

# How to Deal with an Oversize Transaction Log

## What is a transaction log and why can it get so big?

When you add, edit or delete a record in a SQL Server database the change is written first to the transaction log. A background process subsequently writes each transaction in the log to the database then marks the transaction in the log to show it has been “written”.

If the database is set to use the **Simple** Recovery Model, transactions are removed from the log once they have been written to the database. This makes for low maintenance transaction logs but limits how much data you can recover if you restore a backup. If you encounter a problem late in the working day and have to restore the last database backup from the night before, you lose all of today’s work.

Alternatively, if the database is set to the **Full** recovery model, “written” transactions are retained in the log. Using this model, a SQL Server Database Restore can restore the last database backup (e.g. from the night before) but then replay all the transactions that have been written to the database since (as records of these exist in the log). This means you could recover all of today’s work too or all work up to the point the problem occurred.

However, if you use the Full recovery model you do need to manage the transaction log to prevent it growing unchecked. Otherwise it will eventually use all the available disk space or severely affect performance.

The usual way to manage the transaction log is to make regular transaction log backups (we suggest these be made hourly). A transaction log backup copies the transactions from the live log into a backup file then truncates the log by removing all the written transactions. SQL Server can restore the last database backup and then replay all transactions written to the database since the backup from the records in the transaction log backups as well as in the live log. This means you can recover all of today’s data but keep the live transaction log small and manageable.

Strictly speaking, you only need to retain the transaction log backups since the last database backup but it is good practice to retain two or three days worth of transaction log backups in case there is a problem with your last database backup. (All backups retained on disk should be backed up to another location e.g. tape every night so you do not lose you backups if the hard disk is damaged).

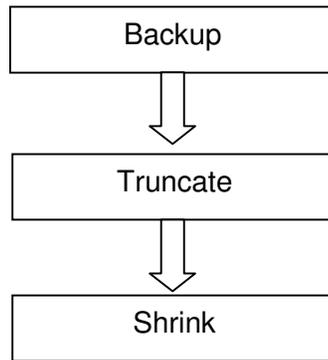
If transaction logs backups fail (or are never scheduled) the log will grow unchecked until it uses up all available disk space or severely impacts performance. Common causes of transaction log backup failure are:

- Backups were never scheduled
- SQL Server Agent (the service that runs scheduled SQL jobs) is not running. It may have been set to Manual start up mode which means it does not start automatically if the server is restarted. The correct start up mode is Automatic.
- Old transaction log backup files are not being removed from the disk causing the disk to fill up to the point where there is no longer enough room to make backups.

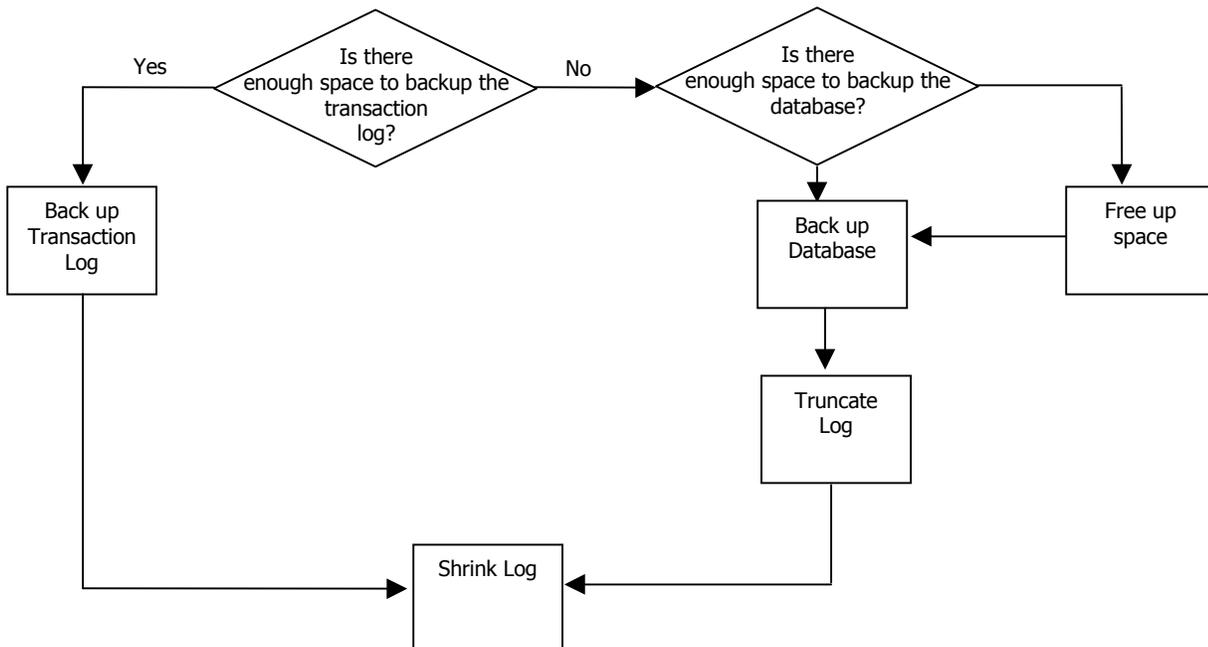
## What to do if you have an Oversized Transaction Log

If your transaction log backups are failing you must identify the cause and resolve the problem. However, if they have been failing for a while you may also be left with an oversized transaction log. The section describes how to truncate the transaction log and shrink the file back to an appropriate size.

The process comprises three steps:



The process can be summarised by the following flow chart



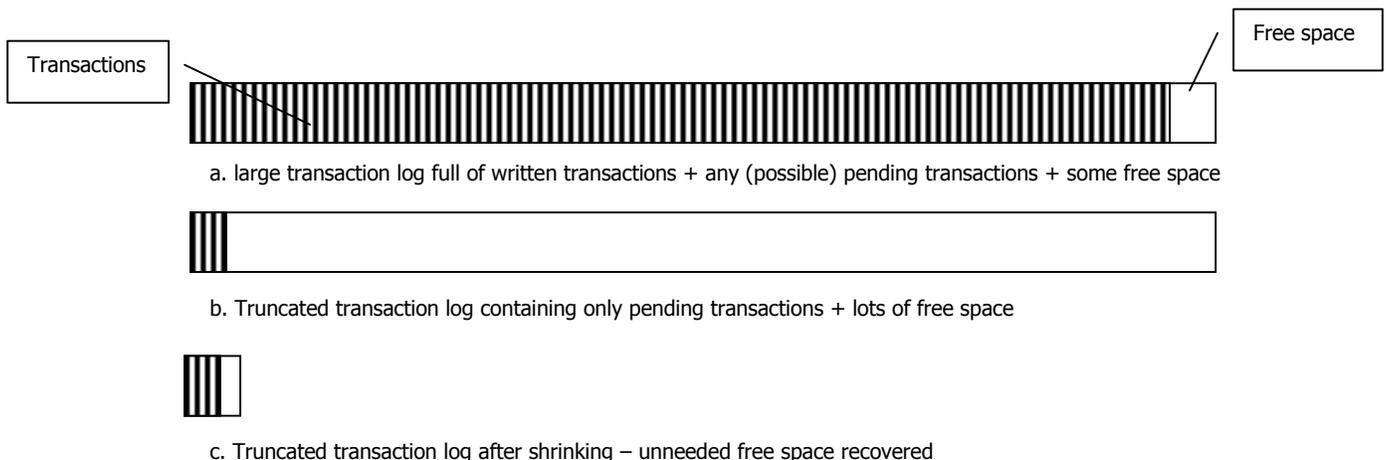
## Step by Step Guide

1. Check the size of your transaction log either by looking at the file in Windows Explorer (it is the file with the .LDF extension) or right click the database in Enterprise Manager/Management Studio/Dbamgr2k choose Properties / Files / Transaction Log.
2. Check your available disk space.
3. If the transaction log file is smaller than the amount of available disk space, perform a transaction log backup. If the transaction log backups have been scheduled you can locate the job under Management/SQL Server Agents/ Current Activity, right click and choose Start Now. If transaction log backups have not been scheduled, right click on the database choose Tasks/Backup Database and choose to backup the transaction log.
4. If there is insufficient space to make a transaction log backup, check the size of the database (.mdf) file. If this is smaller than the amount of free disk space make a full database backup.
5. If there is not space to make a full database backup, see if you can free up enough space and then make a full backup. Only if there is no other option should you proceed without any backup and you should ensure the customer accepts the risk of doing so. The risk being that should anything go wrong they would lose all work since their last good database backup.
6. If you have made a transaction log backup, the log has already been truncated. If you did not have room to make a transaction log backup and have made a database backup, you need to truncate the transaction log. In other words remove the transactions that have been written to the database. You can do this (without creating a transaction log backup) by running the following query from within Enterprise Manager/ Management Studio or Dbamgr2k:

### **BACKUP LOG DawnAC WITH TRUNCATE\_ONLY**

...where DawnAC is your Dawn database name.

Running this query truncates the log (removes the transactions) but does not reduce the size of the file. You simply have a large file consisting of mainly free space. To recover the space you must shrink the transaction log.



7. Shrink the database by right clicking the database and choosing the Shrink Database or Tasks/Shrink Database option in Enterprise Manager/Management Studio or DbAMgr2k. Where your management utility includes a Shrink Files option (Enterprise Manager, Management Studio), choose this option and select the Log as the file to shrink.

If truncating and shrinking do not reduce the size of your transaction log, perform steps 6 and 7 again. (SQL server goes through several steps when marking pages in the log file as available for recovery – sometimes you need to repeat the process to get them to point where they can be freed).