# Chapter 4 Query Execution

***Introduction***

Query processor is one of the part collection, it can convert the user query and data command into the operation sequences on the database and operate operations. Since SQL enables us to query in very high level, then query processor must provide a bunch of details about how these queries are executed. A normal execution strategy may result in the query execution time much higher than the required time.



***Content Involved***

* Algorithm that operates on database data.
* Relation algebra operations: scan, hash, sort, and index are main methods.

*(These operations may make different assumptions. Some will assume that the main memory can hold all data; While others may assume that the operation objects will be too large to hold into the main memory.)*

***Query Compilation Overview***

There include three main steps:

1. *Analysis, construct Query Analysis tree.*
2. *Re – construct Query.* The Query Analysis tree will be initialized as the query plan, and such query plan is usually *Algebra Expression of Query*. Then initialization query plan will be converted as an equal plan which may cost less.
3. *Generate Physics Plan.* For query plan in (b), choose realization algorithm for every operator and execution sequences of these operators. *(Convert Logic Plan to the Physics Query Plan.)* The physics plan is using the expression tree to represent.

b) and c) are seen as Query Optimizer and they are the key point of query compilation.

In order to choose the best query project, we need to make sure:

1. Which algebra equivalent form of query can have the most efficient algorithm for answering query?
2. For each operation, using which algorithm to realize?
3. How to convert one operation to another operation, such as assembly line, main memory buffer or disk?

*(Every choice depends on the metadata of database. The classical metadata includes: the size of relation, statistical data, indexes, and the data disk distribution.)*