



Transaction Log Architecture and Working



Transaction Log Architecture

- The SQL Server transaction log is a single file which usually has an .LDF file extension.
- Although possible to have multiple log files for a database, the transaction log is always written sequentially and multiple physical log files get treated as one continuous circular file.
- SQL Server uses the transaction log to ensure that all transactions maintain their state even in case of a server or database failure.
- All transactions are written to the Transaction Log before it is written to the data files. This is known as write ahead logging.
- Every action performed on SQL Server is logged in the SQL Server transaction log, multiple entries may be created for a transaction as well as all locks that were taken during the operation.
- Each log entry has a unique number known as the LSN (log sequence number).
- Enough information is written to the log to allow for a transaction to be either re-done (rolled forward) or undone (rolled back).



Transaction Log Architecture (contd..)

- Logically the SQL Server Transaction log is divided into multiple sections known as virtual log files or VLFs.
- The logical transaction log gets truncated and expanded in units of VLFs.
- If a VLF no longer contains an active transaction, that VLF can be marked for re-use.
- If the log needs more space, space is allocated in increments of VLFs.
- The number and size of the VLFs is decided by the database engine and it will endeavor to assign as few VLFs as possible.
- Although the size and number of VLFs cannot be configured, it is affected by the initial size and the growth increment of the transaction log.
- If the log growth increment is set too low, it may result in an excessive amount of VLFs which can have an adverse effect on performance.
- In order to avoid this, it is important to size the log correctly and grow it in sufficiently large increments.
- DBCC LOGINFO

Operation And Working of Transaction Logs

