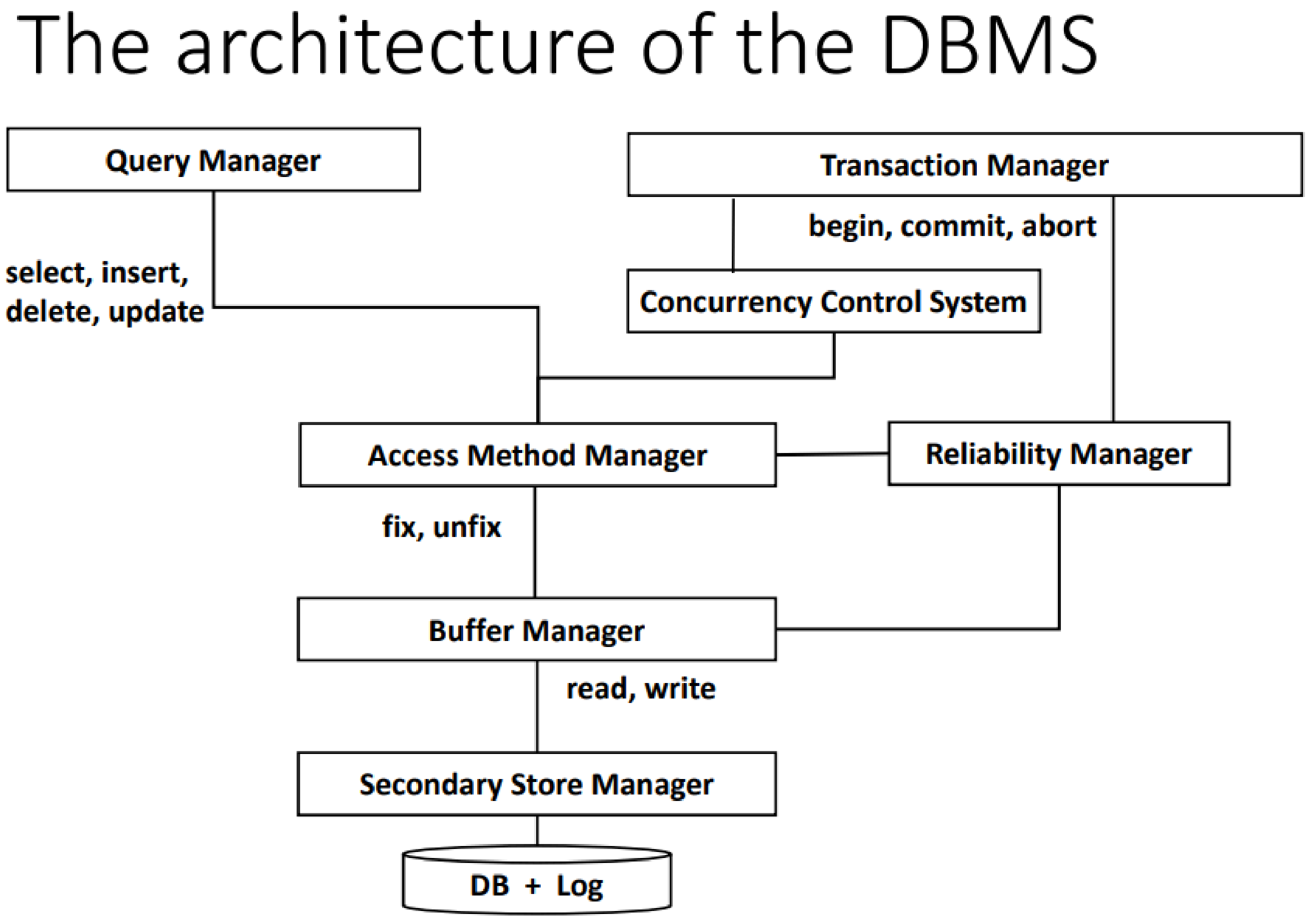
# 01.Transactional Systems

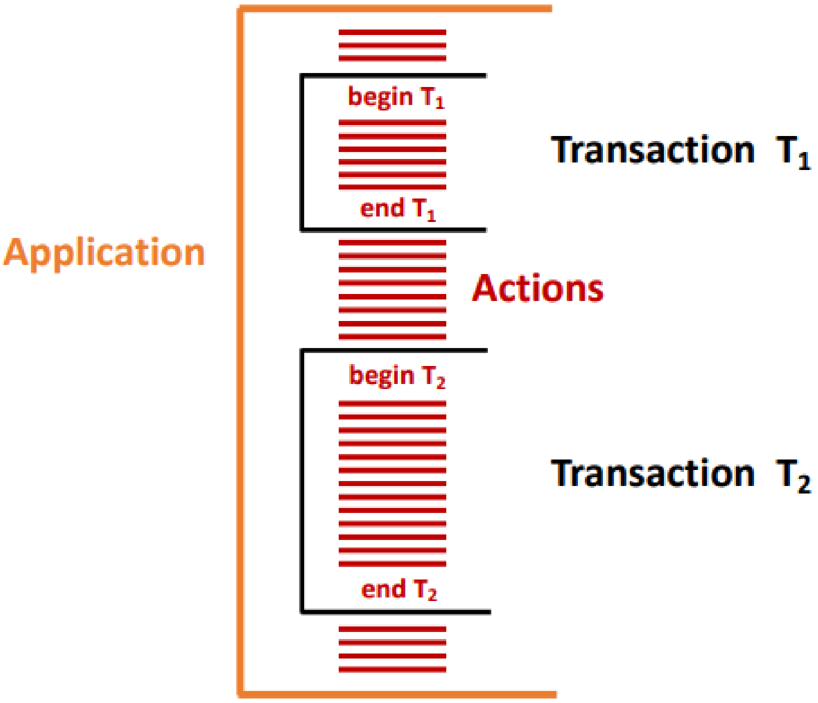


The architecture of the DBMS

A transaction is an elementary, atomic unit of work performed by an application
Each transaction is conceptually encapsulated within two commands:

- begin transaction
- end transaction
  Within a transaction, one of the commands below is executed (exactly once) to signal the end of the transaction:
- commit-work (commit)
- rollback-work (abort)
  A Transactional System (OnLineTransactionProcessing) is a system that supports the execution of transactions on behalf of concurrent applications

# Difference between Application and Transaction



## ACID Properties of Transactions

A transaction is a unit of work enjoying the following properties:

- Atomicity: a transaction is an indivisible unit of execution: either all the operations in the transaction are executed or none is executed. The time in which COMMIT is executed marks the instant in which the transaction ends successfully: an error before the end of the transaction should cause the rollback of the work, an error after the end should not alter the effect of the transaction. The ROLLBACK of the work performed can be caused: by the application with a ROLLBACK statement or by the DBMS, for example for the violation of integrity constraints or for concurrency management. In case of a rollback, the work performed must be undone, bringing the database to the state it had before the start of the transaction. It is the application's responsibility to decide whether an aborted transaction must be redone or not.
- Consistency: a transaction must satisfy the DB integrity constraints: if the initial state $S_0$ is consistent then the final state $S_f$ is also consistent but this is not necessarily true for the intermediate state $S_i$.
- Isolation: the execution of a transaction must be independent of the concurrent execution of other transactions: in particular, the concurrent execution of a number of transactions must produce the same result as the execution of the same transactions in a sequence.

Isolation impacts performance and trade-offs can be defined between isolation and performance

- Durability: the effect of a transaction that has successfully committed will last "forever independently of any system fault and ,sounds obvious, but every DBMS manipulates data in main memory

# ACID Properties and DBMS Mechanisms

- Atomicity
  - Abort-rollback-restart, commit protocols
  - Reliability Manager
- Consistency
  - Integrity checking of the DBMS
  - Integrity Control System at query execution time (with the support of the DDL compiler)
- Isolation
  - Concurrency control
  - Concurrency Control System
- Durability
  - Recovery management
  - Reliability Manager