Turbo-Charged Transaction Logs

David Maxwell

Thank You

Presenting Sponsors

Microsoft

Gain insights through familiar tools while balancing monitoring and managing user created content across structured and unstructured sources.

www.microsoft.com

CISCO

Unifying computer, storage, networking, and virtualization, Cisco UCS is the optimal database and business intelligence platform for SQL Server.

www.cisco.com

DELL Software

Solutions from Dell help you monitor, manage, protect and improve your SQL Server environment.

www.software.dell.com

Supporting Sponsors

SQL SENTRY®

hp

solarwinds

PASS SUMMIT 2015

October 27-30 / Seattle, WA

Planning on attending PASS Summit 2015? Start saving today!

- The world's largest gathering of SQL Server & BI professionals
- Take your SQL Server skills to the next level by learning from the world's SQL Server experts, in over 190 technical sessions
- Over 5000 attendees, representing 2000 companies, from 52 countries, ready to network & learn

Contact your Local or Virtual Chapter for an additional $150 discount.

$1795

until July 12th, 2015

PASS

Speaker Bio

David Maxwell

Production Support DBA

Nationwide Children's Hospital, Columbus Ohio

20 Years in IT

15 Years supporting and maintaining SQL Server from versions 6.5 – 2014.

dmmaxwell@gmail.com

@dmmaxwell

www.dmmaxwell.wordpress.com

What we will cover:

What is *really* in the transaction log?

How does the log manage transactions?

Potential problems and solutions.

Performance tuning the log.

5

What's really in the transaction log?

How SQL Server logs changes in the database:

- The 8k page to be changed is read into buffer pool memory.
- A record of the change is written to the transaction log.
  - Space is reserved in case of rollback.
- The page in memory is modified.
- If the transaction is committed:
  - The *Commit record* is written to the log.
- If the transaction is killed or rolled back:
  - A *Compensation record* is written to the log.
  - The page in memory is changed back to its original values.

A word about CHECKPOINT

CHECKPOINT writes changed (dirty) pages in memory to disk.

- Happens independently of the state of the transaction. Unfinished transactions can still be written to disk.
- By default, CHECKPOINTs occur once every minute, and whenever a backup is started.
- CHECKPOINT interval can be modified at the server or database level.
  - Server Level Setting is *Recovery Interval*. Measured in minutes.
  - Database Level Setting is *Target Recovery Time*. Measured in seconds.

DEMO: Reading the transaction log with fn_dblog().

How is the log managed?

Virtual Log Files, or VLFs

- The number of VLFs depends on the create or grow size of the log.

CREATE OR GROWTH SIZE	NUMBER OF VLFs CREATED
Less than 64 MB	4 VLFs at 1/4 of the size
From 64 MB to 1 GB	8 VLFs at 1/8 of the size
Greater than 1 GB	16 VLFs at 1/16 of the size

- Under SQL 2014, things change. Growths less than 1/8 of the current size use only one VLF.
- Example: 1 GB log file, with 100 MB growth
  - For SQL Server 2012 and lower, there will be 8 VLFs at 12.5 MB each.
  - For SQL Server 2014, there will be 1 VLF at 100 MB.

VLF Re-Use and Recovery Models

VLF behavior depends on the recovery model of the database.

- There must always be at least 1 active VLF.
- A VLF becomes active when an open transaction writes a log record to it.
- VLFs become available for re-use (inactive) when transactions commit or roll back AND…
  - In the SIMPLE recovery model, a CHECKPOINT has occurred.
  - In the FULL or BULK_LOGGED recovery model, a log backup must be taken.
- The transaction log is circular, meaning VLFs at the beginning of the log file can be re-used.
- If no VLFs are available for use, and a transaction starts, the log must grow.

DEMO: Examining VLF re-use with DBCC LOGINFO and fn_dblog().

Troubleshooting the Log

Scenario 1: Running out of space.

- Causes:
  - Long-running transactions.
  - Log backups not keeping up or being done at all.
- Solutions:
  - Increase log backup frequency.
  - Kill open transactions.
  - Add another log file.
- Caveats:
  - Rollback can take time.
  - Determine what transactions are doing before killing.
  - Remember to remove additional log files when finished.

Troubleshooting the Log

Scenario 2: VLF Fragmentation

- Cause:
  - Improper growth settings.
- Solution:
  - Determine correct growth settings.
  - Shrink the log file, and pre-size it appropriately.
- Caveats:
  - Watch out for “rubber-band” logs.
  - Transactions are delayed during growth.

DEMO: Identifying large transactions

DEMO: Cleaning up VLF fragmentation.

Improving Log Performance

Improve Overall IO

- Faster Disk Subsystems
- File Fragmentation (Improvement can be limited.)
- Isolate active log files to their own IO path.
  - Multiple log files on a single drive = random IO, when sequential is desired.

Pre-Allocate Log Space

- Monitor VLF counts.
- Auto-Grow is a good fail-safe, but shouldn't be relied on.

Do Less Work

DEMO: How VLF counts affect recovery time.

DEMO: How index fill factor affects log performance.