

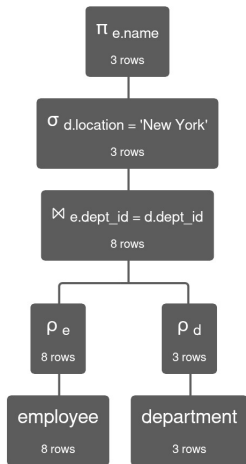
# Relačná algebra

Ján Mazák

FMFI UK Bratislava

- ▶ interný jazyk, do ktorého sa prekladajú všetky dotazy
- ▶ tiež jazyk na formalizáciu relačného modelu a matematické dokazovanie
- ▶ zachytáva postup výpočtu dotazu pomocou **logických operátorov** (nezohľadňujú fyzické uloženie dát)
- ▶ vstupom aj výstupom operátora je relácia
- ▶ k danému dotazu možno zostrojiť rôzne zápisy (operátorové stromy) v relačnej algebre, databáza si sama vyberie ten, čo pokladá za najvhodnejší

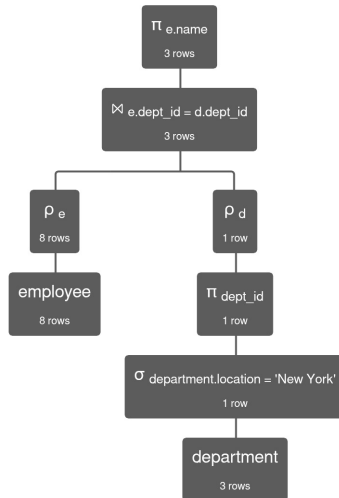
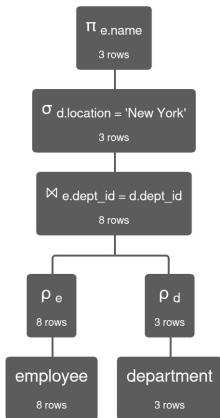
# Relačná algebra



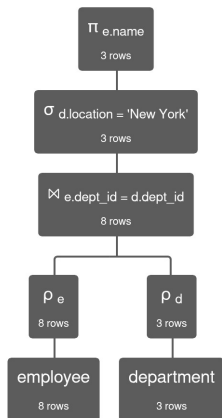
```
SELECT e.name
FROM employee AS e
      JOIN department AS d
      ON e.dept_id = d.dept_id
WHERE
      d.location = 'New York';
```

# Relačná algebra

Porovnajte rýchlosť výpočtu:



# Relačná algebra



$e := \rho_e(\text{employee})$

$d := \rho_d(\text{department})$

$\pi_{e.name}(\sigma_{d.location='New York'}($   
 $e \bowtie_{e.dept\_id=d.dept\_id} d$   
 $))$

*/\* example EXPLAIN output \*/*

Filter: (d.location = 'New York')

-> Hash Join

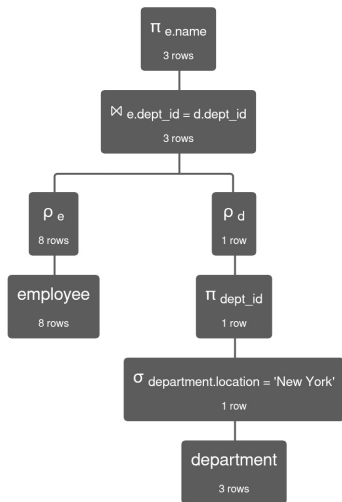
Hash Cond: (e.dept\_id = d.dept\_id)

-> Seq Scan on employee e

-> Hash

-> Seq Scan on department d

# Relačná algebra



$e := \rho_e(\text{employee})$

$d := \rho_d(\text{department})$

$\pi_{e.name}(e \bowtie (\pi_{d.dept\_id}(\sigma_{d.location='New York'}(d))))$

`/* example EXPLAIN output */`

Hash Join

Hash Cond: (e.dept\_id = d.dept\_id)

-> Seq Scan on employee e

-> Hash

-> Seq Scan on department d

Filter: (location = 'New York')

# Logické operátory

- ▶  $\pi$  — projekcia (vyberáme stĺpce)
- ▶  $\sigma$  — selekcia (vyberáme riadky)
- ▶  $\rho$  — premenovanie (relácie či atribútu)
- ▶  $\times$  — karteziánsky súčin
- ▶  $\bowtie$  — natural join
- ▶  $\bowtie_{\theta}$  — theta-join (join s podmienkou  $\theta$ )
- ▶  $\triangleright$  — antijoin (riadky 1. relácie, ktoré sa nedajú joinovať so žiadnymi riadkami 2. relácie)
- ▶  $-, \cup, \cap$  — rozdiel, zjednotenie, prienik množín
- ▶  $\Gamma$  or  $\gamma$  — group by

# Ukážky relačnej algebry

Databáza: *lubi*(Pijan, Alkohol), *capuje*(Krcma, Alkohol, Cena),  
*navstivil*(Id, Pijan, Krcma), *vypil*(Id, Alkohol, Mnozstvo)

1. Alkoholy, ktoré niekto ľúbi, ale nikde ich nečapujú

$$\pi_{Alkohol}(lubi) - \pi_{Alkohol}(capuje)$$

$$\pi_{lubi.Alkohol}(lubi \triangleright_{lubi.Alkohol=capuje.Alkohol} capuje)$$

2. Počet vypití piva pre jednotlivých pijanov

$$\Gamma_{Pijan, COUNT(Id) \rightarrow C}(\sigma_{Alkohol='pivo'}(navstivil \bowtie vypil))$$



# Ukážky relačnej algebry

```
SELECT a1, a2, COUNT(a3) AS b  
FROM r1, r2  
WHERE c1 OR c2  
GROUP BY g1, g2  
HAVING h1 AND h2
```

$$\pi_{a_1, a_2, b}(\sigma_{h_1 \wedge h_2}(\Gamma_{g_1, g_2, COUNT(a_3) \rightarrow b}(r_1 \bowtie_{c_1 \vee c_2} r_2)))$$

$$j := r_1 \bowtie_{c_1 \vee c_2} r_2$$

$$\pi_{a_1, a_2, b}(\sigma_{h_1 \wedge h_2}(\Gamma_{g_1, g_2, COUNT(a_3) \rightarrow b}(j)))$$

- ▶ <https://www.db-book.com/slides-dir/PDF-dir/ch2.pdf>
- ▶ [https://drive.google.com/file/d/1IwVFcAWWDD\\_fAJA0ZruXd1S30nh3XinP/view](https://drive.google.com/file/d/1IwVFcAWWDD_fAJA0ZruXd1S30nh3XinP/view)
- ▶ <https://cs186berkeley.net/notes/note6/>
- ▶ <https://dbis-uibk.github.io/relax/calc/gist/379b0fdd72490e8e634bb193f109d4a8>

# Úlohy: relačná algebra

Databáza: *lubi*(Pijan, Alkohol), *capuje*(Krcma, Alkohol, Cena), *navstivil*(Id, Pijan, Krcma), *vypil*(Id, Alkohol, Mnozstvo)

- ▶ pijani, čo ľúbia pivo
- ▶ koľko stojí najlacnejšie pivo?
- ▶ alkoholy, ktoré čapujú, ale nik ich neľúbi
- ▶ alkoholy, ktoré čapujú, ale nik ich nepil
- ▶ najdrahší čapovaný alkohol (všetky, ak ich je viac)
- ▶ pijani, ktorí navštívili všetky krčmy, čo niečo čapujú
- ▶ krčma s najväčšou celkovou tržbou