

# TRANSACTION

A transaction is a unit of program execution that accesses and possibly updates various data items.

Or we can simply say that, transaction is a sequence of read and write operations on a certain database. That is, transfer the data item from a database to a local buffer and transfer the data item from the local buffer to the database.

Transactions provide an "all-or-nothing" proposition stating that work units performed in a database must be completed in their entirety or take no effect whatsoever. Further, transactions must be isolated from other transactions, results must conform to existing constraints in the database and transactions that complete successfully must be committed to durable storage.

The properties of database transactions are summed up with the acronym ACID:

- 1 Atomicity - all or nothing

All of the tasks (usually SQL requests) of a database transaction must be completed;

If incomplete due to any possible reasons, the database transaction must be aborted.

- 1 Consistency - serializability and integrity

The database must be in a consistent or legal state before and after the database transaction. It means that a database transaction must not break the database integrity constraints.

- 1 Isolation

Data used during the execution of a database transaction must not be used by another database transaction until the execution is completed. Therefore, the partial results of an incomplete transaction must not be usable for other transactions until the transaction is successfully committed. It also means that the execution of a transaction is not affected by the database operations of other concurrent transactions.

- 1 Durability

All the database modifications of a transaction will be made permanent even if a system failure occurs after the transaction has been completed.

Theoretically, a database management system (DBMS) guarantees all the ACID properties for each database transaction. In reality, these ACID properties are frequently more or less reduced to improve performance.

The purpose of obeying these rules is to provide reliable units of work that allow correct recovery from failures and keep a database consistent even in case of system failure, and provide atomic and isolating process so that our data and database is correct when the operation accidentally erupts.

And in summary, I think this part of database knowledge is more helpful when we want to understand the constraints of database operation and the design of database management process.

Source: <http://toyhouse.cc/profiles/blogs/transaction>