Lecture 23:
Query Optimization (3)

Monday, November 22, 2010
Outline

• Search space

• Algorithms for enumerating query plans

• Estimating the cost of a query plan
Computing the Cost of a Plan

- Collect statistical summaries of stored data
- Estimate \textit{size} in a bottom-up fashion
- Estimate \textit{cost} by using the estimated size
Statistics on Base Data

• Collected information for each relation
  – Number of tuples (cardinality)
  – Indexes, number of keys in the index
  – Number of physical pages, clustering info
  – Statistical information on attributes
    • Min value, max value, number distinct values
    • Histograms
  – Correlations between columns (hard)

• Collection approach: periodic, using sampling
Size Estimation Problem

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

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

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

\[ S = \text{SELECT list} \]
\[ \text{FROM} \quad R_1, \ldots, R_n \]
\[ \text{WHERE} \quad \text{cond}_1 \text{ AND cond}_2 \text{ AND } \ldots \text{ AND cond}_k \]

Remark: \( T(S) \leq T(R_1) \times T(R_2) \times \ldots \times T(R_n) \)
Selectivity Factor

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

• Assuming independence, multiply the selectivity factors
Example

```
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

What is the estimated size of the query output?
Rule of Thumb

• If selectivities are unknown, then: selectivity factor = 1/10
  [System R, 1979]
Selectivities from Statistics

• Condition is $A = c$ /* value selection on $R$ */
  – Selectivity = $1/V(R,A)$

• Condition is $A < c$ /* range selection on $R$ */
  – Selectivity = $(c - \text{Low}(R, A))/(\text{High}(R, A) - \text{Low}(R, A))T(R)$

• Condition is $A = B$ /* $R \bowtie_{A=B} S$ */
  – Selectivity = $1 / \max(V(R,A), V(S,A))$
  – (will explain next)
Assumptions

• **Containment of values**: if $V(R,A) \leq V(S,B)$, then the set of $A$ values of $R$ is included in the set of $B$ values of $S$
  – 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 $V(S, C)$)
Selectivity of \( R \Join_{A=B} S \)

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

- mmmmmmmmhk,mmmbknmmmmmmmmmmmmktt
- Each tuple \( t \) in \( R \) joins with \( T(S)/V(S,B) \) tuple(s) in \( S \)

- Hence \( T(R \Join_{A=B} S) = T(R) \times T(S) / V(S,B) \)

In general: \( T(R \Join_{A=B} S) = T(R) \times T(S) / \max(V(R,A),V(S,B)) \)
Size Estimation for Join

Example:

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

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

Employee(ssn, name, age)

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

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

Employee(ssn, name, age)

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

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

Estimate = \( 25000 / 50 = 500 \)

Estimate = \( 25000 \ * \ 6 / 60 = 2500 \)
Histograms

Employee(ssn, name, age)

<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>

$T(\text{Employee}) = 25000$, $V(\text{Employee, age}) = 50$

$\min(\text{age}) = 19$, $\max(\text{age}) = 68$

$\sigma_{\text{age}=48}(\text{Employee}) = ?$

$\sigma_{\text{age}>28 \text{ and age}<35}(\text{Employee}) = ?$
## Histograms

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

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

\[ \sigma_{\text{age}=48}(\text{Employee}) = \text{?} \]

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

<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>

**Estimate = 1200**

**Estimate = 2*80 + 5*500 = 2660**
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
Employee($ssn$, name, age)

Histograms

**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..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>1800</td>
<td>2000</td>
<td>2100</td>
<td>2200</td>
<td>1900</td>
<td>1800</td>
</tr>
</tbody>
</table>

**Compressed:** store separately some highly frequent values: (48,1900)
Difficult Questions on Histograms

• Small number of buckets
  – Hundreds, or thousands, but not more
  – WHY?

• Not updated during database update, but recomputed periodically
  – WHY?

• Multidimensional histograms rarely used
  – WHY?
Summary of Query Optimization

• Three parts:
  – search space, algorithms, size/cost estimation

• Ideal goal: find optimal plan. But
  – Impossible to estimate accurately
  – Impossible to search the entire space

• Goal of today’s optimizers:
  – Avoid very bad plans