# What Does Transaction Mean?

A transaction, in the context of a database, is a logical unit that is independently executed for data retrieval or updates. Experts talk about a database transaction as a “unit of work” that is achieved within a database design environment.

In relational databases, database transactions must be atomic, consistent, isolated and durable—summarized as the ACID acronym. Engineers have to look at the build and use of a database system to figure out whether it supports the ACID model or not. Then, as newer kinds of database systems have emerged, the question of how to handle transactions becomes more complex.

Techopedia Explains Transaction

In traditional relational database design, transactions are completed by COMMIT or ROLLBACK SQL statements, which indicate a transaction’s beginning or end. The ACID acronym defines the properties of a database transaction, as follows:

* **Atomicity:** A transaction must be fully complete, saved (committed) or completely undone (rolled back). A sale in a retail store database illustrates a scenario which explains atomicity, e.g., the sale consists of an inventory reduction and a record of incoming cash. Both either happen together or do not happen—it's all or nothing.
* **Consistency:** The transaction must be fully compliant with the state of the database as it was prior to the transaction. In other words, the transaction cannot break the database’s constraints. For example, if a database table’s Phone Number column can only contain numerals, then consistency dictates that any transaction attempting to enter an alphabetical letter may not commit.
* **Isolation:** Transaction data must not be available to other transactions until the original transaction is committed or rolled back.
* **Durability:** Transaction data changes must be available, even in the event of database failure.

Transactions and Terminology

For reference, one of the easiest ways to describe a database transaction is that it is any change in a database, any “transaction” between the database components and the data fields that they contain.

However, the terminology becomes confusing, because in enterprise as a whole, people are so used to referring to financial transactions as simply “transactions.” That sets up a central conflict in tech-speak versus the terminology of the average person.

A database “transaction” is any change that happens. To talk about handling financial transactions in database environments, the word “financial” should be used explicitly. Otherwise, confusion can easily crop up. Database systems will need specific features, such as PCI compliance features, in order to handle financial transactions specifically.

As databases have evolved, transaction handling systems have also evolved. A new kind of database called NoSQL is one that does not depend on the traditional relational database data relationships to operate.

While many NoSQL systems offer ACID compliance, others utilize processes like snapshot isolation or may sacrifice some consistency for other goals. Experts sometimes talk about a trade-off between consistency and availability, or similar scenarios where consistently may be treated differently by modern database environments. This type of question is changing how stakeholders look at database systems, beyond the traditional relational database paradigms.