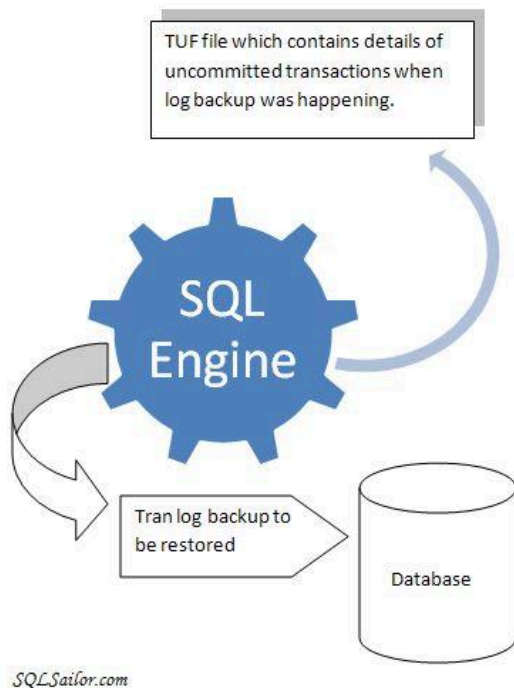


TUF Files in SQL Server LogShipping



Shared by [Chandrasekhar Kuntimaddi](#)



bachamps.com

In SQL Server, log shipping is a robust method for maintaining a backup of your database on a secondary server.

A critical component of this process is the Transaction Undo File (TUF), which plays a significant role in ensuring data integrity and consistency across your primary and secondary databases.

TUF File: A transaction undo file created during log shipping in standby mode.

Uncommitted Transactions: Stores modifications not yet committed on the source database.

Location: Found in the same location as TRN files and secondary database LDF files.

Data Protection: Ensures users on the secondary database cannot read uncommitted data.

Log Backup Process: Committed transactions are written to the secondary database; uncommitted transactions remain in TUF.

Impact of Missing TUF: Absence of the TUF file can prevent the secondary database from being operational.

Understanding TUF Files: The TUF file plays a critical role in log shipping, capturing uncommitted transactions, and ensuring data integrity on the secondary database.

Transaction Management: TUF files prevent users from accessing potentially unstable data by holding uncommitted transactions until they are resolved.

Backup Restoration: The process of restoring transaction logs involves checking TUF files to determine whether transactions should be committed or rolled back.

File Location: Knowing where TUF files are located helps in database management and troubleshooting, as they reside alongside transaction log and data files.

Operational Risks: If a TUF file is missing, it can lead to a scenario where the secondary database cannot be brought online, emphasizing the need for proper file management.

Read-Only Access: In standby mode, TUF files facilitate read-only access for users while ensuring data consistency on the secondary server.

Understanding how TUF files work, where they are located, and how to manage them is essential for any database administrator involved in log shipping and disaster recovery planning.

Proper handling of TUF files will ensure that your secondary databases remain reliable and operational, safeguarding against data loss and maintaining consistency across your SQL Server environments.