DEliVERiNGSKiLLS
DRiViNGSUCCESS

www.VINSYS.COM

ACID properties

When a transaction processing system creates a transaction, it will ensure that the transaction will have certain characteristics.

The developers of the components that comprise the transaction are assured that these characteristics are in place.

They do not need to manage these characteristics themselves.

These characteristics are known as the ACID properties.

ACID is an acronym for atomicity, consistency, isolation, and durability.

Atomicity

- The atomicity property identifies that the transaction is atomic. An **atomic transaction** is either fully completed, or is not begun at all. Any updates that a transaction might affect on a system are completed in their entirety. If for any reason an error occurs and the transaction is unable to complete all of its steps, the then system is returned to the state it was in before the transaction was started. An example of an atomic transaction is an account transfer transaction. The money is removed from account A then placed into account B. If the system fails after removing the money from account A, then the transaction processing system will put the money back into account A, thus returning the system to its original state. This is known as a **rollback**

Consistency

- A transaction enforces **consistency** in the system state by ensuring that at the end of any transaction the system is in a valid state. If the transaction completes successfully, then all changes to the system will have been properly made, and the system will be in a valid state. If any error occurs in a transaction, then any changes already made will be automatically rolled back. This will return the system to its state before the transaction was started. Since the system was in a consistent state when the transaction was started, it will once again be in a consistent state.
- Looking again at the account transfer system, the system is consistent if the total of all accounts is constant. If an error occurs and the money is removed from account A and not added to account B, then the total in all accounts would have changed. The system would no longer be consistent. By rolling back the removal from account A, the total will again be what it should be, and the system back in a consistent state.

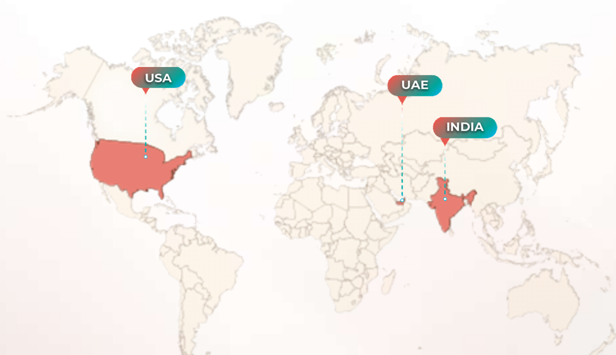
Isolation

- When a transaction runs in **isolation**, it appears to be the only action that the system is carrying out at one time. If there are two transactions that are both performing the same function and are running at the same time, transaction isolation will ensure that each transaction thinks it has exclusive use of the system. This is important in that as the transaction is being executed, the state of the system may not be consistent. The transaction ensures that the system remains consistent after the transaction ends, but during an individual transaction, this may not be the case. If a transaction was not running in isolation, it could access data from the system that may not be consistent. By providing transaction isolation, this is prevented from happening.

Durability

- A transaction is **durable** in that once it has been successfully completed, all of the changes it made to the system are permanent. There are safeguards that will prevent the loss of information, even in the case of system failure. By logging the steps that the transaction performs, the state of the system can be recreated even if the hardware itself has failed. The concept of durability allows the developer to know that a completed transaction is a permanent part of the system, regardless of what happens to the system later on.

Physical Presence



www.vinsys.com

enquiry@vinsys.com

Shivaji Niketan, Tejas Society, Behind Kothrud Bus Stand, Near Mantri Park, Kothrud, Pune - 411029.

304, City Tower 2, Near Crown Plaza, Sheikh Zayed Road. Dubai, UAE. PO.Box - 213279

132 West 31st Street, First Floor, New York, 10001, USA