**<http://sqlmag.com/blog/when-do-checkpoints-occur-tempdb>**

**When do checkpoints occur for tempdb?**

*Mar 22, 2011* by [Paul S. Randal](http://sqlmag.com/blogger/paul-s-randal) in [SQL Server Questions Answered](http://sqlmag.com/blog/sql-server-questions-answered)

[RSS](http://sqlmag.com/blog/10118/feed)

**Question:** (From the #sqlhelp Twitter hash-tag this afternoon, paraphrasing) I’ve just started monitoring the health of a number of SQL Server instances and I’m seeing some strange behavior with tempdb. One of the counters I’m monitoring is the Percent Log Used in the Databases perfmon object. For normal databases it fluctuates but for tempdb on one server it seems to just increase and never drop. What’s going on?

**Answer:** The answer is that log management is different for tempdb than for all other databases.

For those who don’t know, a checkpoint operation occurs regularly in all databases except tempdb. A checkpoint has two purposes: firstly to write out data pages so that what is in the data files is up-to-date with what is in the log, so there is less REDO recovery to perform if a crash occurs; secondly to reduce the amount of writes to the data files by only writing out data file pages periodically instead of after every change to a page.

A checkpoint is triggered automatically under a variety of conditions. The most common condition is that enough transaction log has been generated that SQL Server estimates that if the server was to crash, it would take about one minute for crash recovery to complete. This is calculated based on the number of log records that have been generated since the last checkpoint and is known as the *recovery interval*. The next most common condition that triggers a checkpoint is when the log becomes 70% full.

When a checkpoint occurs for a database in the SIMPLE recovery model, log clearing (a.k.a. log truncation) occurs. This means that some portions of the transaction log may be able to be marked as reusable if nothing else requires them to be kept around, thus decreasing the amount of space used in the log. (I’m vastly over-simplifying – see my article [here](http://technet.microsoft.com/en-us/magazine/2009.02.logging.aspx) for more details.)

If you watch the Percent Log Used counter in Performance Monitor for a database in the SIMPLE recovery model you should see it trace a saw-tooth pattern as the log space is reduced by the checkpoint and then builds back up again until the next checkpoint.

When a server crash occurs, tempdb is automatically recreated – it is never crash recovered (hence its name). This means that the log management system for tempdb does not trigger a checkpoint based on the recovery interval but instead uses the 70% threshold.

If you watch the Percent Log Used counter in Performance Monitor for tempdb you will see it increase until it reaches 70% and then a checkpoint should reduce it, and it will slowly build back up again.

Another checkpoint behavior that is different for tempdb is that changed data file pages are not written to disk when an automatic tempdb checkpoint occurs. There is no reason for them to be written to disk as, again, there is no crash recovery for tempdb and no reason to provide on-disk durability of committed operations for tempdb.