**CS 133: Databases**

Fall 2018  
Lec 14 – 10/25  
Prof. Beth Truskowsky

---

**Goals for Today**

- Continue to reason about estimating the result cardinality for selections and joins  
  - System R heuristics  
  - More advanced: histograms – Lab 3!

- Begin to explore the search space explosion for alternate query plans

---

**Warm-up Exercise**

(See exercise sheet. You can start before class.)

(a) ORDER BY(day), two-way external merge sort  
   SELECT(bid=42), on the fly  
   SEQ SCAN(Reserves)

(b) $1000 + (0 + 10) + (10 + 10) + (10 + 10) + (10 + 10) + (10 + 0)$  
   = 1080 I/Os

---

**Cost-based Query Sub-System**

![Diagram of cost-based query system with labels](Diagram)
Result Size Estimation for Joins

- For equi-join of R and S, range of result sizes (# tuples)
  - If R and S have no join attribute values in common?
  
  - If join attributes are a key for S?

- And if the join attributes also comprise a foreign key in R?

Result Size Estimation for Joins

- General case: relations have join attributes \( a \) in common, \( a \) is a key for neither
  
  - Assumption: set of distinct R.\( a \) values is contained in S.\( a \)
  
  - \( (N\text{Keys} = \text{number of distinct values}) \)

  - Idea: each tuple of R has a \( \frac{1}{N\text{Keys}(S)} \) chance of joining
    with each tuple in S

  
  \[
  \text{NTuples}(R) \times \frac{\text{NTuples}(S)}{N\text{Keys}(S)}
  \]

  - Reversing above yields

  \[
  \text{NTuples}(S) \times \frac{\text{NTuples}(R)}{N\text{Keys}(R)}
  \]

  (use smaller of two if different)

Exercise 2-3

2. Estimate the result cardinality for this SQL query:

   ```sql
   SELECT * 
   FROM Sailors NATURAL JOIN Reserves 
   NATURAL JOIN Boats;
   ```

   Answer: number of tuples in Reserves (1000 pages, with 100 tuples/page)

3. Estimate the cost in I/Os of this query plan:

   ![Query Plan Diagram]

   Answer:
   Join 1 I/O cost = 1000+10*500
   Join 1 produces 10 pages of tuples, which is then filtered to 1 page
   Join 2 I/O cost = 1*200
   Total = 6000 + 200 = 6200 I/Os
Histograms: Finer-Grained Statistics

- For better RF estimation, many systems use histograms
- Histogram is approximation of a data distribution

- Example: ratings of Sailors (40,000 total tuples)

Equi-width vs. Equi-depth Histograms

- Equi-width
  - # values represented by each bucket is the same

- Equi-depth
  - # of records in each bucket is about the same

Which has better selectivity estimate for a very frequently occurring value?

Exercise 4: Histograms

Exercise 4

a) 41 values into 10 buckets. 4 in each, last one 5

b) Amount within bucket = 0.25*h_b tuples
   \[ \Rightarrow \text{Overall amount} = \frac{0.25 \times h_b \text{ tuples}}{n_{\text{tups}}} \]

c) 0.25*h_b tuples + all tuples from buckets i > b
   \[ \Rightarrow \text{Divide sum by } n_{\text{tups}} \]

d) 0
Creating Equi-width histograms

- Suppose you want to be able to estimate the selectivity (reduction factor) for this query:

```
SELECT * FROM Sailors S
WHERE S.age = 40 AND S.rating > 5;
```

- Recall that we assume independence of terms and so the filter’s RF is the product of the terms’ RFs

- Discuss with a neighbor:
  - How many histograms would we need?
  - Suppose you want to create new histogram(s) on an existing relation. Brainstorm what you would need to do. Think of the functionality from Exercise 4.