

Example of what happens with PAGELATCH* contention in tempdb. With one file initially, all allocations occur from one file. When multiple files are created, allocations round-robin between the files. See <http://support2.microsoft.com/kb/2154845>

After initially creating a database, create a second filegroup (named HR in this example) with one to many files and put user objects in this filegroup by including ON FGNAME in CREATE syntax. In addition, set a filegroup other than the PRIMARY as the default, so that if anyone forgets to include ON FGNAME, the object won't accidentally be created in the PRIMARY filegroup.

Example of a database with the PRIMARY filegroup, in which system objects are located (by default, this cannot be changed). There is one log file, and then another filegroup, FG1, with two files which holds user data and has been made the DEFAULT filegroup.

In a database with FULL recovery model, transactions are written to the transaction log and then backed up with a transaction log backup (typically a .trn file). Once transactions are backed up, VLFs are cleared, assuming the transactions have been committed are no longer needed.

Example schedule of FULL (Sunday) and DIFFERENTIAL (daily at 10PM) backups, with an ad-hoc FULL backup taken at 5PM on Wednesday. If it is **not** taken as a COPY ONLY backup, and it is discarded, the backup chain has been broken, and you can only restore the Sunday FULL and then Tuesday's DIFFERENTIAL (and any log backups taken before 5PM).

Example schedule of FULL (10PM) and LOG (every 2 hours) backups, with an ad-hoc transaction log backup taken for a developer at 11 AM. If it is **not** taken as a COPY ONLY backup, and it is discarded, the log backup chain has been broken, and you can only restore the 10PM FULL and then log backups through 10AM, assuming a failure at 3PM. If it **is** taken as a COPY ONLY backup, and there is a failure at 3PM, you can restore all the way to the 2PM transaction log backup.

Example schedule of FULL (10PM) and LOG (every 1 hour) backups. If there is a drive failure at 8:30 AM, and the database becomes inaccessible, as long as the instance is still running you can take a tail-log backup and try to recover up to the point of the drive failure. Remember to use the TSQL syntax for the backup, which includes NO_TRUNCATE and also RECOVERY. With this option, the transactions in the log are not truncated, and the database is **not** left in the RESTORING state. Using NO RECOVERY instead **would** put the database into a RESTORING state.

Example of how, after creating two indexes (F, P – figure 1), and then deleting one object (figure 2), when a shrink is performed, the P index is perfectly fragmented as pages are moved from the very end of the file to the front (figure 3).

Root page →

Intermediate level →

← Leaf level

Example of what a b-tree index looks like and how the pages are linked (leaf level is a doubly-linked list). See also <http://technet.microsoft.com/en-us/library/ms177443%28v=sql.105%29.aspx> for clustered indexes and <http://technet.microsoft.com/en-us/library/ms177484%28v=sql.105%29.aspx> for nonclustered indexes. Reference <http://technet.microsoft.com/en-us/library/ms188270%28v=sql.105%29.aspx> for heaps.