## Chapter 5.5 Introduction of Plan Selection Based on Cost

***Introduction:***

No matter choose the Logical Query Plan or construct the Physical Plan from the Logical Query Plan, the Query Optimizer needs to estimate the cost of the specific Expression.

Just like before, we use Executable Disk I/O to make estimation about the cost of Expression. However, Disk I/O influenced by factors below:

* Select the Realization Operator for Specific Logic Operator, this is what we choose when we choose the Logic Query Plan.
* The Size of Intermediate Relation.
* The Physical Operator that is used to Realize Logic Operator.
* Sort the Similar Operators.
* The Parameters that Passed from Physical Operator to the next Physical Operator.

Here, in order to make Plan effect, we need to solve many questions.

### Chapter 5.5.1 Get Estimation of Size Parameter

***Instruction:***

Before we can solve the issue, we need to know the size of tuples in Relation R and the V(R, a), which is the different number of the attribute a in Relation R. Nowadays, DBMS allows the normal users or managers to collect the statistical information, such as T(R) and V(R, a). These statistical value can be used in the latter Query Optimizer, the statistical number will be updated in the next round that get the collect command.

Through scan the whole Relation R, obviously we can get the count of tuples T(R), and find different number of attribute A, V(R, a). The block number B(R) in Relation R can be used by calculating the real block number or through T(R) divided by tuple numbers that one block can contain.

***Definition:***

* DBMS can calculate one ***Histogram*** by giving values on several attributes.
* If V(R, A) is not too big, then the Histogram will be consisted by each tuple number with attribute A.
* If the attributes exist a lot of different values, then only the most normal value will be recorded, while other value will be included as one group and be calculated together.

The normally used Histogram is called:

1. ***Equal Width:***

* Here, we choose Selected Width w and Normal Value v0. The tuple number v is provided, the range of v will be v0 <= v < v0 + w, v0 + w <= v < v0 + 2w, and so on.
* The value v0 will be the Least Possible Number or Known Lower Bound so far. If it is the latter one, then when meets one new smaller value, then we will decrease w on v0 and add into ***Histogram***.

1. ***Equal Height:***

* They have the common ‘percentage point’. We choose some small number p, and list all numbers that bigger than the least number for about p number, 2\*p, 3\*p and so on, until the final biggest number.

1. ***The Most Frequent Value:***

* We list the most common value and the times they appear. However, this message can be provided with all other values or other than the equal width and equal height, then we list all other values that appears normally.

1. ***Advantage:***

* By using Histogram, we can get the estimation of Join. Especially when Join Attributes appear in the Histogram, then we can get the exact number of tuples that have the value.

***Explanation:***

For those do not show Exact Value on Histogram, then we just need to use the Estimation Method to evaluate the result in Chapter 5.4.

If we use Equal Width Histogram to evaluate Exact Value after Join, then we just need to estimate the Join size on the same Width range and sum all these estimation. This result is the good estimation however and only when Join on the same Width Histogram.

The example in this Chapter only lists some useful aspect but does not use this for estimation later.

### Chapter 5.5.2 Calculation of Statistical Value

### Chapter 5.5.3 Heuristic Estimation of Decreasing Logical Query Plan

### Chapter 5.5.4 The Method to Enumerate Physical Project