B0B36DBS: **Database Systems**

http://www.ksi.mff.cuni.cz/~svoboda/courses/202-B0B36DBS/

Practical Class 9

Query Evaluation

Author: **Martin Svoboda**, martin.svoboda@fel.cvut.cz
Tutors: **Ahmad**, **Černoch**, **Kostov**, **Nagyová**, **Řimnáč**, **Svoboda**, **Šír**

13. 4. 2021

Czech Technical University in Prague, Faculty of Electrical Engineering

Database Schema

Ticketing information system

- **Passenger** (pid, name, email, phone, address, category)
  - $r_P = 30000$ tuples
  - Blocking factor $b_P = 15$
  - Histogram for passenger **categories**
    - $h_{P.category.Child} = 0.1$
    - $h_{P.category.Student} = 0.3$
    - $h_{P.category.Adult} = 0.5$
    - $h_{P.category.Senior} = 0.1$
  - **Sorted file** using `pid`
  - **Clustered B$^+$-tree index** using `pid`
    - $f_{P.pid} = 200$
    - $I_{P.pid} = 2$
    - $p_{P.pid} = 175$

Database Schema

Ticketing information system (cont'd)

- **Ticket** (tid, connection, passenger, data)
  - $r_T = 900000$ tuples
  - Blocking factor $b_T = 40$
  - **Connection** references
    - Foreign key: Ticket (connection) $\subseteq$ Connection (cid)
    - $V_{T.connection} = r_C = 20000$ different connections
  - **Passenger** references
    - Foreign key: Ticket (passenger) $\subseteq$ Passenger (pid)
    - $h_{T.passenger.NULL} = 1/3$
    - I.e., not all tickets are sold to registered passengers
  - **Heap file**

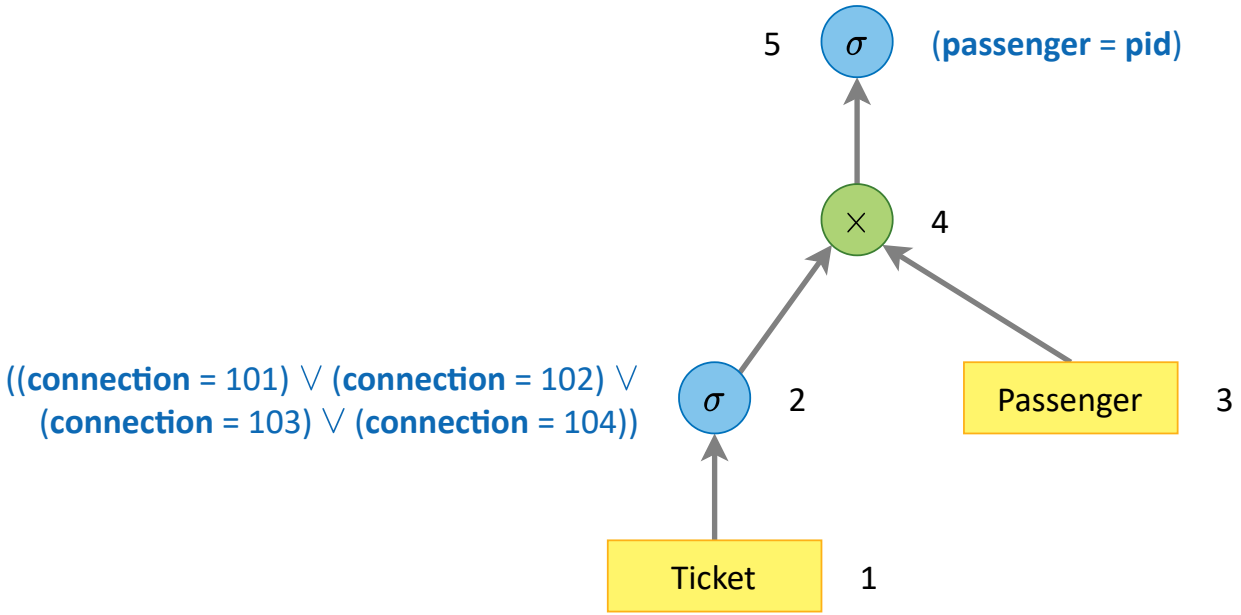
Exercise 1

Tickets with passenger details for connections with identifiers 101, 102, 103, and 104

$$(\text{Ticket}((\text{connection} = 101) \vee (\text{connection} = 102) \vee (\text{connection} = 103) \vee (\text{connection} = 104)) \times \text{Passenger})(\text{passenger} = \text{pid})$$

- Requirements
  - Cross join using nested loops with zig-zag improvement
- Available system memory: $M = 5$ pages

Exercise 1


5
σ
(passenger = pid)
×
4
((connection = 101) ∨ (connection = 102) ∨
(connection = 103) ∨ (connection = 104))
σ
2
Passenger
3
Ticket
1

Database Schema

Ticketing information system (cont'd)

- **Deposit** (did, date, time, passenger, amount)
  - $r_D = 300000$ tuples, blocking factor $b_D = 60$
  - Active domain for **dates**
    - $min_{D.date}$ ='2021-01-01' and $max_{D.date}$ ='2021-04-10'
  - Histogram for **amounts**
    - $h_{D.amount.[0..500)} = 0.80$
    - $h_{D.amount.[500..1000)} = 0.15$
    - $h_{D.amount.[1000..)} = 0.05$
  - **Passenger** references
    - Foreign key: Deposit (passenger) $\subseteq$ Passenger (pid)
    - Deposits can only be made by registered passengers
    - $V_{D.passenger} = r_P = 30000$ different passengers
  - **Sorted file** using `date` and `time`

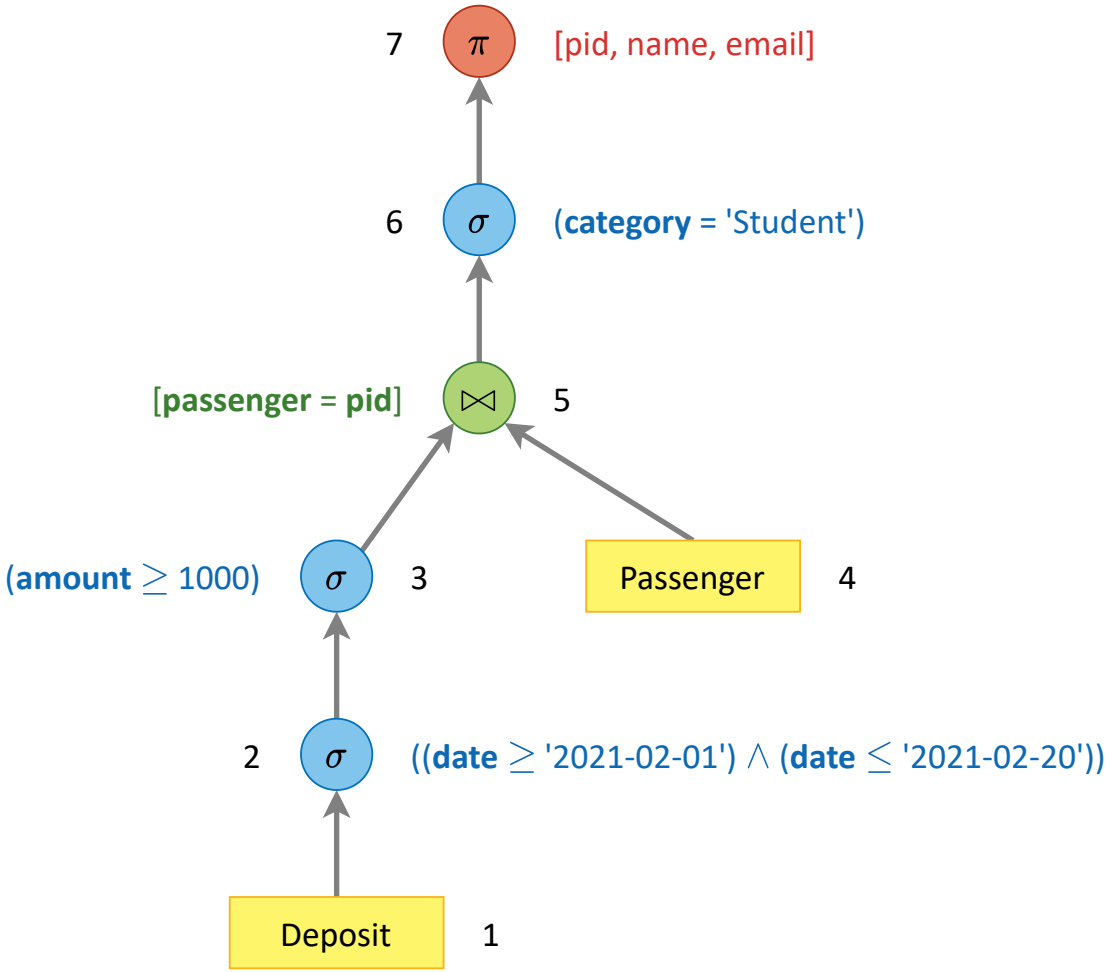
Exercise 2

Student passengers with at least one deposit of at least 1000 CZK between February 1 and 20, 2021

```
(
    Deposit ((date ≥ '2021-02-01') ∧ (date ≤ '2021-02-20'))
        (amount ≥ 1000)
    [passenger = pid]
    Passenger
) (category = 'Student') [pid, name, email]
```

- Requirements
  - Theta join using nested loops with passenger index traversal
- Available system memory: $M = 5$ pages
- Assumptions: blocking factor for projection $b_7 = 30$

Exercise 2


7
π
[pid, name, email]
6
σ
(category = 'Student')
[passenger = pid]
⋈
5
(amount ≥ 1000)
σ
3
Passenger
4
2
σ
((date ≥ '2021-02-01') ∧ (date ≤ '2021-02-20'))
Deposit
1

Database Schema

Ticketing information system (cont'd)

- **Connection** (cid, date, time, origin, destination, capacity)
  - $r_C = 20000$ tuples
  - Blocking factor $b_C = 50$
  - Active domain for **dates**
    - $min_{C.date}$ ='2021-01-01' and $max_{C.date}$ ='2021-04-10'
  - Concealed **motivation**
    - Two-way star-shaped lines between Prague and 10 other cities
    - 10 connections each day in both directions
  - Histogram for origin and destination **cities**
    - $h_{C.origin.Prague} = h_{C.destination.Prague} = 1/2$
    - $h_{C.origin.c} = h_{C.destination.c} = 1/20$ for any other city c
  - **Hashed file** using `date`
    - $K_C = 80$ buckets

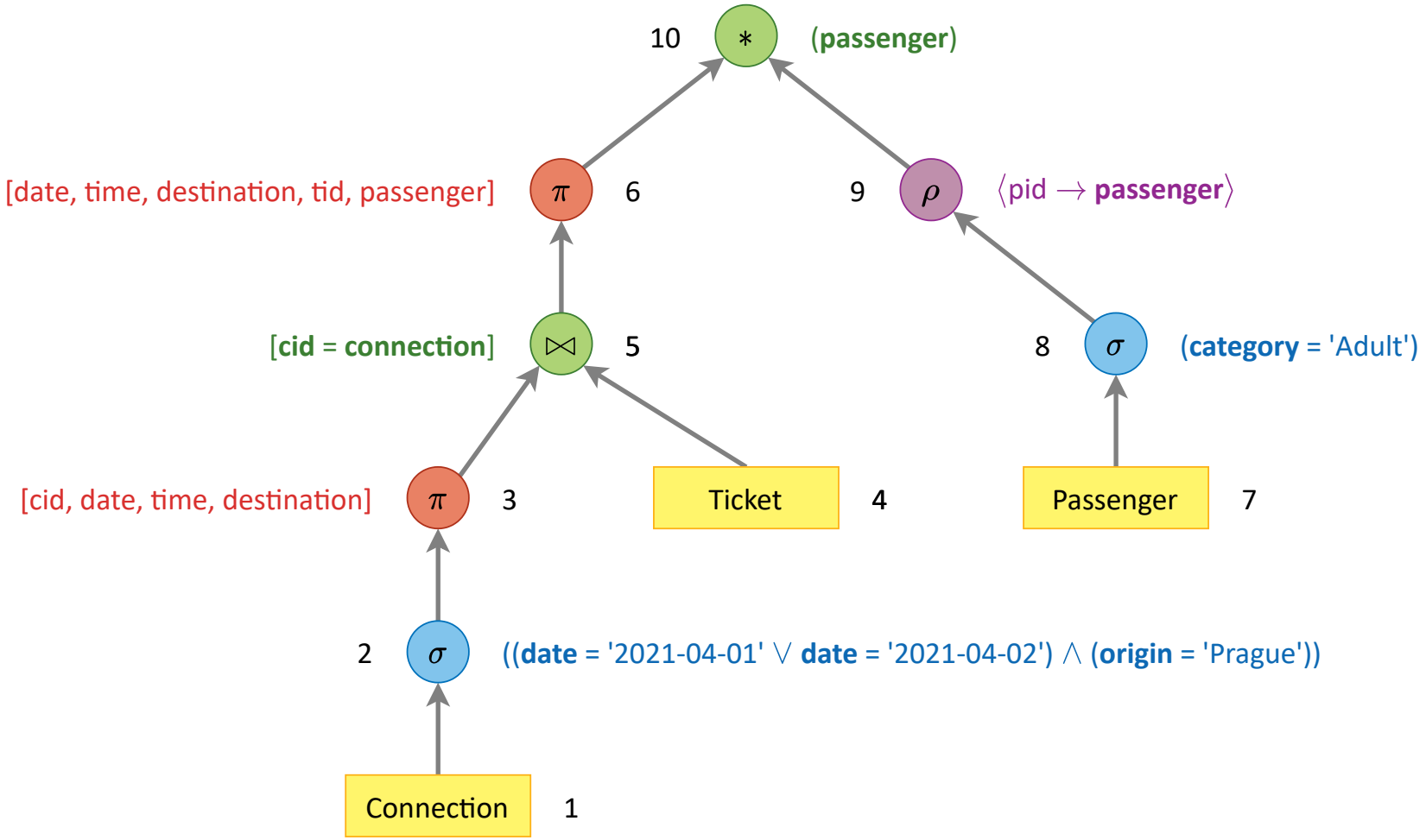
Exercise 3

Tickets of adult passengers leaving Prague on April 1 and 2, 2021

$$
\begin{array}{l}
(\\
\quad \textbf{Connection}\ ((\textbf{date} = \texttt{'2021-04-01'} \vee \textbf{date} = \texttt{'2021-04-02'}) \wedge \\
\qquad (\textbf{origin} = \texttt{'Prague'}))\ [\text{cid, date, time, destination}] \\
\quad [\textbf{cid} = \textbf{connection}] \\
\quad \textbf{Ticket} \\
)\ [\text{date, time, destination, tid, passenger}] \\
* \\
\textbf{Passenger}\ (\textbf{category} = \texttt{'Adult'})\ \langle \text{pid} \rightarrow \textbf{passenger} \rangle
\end{array}
$$

- Requirements
  - Theta join using nested loops, natural join using sort-merge join
- Available system memory: $M = 20$ pages
- Assumptions: blocking factors for projections $b_3 = 60$, $b_6 = 30$

Exercise 3


10
∗
(passenger)
[date, time, destination, tid, passenger]
π
6
9
ρ
⟨pid → passenger⟩
[cid = connection]
⋈
5
8
σ
(category = 'Adult')
[cid, date, time, destination]
π
3
Ticket
4
Passenger
7
2
σ
((date = '2021-04-01' ∨ date = '2021-04-02') ∧ (origin = 'Prague'))
Connection
1