# Chapter 7 Concurrency Control

*Background:*

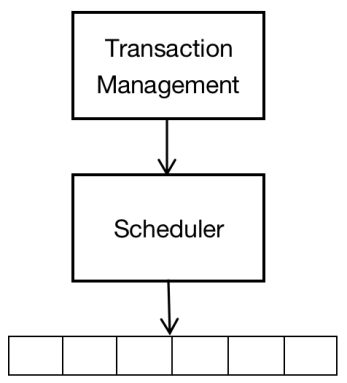
The Transactions that are executed concurrently may influence each other and cause the different status for Database System, even each Transaction can keep the correctness of its status, and without any failure happening.

*Definition - Norm | Concurrency Control:*

Each execution steps of different Transactions need to have some norms. *Norm* is finished by *DBMS Scheduler*, and the whole process that keeps Transactions consistent is called *Concurrency Control*.

*Process:*

When Transactions ask Write and Read for Database Elements, these Requests would be passed to Scheduler. In most situations, Scheduler would execute read and write directly, however if Database Elements are not in Buffer Area, then Buffer Management would be called and read Database Elements into the Buffer Area.



*Scheduler receives Read and Write commands from Transaction Management. These commands would be executed in the Buffer Area or be put off.*

*Drawback:*

But under some situations, executing Transactions directly is not safe. Scheduler needs to push back the Request Execution. Also in some Currency Control technologies, the scheduler may abort the request.

*Outline:*

In this Chapter, we would discuss how to ensure that the concurrent transactions would keep the Database System in the correctness status. The abstract requirement is Serializable, another more powerful and important condition is called Conflict Serializable, which is the most Scheduler really do. We need to consider the most important technologies: including *Locking*, *Time - Stamp*, and *Effectiveness Confirmation*. Also we includes *‘Two - Phase Locking’* technology.