}$.

Pokażemy, że dla każdego $\phi(x_1, \ldots, x_k)$ — bezpiecznej formuły relacyjnego rachunku dziedzin o zmiennych wolnych $x_1, x_2, \ldots, x_k$ istnieje wyrażenie algebry relacji W_ϕ, którego wartością jest $\{x_1, x_2, \ldots, x_k \mid \phi(x_1, x_2, \ldots, x_k)\}$.

1. Zdefiniujmy dziedzinę ϕ:

$$D_\phi \; = \; \{c_1, c_2, \ldots, c_\ell\} \cup \bigcup \pi_{A \in attr(R_i)}(R_i),$$

   gdzie $c_1, c_2, \ldots, c_\ell$ to wszystkie stałe występujące w ϕ, a $R_1, R_2, \ldots, R_m$ to symbole wszystkich relacji występujących w ϕ.

Pokażemy, że dla każdego $\phi(x_1, \ldots, x_k)$ — bezpiecznej formuły relacyjnego rachunku dziedzin o zmiennych wolnych $x_1, x_2, \ldots, x_k$ istnieje wyrażenie algebry relacji W_ϕ, którego wartością jest $\{x_1, x_2, \ldots, x_k \mid \phi(x_1, x_2, \ldots, x_k)\}$.

1. Zdefiniujmy dziedzinę ϕ:

$$D_\phi = \{c_1, c_2, \ldots, c_\ell\} \cup \bigcup \pi_{A \in attr(R_i)}(R_i),$$

   gdzie $c_1, c_2, \ldots, c_\ell$ to wszystkie stałe występujące w ϕ, a $R_1, R_2, \ldots, R_m$ to symbole wszystkich relacji występujących w ϕ.

2. Przekształćmy formuły atomowe występujące w ϕ tak, by nie zawierały stałych i powtarzających się zmiennych:

$$R(x, y, x, 13) \rightarrow (\exists z, u) R(x, y, z, u) \wedge x = z \wedge u = 13$$

Pokażemy, że dla każdego $\phi(x_1, \ldots, x_k)$ — bezpiecznej formuły relacyjnego rachunku dziedzin o zmiennych wolnych $x_1, x_2, \ldots, x_k$ istnieje wyrażenie algebry relacji W_ϕ, którego wartością jest $\{x_1, x_2, \ldots, x_k \mid \phi(x_1, x_2, \ldots, x_k)\}$.

1. Zdefiniujmy dziedzinę ϕ:

$$D_\phi = \{c_1, c_2, \ldots, c_\ell\} \cup \bigcup \pi_{A \in attr(R_i)}(R_i),$$

   gdzie $c_1, c_2, \ldots, c_\ell$ to wszystkie stałe występujące w ϕ, a $R_1, R_2, \ldots, R_m$ to symbole wszystkich relacji występujących w ϕ.

2. Przekształćmy formuły atomowe występujące w ϕ tak, by nie zawierały stałych i powtarzających się zmiennych:

$$R(x, y, x, 13) \rightarrow (\exists z, u) R(x, y, z, u) \wedge x = z \wedge u = 13$$

3. Przekształćmy ϕ w ten sposób, by nie zawierała spójników $\wedge$ i kwantyfikatorów $\forall$.

Podstawa indukcji

- dla $\phi(x, y, z) \equiv R(x, y, z)$ definiujemy

Podstawa indukcji

- dla $\phi(x, y, z) \equiv R(x, y, z)$ definiujemy

$$W_\phi = R, \ \ \textit{ewentualnie} \ \ W_\phi = \rho_{x,y,z}(R)$$

Podstawa indukcji

- dla $\phi(x, y, z) \equiv R(x, y, z)$ definiujemy

$$W_\phi = R, \ \ \textit{ewentualnie} \ \ W_\phi = \rho_{x,y,z}(R)$$

- dla $\phi(x) \equiv x > \textit{const}$ definiujemy

Podstawa indukcji

- dla $\phi(x, y, z) \equiv R(x, y, z)$ definiujemy

$$W_\phi = R, \ \ \textit{ewentualnie} \ \ W_\phi = \rho_{x,y,z}(R)$$

- dla $\phi(x) \equiv x > \mathit{const}$ definiujemy

$$W_\phi = \sigma_{x > \mathit{const}}(\rho_x(D))$$

Podstawa indukcji

- dla $\phi(x, y, z) \equiv R(x, y, z)$ definiujemy

$$W_\phi = R, \ \ \textit{ewentualnie} \ \ W_\phi = \rho_{x,y,z}(R)$$

- dla $\phi(x) \equiv x > \textit{const}$ definiujemy

$$W_\phi = \sigma_{x > \textit{const}}(\rho_x(D))$$

- dla $\phi(x) \equiv x = \textit{const}$ definiujemy

Podstawa indukcji

- dla $\phi(x, y, z) \equiv R(x, y, z)$ definiujemy

$$W_\phi = R, \ \ \textit{ewentualnie} \ \ W_\phi = \rho_{x,y,z}(R)$$

- dla $\phi(x) \equiv x > \mathit{const}$ definiujemy

$$W_\phi = \sigma_{x > \mathit{const}}(\rho_x(D))$$

- dla $\phi(x) \equiv x = \mathit{const}$ definiujemy

$$W_\phi = \sigma_{x = \mathit{const}}(\rho_x(D)) \ \ \mathit{lub} \ \ \rho_x(\{\mathit{const}\})$$

Podstawa indukcji

- dla $\phi(x, y, z) \equiv R(x, y, z)$ definiujemy

$$W_\phi = R, \; \textit{ewentualnie} \; W_\phi = \rho_{x,y,z}(R)$$

- dla $\phi(x) \equiv x > \textit{const}$ definiujemy

$$W_\phi = \sigma_{x>\textit{const}}(\rho_x(D))$$

- dla $\phi(x) \equiv x = \textit{const}$ definiujemy

$$W_\phi = \sigma_{x=\textit{const}}(\rho_x(D)) \; \textit{lub} \; \rho_x(\{\textit{const}\})$$

- dla $\phi(x, y) \equiv x \neq y$ definiujemy

Podstawa indukcji

- dla $\phi(x, y, z) \equiv R(x, y, z)$ definiujemy

$$W_\phi = R, \ \ \textit{ewentualnie} \ \ W_\phi = \rho_{x,y,z}(R)$$

- dla $\phi(x) \equiv x > \textit{const}$ definiujemy

$$W_\phi = \sigma_{x>\textit{const}}(\rho_x(D))$$

- dla $\phi(x) \equiv x = \textit{const}$ definiujemy

$$W_\phi = \sigma_{x=\textit{const}}(\rho_x(D)) \ \ \textit{lub} \ \ \rho_x(\{\textit{const}\})$$

- dla $\phi(x, y) \equiv x \neq y$ definiujemy

$$W_\phi = \sigma_{x \neq y}(\rho_x(D) \times \rho_y(D))$$

Krok indukcyjny

- dla $\phi(x) \equiv \neg\psi(x)$ i wyrażenia W_ψ z atrybutem x definiujemy

Krok indukcyjny

- dla $\phi(x) \equiv \neg\psi(x)$ i wyrażenia W_ψ z atrybutem x definiujemy

$$W_\phi = \rho_x(D) \setminus W_\psi$$

Krok indukcyjny

- dla $\phi(x) \equiv \neg\psi(x)$ i wyrażenia W_ψ z atrybutem x definiujemy

$$W_\phi = \rho_x(D) \setminus W_\psi$$

- dla $\phi(x, y, z) \equiv \psi(x, y) \vee \eta(y, z)$ i wyrażeń W_ψ i W_η z atrybutami odpowiednio: x, y oraz y, z definiujemy

Krok indukcyjny

- dla $\phi(x) \equiv \neg\psi(x)$ i wyrażenia W_ψ z atrybutem x definiujemy

$$W_\phi = \rho_x(D) \setminus W_\psi$$

- dla $\phi(x,y,z) \equiv \psi(x,y) \vee \eta(y,z)$ i wyrażeń W_ψ i W_η z atrybutami odpowiednio: x, y oraz y, z definiujemy

$$W_\phi = (W_\psi \times \rho_z(D)) \cup (\rho_x(D) \times W_\eta)$$

Krok indukcyjny

- dla $\phi(x) \equiv \neg\psi(x)$ i wyrażenia W_ψ z atrybutem x definiujemy
$$W_\phi = \rho_x(D) \setminus W_\psi$$
- dla $\phi(x, y, z) \equiv \psi(x, y) \vee \eta(y, z)$ i wyrażeń W_ψ i W_η z atrybutami odpowiednio: x, y oraz y, z definiujemy
$$W_\phi = (W_\psi \times \rho_z(D)) \cup (\rho_x(D) \times W_\eta)$$
- dla $\phi(x) \equiv (\exists y)\psi(x, y)$ i wyrażenia W_ψ z atrybutami x, y definiujemy

Krok indukcyjny

- dla $\phi(x) \equiv \neg\psi(x)$ i wyrażenia W_ψ z atrybutem x definiujemy
$$W_\phi = \rho_x(D) \setminus W_\psi$$
- dla $\phi(x,y,z) \equiv \psi(x,y) \vee \eta(y,z)$ i wyrażeń W_ψ i W_η z atrybutami odpowiednio: x, y oraz y, z definiujemy
$$W_\phi = (W_\psi \times \rho_z(D)) \cup (\rho_x(D) \times W_\eta)$$
- dla $\phi(x) \equiv (\exists y)\psi(x,y)$ i wyrażenia W_ψ z atrybutami x, y definiujemy
$$W_\phi = \pi_x(W_\psi)$$

Wnioski

Wnioski

1. Mamy trzy równoważne języki zapytań dla modelu relacyjnego:
   - Dwa z nich, rachunki relacyjne, są deklaratywne — formułując zapytanie podajemy jego znaczenie, a nie sposób obliczania.
   - Jeden z nich, algebra relacji, jest imperatywny — pisząc wyrażenie podajemy sposób wyliczania, ale znaczenie wyrażenia nie musi być jasne.

Wnioski

1. Mamy trzy równoważne języki zapytań dla modelu relacyjnego:
   - Dwa z nich, rachunki relacyjne, są deklaratywne — formułując zapytanie podajemy jego znaczenie, a nie sposób obliczania.
   - Jeden z nich, algebra relacji, jest imperatywny — pisząc wyrażenie podajemy sposób wyliczania, ale znaczenie wyrażenia nie musi być jasne.
2. Moc, czyli możliwości ekspresji rachunków relacyjnych, jest bardzo dobrze znana:
   - to logika pierwszego rzędu, w której można opisać wiele własności,
   - nie można jednak wyrazić np. domknięcia tranzytywnego, do czego potrzebne jest kwantyfikowanie po relacjach (powiedzenie, że "istnieje relacja, taka że. . .),
   - łatwiej zastanawiać się czy zapytania są równoważne lub zawierają się w sobie (optymalizacje!).

Zapytania koniunkcyjne

- Wiele naturalnych własności można wyrazić wyrażeniami algebry relacji używającymi wyłącznie selekcji, projekcji, złączeń i przemianowań.

Zapytania koniunkcyjne

- Wiele naturalnych własności można wyrazić wyrażeniami algebry relacji używającymi wyłącznie selekcji, projekcji, złączeń i przemianowań.
- Fragmentowi temu odpowiadają zapytania koniunkcyjne, są to formuły rrd/rrk postaci

$$\varphi(\vec{x}) = \exists \vec{y} \bigwedge_i R_i(\vec{x}, \vec{y})$$

Zapytania koniunkcyjne

- Wiele naturalnych własności można wyrazić wyrażeniami algebry relacji używającymi wyłącznie selekcji, projekcji, złączeń i przemianowań.
- Fragmentowi temu odpowiadają zapytania koniunkcyjne, są to formuły rrd/rrk postaci

$$\varphi(\vec{x}) = \exists \vec{y} \bigwedge_i R_i(\vec{x}, \vec{y})$$

- Co właściwie da się zapisać w tak prostym języku?

Zapytania koniunkcyjne

- Wiele naturalnych własności można wyrazić wyrażeniami algebry relacji używającymi wyłącznie selekcji, projekcji, złączeń i przemianowań.
- Fragmentowi temu odpowiadają zapytania koniunkcyjne, są to formuły rrd/rrk postaci

$$\varphi(\vec{x}) = \exists \vec{y} \bigwedge_i R_i(\vec{x}, \vec{y})$$

- Co właściwie da się zapisać w tak prostym języku? przykłady

Baza danych

Na każdą bazę danych możemy patrzeć jak na

- zbiór tabel,
- strukturę relacyjną,
- zbiór faktów.