CS 457: Database Management Systems

Query Plan Cost Estimation
Query Optimization Summary

Goal: find a physical plan that has minimal cost

What is the cost of a plan?
For each operator, cost is function of CPU, IO, network

Total Cost = CPUCost + w_{IO} IOCost + w_{BW} BWCost

Cost of plan is total for all operators
In this class, we look only at IO
Query Optimization Summary

Goal: find a physical plan that has minimal cost
Query Optimization Summary

Goal: find a physical plan that has minimal cost

Know how to compute cost if know cardinalities
Goal: find a physical plan that has minimal cost

Know how to compute cost if know cardinalities
Query Optimization Summary

Goal: find a physical plan that has minimal cost

Know how to compute cost if know cardinalities

– Eg. \( \text{Cost}(V \bowtie T) = 3B(V) + 3B(T) \)
– \( B(V) = T(V) / \text{PageSize} \)
– \( T(V) = T(\sigma(R) \bowtie S) \)
Query Optimization Summary

Goal: find a physical plan that has minimal cost

Know how to compute cost if know cardinalities

- Eg. Cost(V \bowtie T) = 3B(V) + 3B(T)
- B(V) = T(V) / PageSize
- T(V) = T(\sigma(R) \bowtie S)

Cardinality estimation problem: e.g. estimate T(\sigma(R) \bowtie S)
Database Statistics

- **Collect** statistical summaries of stored data

- **Estimate size** (=cardinality) in a bottom-up fashion
  - This is the most difficult part, and still inadequate in today’s query optimizers

- **Estimate cost** by using the estimated size
  - Hand-written formulas, similar to those we used for computing the cost of each physical operator
Database Statistics

- Number of tuples (cardinality) $T(R)$
- Indexes, number of keys in the index $V(R,a)$
- Number of physical pages $B(R)$
- Statistical information on attributes
  - Min value, Max value, $V(R,a)$
- Histograms

- Collection approach: periodic, using sampling
Size Estimation Problem

\[ Q = \text{SELECT list} \]
FROM R1, ..., Rn
WHERE \( \text{cond}_1 \text{ AND cond}_2 \text{ AND} \ldots \text{ AND cond}_k \)

Given \( T(R1), T(R2), \ldots, T(Rn) \)
Estimate \( T(Q) \)

How can we do this? Note: doesn’t have to be exact.
Size Estimation Problem

\[ Q = \text{SELECT list} \]
\[ \text{FROM R1, \ldots, Rn} \]
\[ \text{WHERE cond}_1 \text{ AND cond}_2 \text{ AND \ldots AND cond}_k \]

Remark: \( T(Q) \leq T(R1) \times T(R2) \times \ldots \times T(Rn) \)
Size Estimation Problem

Q = \texttt{SELECT} \texttt{list}
\hspace{1cm} \texttt{FROM} \ R1, \ldots, \ Rn
\hspace{1cm} \texttt{WHERE} \ \texttt{cond}_1 \ \texttt{AND} \ \texttt{cond}_2 \ \texttt{AND} \ldots \ \texttt{AND} \ \texttt{cond}_k

Remark: \( T(Q) \leq T(R1) \times T(R2) \times \ldots \times T(Rn) \)

Key idea: each condition reduces the size of \( T(Q) \) by some factor, called \texttt{selectivity factor}
Selectivity Factor

• Each condition \textit{cond} reduces the size by some factor called \textit{selectivity factor}

• Assuming independence, \textit{multiply} the selectivity factors
Example

Q = SELECT * 
FROM R, S, T 
WHERE R.B=S.B and S.C=T.C and R.A<40

R(A,B)  
S(B,C)  
T(C,D)

T(R) = 30k, T(S) = 200k, T(T) = 10k

Selectivity of R.B = S.B is 1/3
Selectivity of S.C = T.C is 1/10
Selectivity of R.A < 40 is ½

Q: What is the estimated size of the query output T(Q)?
Example

\[ Q = \text{SELECT } * \text{ FROM } R, S, T \text{ WHERE } R.B = S.B \text{ and } S.C = T.C \text{ and } R.A < 40 \]

\[ T(R) = 30k, \ T(S) = 200k, \ T(T) = 10k \]

Selectivity of \( R.B = S.B \) is 1/3
Selectivity of \( S.C = T.C \) is 1/10
Selectivity of \( R.A < 40 \) is 1/2

Q: What is the estimated size of the query output \( T(Q) \)?

A: \[ T(Q) = 30k \times 200k \times 10k \times \frac{1}{3} \times \frac{1}{10} \times \frac{1}{2} = 10^{12} \]
Selectivity Factors for Conditions

- $A = c$ /* $\sigma_{A=c}(R)$ */
  - Selectivity $= 1/V(R,A)$
Selectivity Factors for Conditions

- $A = c$  
  \[ \text{Selectivity} = \frac{1}{V(R,A)} \]

- $A < c$  
  \[ \text{Selectivity} = \frac{c - \text{Low}(R,A)}{\text{High}(R,A) - \text{Low}(R,A)} \]
Selectivity Factors for Conditions

• \( A = c \)  
  
  \[ \text{Selectivity} = 1 / V(R, A) \]

• \( A < c \)  
  
  \[ \text{Selectivity} = (c - \text{Low}(R, A)) / (\text{High}(R, A) - \text{Low}(R, A)) \]

• \( A = B \)  
  
  \[ \text{Selectivity} = 1 / \max(V(R, A), V(S, B)) \]
  
  (will explain next)
Assumptions

• **Containment of values**: if $V(R,A) \leq V(S,B)$, then all values $R.A$ occur in $S.B$
  
  – Note: this indeed holds when $A$ is a foreign key in $R$, and $B$ is a key in $S$

• **Preservation of values**: for any other attribute $C$, $V(R \bowtie_{A=B} S, C) = V(R, C)$ (or $V(S, C)$)
  
  – Note: we don’t need this to estimate the size of the join, but we need it in estimating the next operator
Selectivity of $R \bowtie_{A=B} S$

Assume $V(R,A) \leq V(S,B)$

- A tuple $t$ in $R$ joins with $T(S)/V(S,B)$ tuple(s) in $S$
  
- Hence $T(R \bowtie_{A=B} S) = T(R) \cdot T(S) / V(S,B)$

\[
T(R \bowtie_{A=B} S) = T(R) \cdot T(S) / \max(V(R,A),V(S,B))
\]
Size Estimation for Join

Example:

- \(T(R) = 10000, \ T(S) = 20000\)
- \(V(R,A) = 100, \ V(S,B) = 200\)
- How large is \(R \bowtie_{A=B} S\) ?

(In class…)

\[
= 10K \times 20K / \max(100, 200) \\
= 10 \times 20 K / 200 \\
= 1M
\]
Complete Example

Supplier($\text{sid}$, $\text{sname}$, $\text{scity}$, $\text{sstate}$)
Supply($\text{sid}$, $\text{pno}$, $\text{quantity}$)

- Some statistics
  - $T(\text{Supplier}) = 1000$ records
  - $T(\text{Supply}) = 10,000$ records
  - $B(\text{Supplier}) = 100$ pages
  - $B(\text{Supply}) = 100$ pages
  - $V(\text{Supplier}, scity) = 20, V(\text{Suppliers}, state) = 10$
  - $V(\text{Supply}, pno) = 2,500$
  - Both relations are clustered

- $M = 11$

SELECT $\text{sname}$
FROM Supplier $x$, Supply $y$
WHERE $x.\text{sid} = y.\text{sid}$
and $y.\text{pno} = 2$
and $x.\text{scity} = \text{‘Seattle’}$
and $x.\text{sstate} = \text{‘WA’}$
Computing the Cost of a Plan

• Estimate **cardinality** in a bottom-up fashion
  – Cardinality is the **size** of a relation (nb of tuples)
  – Compute size of *all* intermediate relations in plan

• Estimate **cost** by using the estimated cardinalities
Physical Query Plan 1

**π_{sname}**

Selection and project on-the-fly

\(\sigma_{\text{scity}='Seattle' \land \text{sstate}='WA' \land \text{pno}=2}\)

Total cost of plan is thus cost of join:

\[= B(\text{Supplier}) + B(\text{Supplier}) \times B(\text{Supply})\]

\[= 100 + 100 \times 100\]

\[= 10,100 \text{ I/Os}\]
Physical Query Plan 2

(On the fly)

(Scan write to T1)
(a) $\sigma_{\text{scity}=\text{Seattle} \land \text{ststate}=\text{WA}}$

(Sort-merge join)

(b) $\sigma_{\text{pno}=2}$

(Scan write to T2)

Total cost $\approx 204$ I/Os

Total cost $= 100 + 100 \times 1/20 \times 1/10 \ (a) + 100 + 100 \times 1/2500 \ (b) + 2 \ (c) \ \text{why? (output: 1+1)} + 0 \ (d)$

B(Supplier) = 100
B(Supply) = 100
V(Supplier,scity) = 20
V(Supplier,state) = 10
V(Supply,pno) = 2,500

T(Supplier) = 1000
T(Supply) = 10,000

M = 11

($\pi_{\text{sname}}$)

sno = sno
Plan 2 with Different Numbers

What if we had:
10K pages of Supplier
10K pages of Supply
(Sort-merge join)

\[ \pi_{sname} \]
\[ \sigma_{scity='Seattle' \land sstate='WA'} \]
\[ \sigma_{pno=2} \]

Total cost
\[ = 10000 + 50 \] (a)
\[ + 10000 + 4 \] (b)
\[ + 3\times50 + 4 \] (c)
\[ + 0 \] (d)
\[ \approx 20,208 \text{ I/Os} \]

Need to do a two-pass sort algorithm
Physical Query Plan 3

(On the fly) (d) $\pi_{\text{sname}}$

(On the fly)

(On the fly)

(Use hash index)

(a) $\sigma_{\text{pno}=2}$

Supply

(Hash index on pno)

Assume: clustered

4 tuples

(Total cost)

= 1 (a)

+ 4 (b) why?

+ 0 (c)

+ 0 (d)

Total cost $\approx 5$ I/Os

(Use hash index) (c) $\sigma_{\text{scity}=\text{Seattle} \land \text{sstate}=\text{WA}}$

[Index nested loop]

(b) $\text{sno} = \text{sno}$

Supplier

(Hash index on sno)

Clustering does not matter

Supply

(B(Supplier) = 100)

V(Supplier,scity) = 20

V(Supplier,state) = 10

V(Supply,pno) = 2,500

M = 11

T(Supplier) = 1000

T(Supply) = 10,000

B(Supplier) = 100

B(Supply) = 100

V(Supplier,scity) = 20

V(Supplier,state) = 10

V(Supply,pno) = 2,500

M = 11
Histograms

- Statistics on data maintained by the RDBMS
- Makes size estimation much more accurate (hence, cost estimations are more accurate)
Histograms

Employee(\texttt{ssn}, \texttt{name}, \texttt{age})

\(T(\text{Employee}) = 25000, \quad V(\text{Empolyee, age}) = 50\)
\(\min(\text{age}) = 19, \quad \max(\text{age}) = 68\)

\(\sigma_{\text{age}=48}(\text{Empolyee}) = ? \quad \sigma_{\text{age}>28 \text{ and age}<35}(\text{Empolyee}) = ?\)
Employee(ssn, name, age)

\[ T(\text{Employee}) = 25000, \quad V(\text{Employee}, \text{age}) = 50 \]
\[ \min(\text{age}) = 19, \quad \max(\text{age}) = 68 \]

\[ \sigma_{\text{age}=48}(\text{Employee}) = ? \quad \sigma_{\text{age}>28 \text{ and age}<35}(\text{Employee}) = ? \]

Estimate = 25000 / 50 = 500  Estimate = 25000 * 6 / 50 = 3000
Histograms

Employee(ssn, name, age)

\[ T(\text{Employee}) = 25000, \quad V(\text{Employee}, \text{age}) = 50 \]
\[ \min(\text{age}) = 19, \quad \max(\text{age}) = 68 \]

\[ \sigma_{\text{age}=48}(\text{Employee}) = ? \quad \sigma_{\text{age}>28 \text{ and } \text{age}<35}(\text{Employee}) = ? \]

<table>
<thead>
<tr>
<th>Age:</th>
<th>0..20</th>
<th>20..29</th>
<th>30-39</th>
<th>40-49</th>
<th>50-59</th>
<th>&gt; 60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tuples</td>
<td>200</td>
<td>800</td>
<td>5000</td>
<td>12000</td>
<td>6500</td>
<td>500</td>
</tr>
</tbody>
</table>
Histograms

Employee(ssn, name, age)

T(Employee) = 25000, V(Employyee, age) = 50
min(age) = 19, max(age) = 68

\[ \sigma_{age=48}(Employee) = ? \quad \sigma_{age>28 \text{ and } age<35}(Employee) = ? \]

<table>
<thead>
<tr>
<th>Age:</th>
<th>0..20</th>
<th>20..29</th>
<th>30-39</th>
<th>40-49</th>
<th>50-59</th>
<th>&gt; 60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tuples</td>
<td>200</td>
<td>800</td>
<td>5000</td>
<td>12000</td>
<td>6500</td>
<td>500</td>
</tr>
</tbody>
</table>

\[ \text{Estimate} = 1200 \quad \text{Estimate} = 1 \times 80 + 5 \times 500 = 2580 \]
Types of Histograms

• How should we determine the bucket boundaries in a histogram?
Types of Histograms

• How should we determine the bucket boundaries in a histogram?

• Eq-Width
• Eq-Depth
• Compressed
• V-Optimal histograms
### Histograms

**Employee(ssn, name, age)**

**Eq-width:**

<table>
<thead>
<tr>
<th>Age</th>
<th>0..20</th>
<th>20..29</th>
<th>30-39</th>
<th>40-49</th>
<th>50-59</th>
<th>&gt; 60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tuples</td>
<td>200</td>
<td>800</td>
<td>5000</td>
<td>12000</td>
<td>6500</td>
<td>500</td>
</tr>
</tbody>
</table>

**Eq-depth:**

<table>
<thead>
<tr>
<th>Age</th>
<th>0..33</th>
<th>33..38</th>
<th>38-43</th>
<th>43-45</th>
<th>45-54</th>
<th>&gt; 54</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tuples</td>
<td>1800</td>
<td>2000</td>
<td>2100</td>
<td>2200</td>
<td>1900</td>
<td>1800</td>
</tr>
</tbody>
</table>

**Compressed:** store separately highly frequent values: (48,1900)
V-Optimal Histograms

- Defines bucket boundaries in an optimal way, to minimize the error over all point queries
- Computed rather expensively, using dynamic programming
- Modern databases systems use V-optimal histograms or some variations
Difficult Questions on Histograms

• Small number of buckets
  – Hundreds, or thousands, but not more
  – WHY? *All histograms are kept in main memory during query optimization; plus need fast access*

• *Not* updated during database update, but recomputed periodically
  – WHY? *Histogram update creates a write conflict; would dramatically slow down transaction throughput*

• Multidimensional histograms rarely used
  – WHY? *Too many possible multidimensional histograms, unclear which ones to choose.*
Multidimensional histograms

Stacked 2D histograms