Isolation Levels

About Read Uncommitted, Read Committed, and Repeatable read -- these are different isolation levels. Also do Serializable.

READ UNCOMMITTED  
Specifies that statements can read rows that have been modified by other transactions but not yet committed. This is a dirty read (JL).

Transactions running at the READ UNCOMMITTED level do not issue shared locks to prevent other transactions from modifying data read by the current transaction. READ UNCOMMITTED transactions are also not blocked by exclusive locks that would prevent the current transaction from reading rows that have been modified but not committed by other transactions. When this option is set, it is possible to read uncommitted modifications, which are called dirty reads.

-----------------------------------------------------------------------------------------------------

READ COMMITTED  
Specifies that statements cannot read data that has been modified but not committed by other transactions. This prevents dirty reads. Data can be changed by other transactions between individual statements within the current transaction, resulting in nonrepeatable reads. This option is the SQL Server default.

---------------------------------------------------------------------------------------------------------

REPEATABLE READ  
Specifies that statements cannot read data that has been modified but not yet committed by other transactions(no dirty read allowed) and that **no other transactions can modify data** that has been read by the current transaction until the current transaction completes.

Shared locks are placed on all data read by each statement in the transaction and are held until the transaction completes. This prevents other transactions from modifying any rows that have been read by the current transaction. **Other transactions can insert new rows** that match the search conditions of statements issued by the current transaction. **If the current transaction then retries the statement it will retrieve the new rows.**

Can have many statements inside of one transaction (Begin Transaction). (JL).

-----------------------------------------------------------------------------------------------------------

SERIALIZABLE   
Specifies the following:

* Statements cannot read data that has been modified but not yet committed by other transactions. No dirty Reads allowed.
* No other transactions can modify data that has been read by the current transaction until the current transaction completes.
* Other transactions cannot insert new rows with key values that would fall in the range of keys read by any statements in the current transaction until the current transaction completes.

Range locks are placed in the range of key values that match the search conditions of each statement executed in a transaction. This blocks other transactions from updating or inserting any rows that would qualify for any of the statements executed by the current transaction. **This means that if any of the statements in a transaction are executed a second time, they will read the same set of rows**. The range locks are held until the transaction completes. This is the most restrictive of the isolation levels because it locks entire ranges of keys and holds the locks until the transaction completes. Because concurrency is lower, use this option only when necessary.

**So, this does Not allow REPEATABLE READ**

## Remarks

Only one of the isolation level options can be set at a time, and it remains set for that connection until it is explicitly changed.

The transaction isolation levels define the type of locks acquired on read operations. Shared locks acquired for READ COMMITTED or REPEATABLE READ are generally row locks, although the row locks can be escalated to page or table locks if a significant number of the rows in a page or table are referenced by the read.

If a row is modified by the transaction after it has been read, the transaction acquires an exclusive lock to protect that row, and the exclusive lock is retained until the transaction completes. For example, if a REPEATABLE READ transaction has a shared lock on a row, and the transaction then modifies the row, the shared row lock is converted to an exclusive row lock.

-----------------------------------------------------------------------------------------------------------

## Examples

The following example sets the TRANSACTION ISOLATION LEVEL **for the session**. For each Transact-SQL statement that follows, SQL Server holds all of the shared locks until the end of the transaction.

SQLCopy

USE AdventureWorks2012;

GO

SET TRANSACTION ISOLATION LEVEL REPEATABLE READ;

GO

BEGIN TRANSACTION;

GO

SELECT \*

FROM HumanResources.EmployeePayHistory;

GO

SELECT \*

FROM HumanResources.Department;

GO

COMMIT TRANSACTION;

GO