



## Strategic Sponsor

Microsoft

## Gold Sponsors

PYRAMID ANALYTICS

KRAJOWY REJESTR DŁUGÓW www.krd.pl

devart Database Interaction Solutions

## Silver Sponsors

Profisee®

SQL EXPERT.pl

CREDIT SUISSE

## Technical Partners

Microsoft

PTI POLSKIE TOWARZYSTWO INFORMATYCZNE ODDZIAŁ GÓRNOŚLĄSKI

## Academic Partners

Wyższa Szkoła Zarządzania i Bankowości w Krakowie

Wyższe Szkoły Bankowe

## Media Partners

GEEKCLUB.pl wss.pl

ERP24.pl

4programmers.net

IT professional

IN FACT PR

itnews24

virtual-it.pl

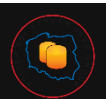
IT WORKS POLSKA

VirtualStudy

MAGAZYN IT PL

Made in Wro Do IT here!

ERP-view.pl Portal rozwiązań IT w biznesie





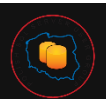
# SQL Server Transaction Log from A to Z

**Paweł Potasiński**

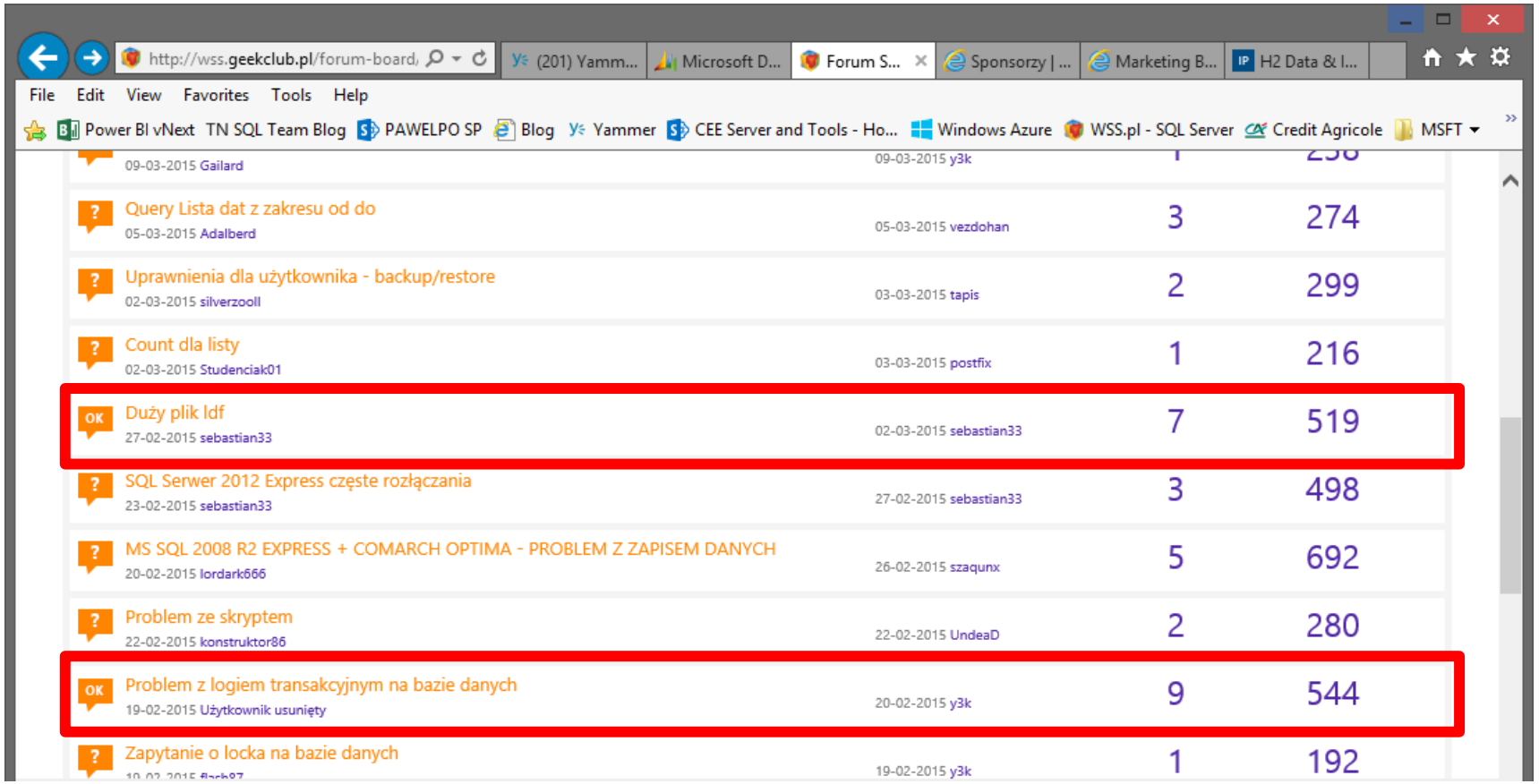
Product Manager Data Insights

pawelpo@microsoft.com

http://blogs.technet.com/b/sqlblog_pl/



# Why About Transaction Log (Again)?




http://zine.net.pl/blogs/sqlgeek/archive/2008/07/25/pl-m-j-log-jest-za-du-y.aspx

# Agenda

- Introduction
- Log Architecture
- Log Nature
- Tools
- Checkpoints
- Log Flushes and Limits
- Waits
- SQL Server 2014

# Introduction to Transaction Log

- Captures changes that occur in the database
- Changes occur here first (mostly)
- Implements Atomicity & Durability of ACID
- Written sequentially, so one is enough per database
- Memory-Optimized Tables have different logging semantics
- Useful for data recovery

# Recovery Models

- Simple Recovery
  - No log backups allowed
  - Potential data loss
- Full Recovery
  - Log backups are a MUST
  - Runaway log file otherwise
- Bulk Logged Recovery Model
  - Save log space with minimally logged operations
  - Makes log smaller, but not log backups
  - TF 610 used for clustered index inserts

# Best Practices

- Put the log on a fast drive (SSD, RAID-10, RAID-1)
- Separate log from data (LUNs)
- Turn off AutoShrink
- AutoGrow just as the last resort
- Regularly take transaction log backups

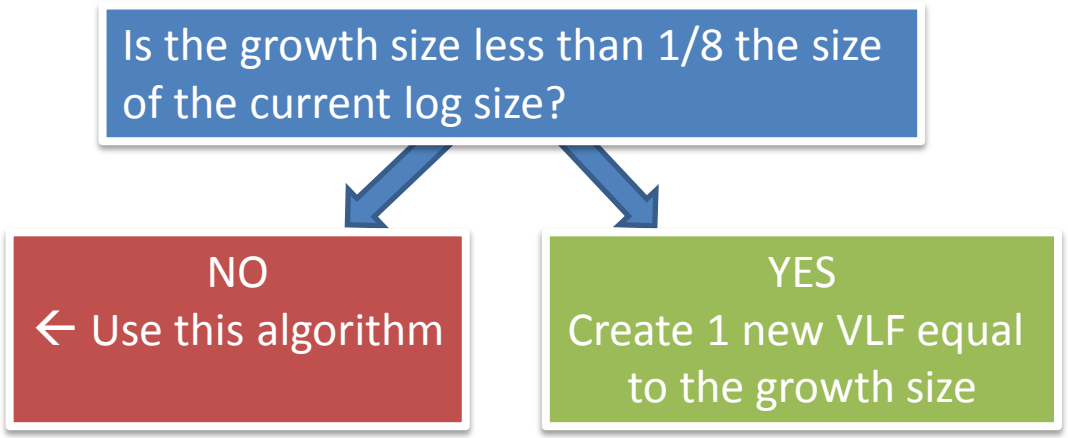
# Virtual Log Files (VLF)

- The unit of truncation of the transaction log file
- Generated at log file creation or log file growth

**SQL Server 2012 and earlier**

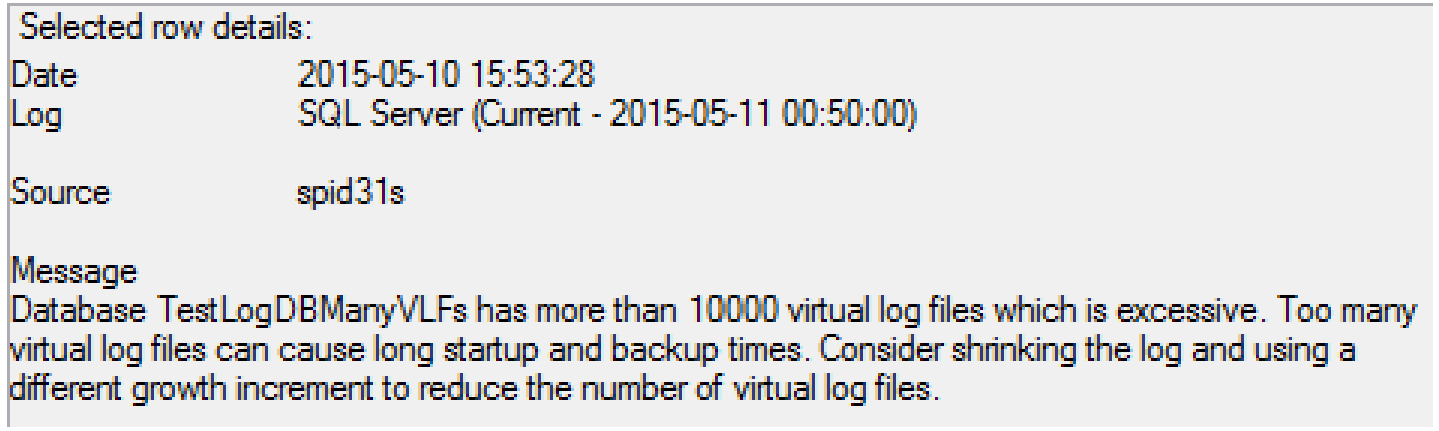
| Log File Growth | VLFs Added |
|---|---|
| <= 64MB | 4 |
| > 64 MB AND <= 1 GB | 8 |
| > 1 GB | 16 |

**SQL Server 2014**



# The Number of VLFs Matters

- Too many is bad, too few is also bad
- FIX: https://support.microsoft.com/en-us/kb/2455009
- SQL Server 2012+: https://support.microsoft.com/en-us/kb/2882905



# VLF Reuse

- A VLF can be reused if:
  - No active transactions are in the VLF or previous VLFs
  - No „unreplicated” transactions are contained in the VLF
- Occurs at Checkpoint in the Simple Recovery Model
- Otherwise, ONLY occurs at log backup

- Use sys.databases (*log_reuse_wait_desc*) to check why log cannot be reused

# VLF Reuse (ctnd)

**Log_reuse_wait_desc** in **sys.databases**

- NOTHING
- CHECKPOINT
- LOG_BACKUP
- ACTIVE_BACKUP_OR_RESTORE
- ACTIVE_TRANSACTION
- DATABASE_MIRRORING
- REPLICATION
- DATABASE_SNAPSHOT_CREATION
- LOG_SCAN
- OTHER_TRANSIENT

# Log Blocks & Log Records

**Log Block**

- The atomic unit of physical commit to a log file
- Contains a Header, Log Records, and a Slot Array
- Size from 512B to 60KB

**Log Record**

- An atomic database change
- Identified by Log Sequence Number
- Not only associated with committed transactions

# Log Sequence Number (LSN)

- Uniquely identifies a Log Record
- Monotonically increasing
- VLF Number: Log Block Offset: Slot Number
  - Every time a VLF is reused, it is given a new file sequence no

DBCC LOGINFO

| | FileId | FileSize | StartOffset | FSeqNo | Status | Parity |
|---|---|---|---|---|---|---|
| 1 | 2 | 253952 | 8192 | 38 | 2 | 64 |
| 2 | 2 | 253952 | 262144 | 39 | 2 | 64 |
| 3 | 2 | 253952 | 516096 | 40 | 2 | 64 |
| 4 | 2 | 278528 | 770048 | 41 | 2 | 64 |
| 5 | 2 | 253952 | 1048576 | 42 | 2 | 64 |
| 6 | 2 | 270336 | 1302528 | 43 | 2 | 64 |
| 7 | 2 | 253952 | 1572864 | 44 | 2 | 64 |
| 8 | 2 | 270336 | 1826816 | 45 | 2 | 64 |
| 9 | 2 | 253952 | 2097152 | 46 | 2 | 64 |
| 10 | 2 | 270336 | 2351104 | 47 | 2 | 64 |
| 11 | 2 | 253952 | 2621440 | 48 | 2 | 64 |
| 12 | 2 | 270336 | 2875392 | 49 | 2 | 64 |
| 13 | 2 | 253952 | 3145728 | 50 | 2 | 64 |
| 14 | 2 | 270336 | 3399680 | 51 | 2 | 64 |
| 15 | 2 | 253952 | 3670016 | 52 | 2 | 64 |

fn_dblog

| Current LSN |
|---|
| 00000026:00000036:0040 |
| 00000026:0000004f:0001 |
| 00000026:00000050:0001 |
| 00000026:00000050:0002 |
| 00000026:00000050:0003 |
| 00000026:00000052:0001 |
| 00000026:00000053:0001 |
| 00000026:00000053:0002 |
| 00000026:00000053:0003 |
| 00000026:00000058:0001 |
| 00000026:00000058:0002 |

# Checkpoint

- Minimizes the REDO portion of recovery

- Writes modified (dirty) buffer pages to disk (data file)
- Writes ALL dirty pages to disk – transaction state does not matter
- Can be very IO intensive

- Does not remove pages from the buffer pool
- Lazy Writer does this when there is pressure

# Checkpoint Options

- Option 'recovery interval' on the instance level
- In SQL Server 2012+ also on the database level
  - „Indirect checkpoint”

```
sp_configure 'recovery interval'
```

Results | Messages

| | name | minimum | maximum | config_value | run_value |
|---|---|---|---|---|---|
| 1 | recovery interval (min) | 0 | 32767 | 0 | 0 |

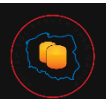
```
ALTER DATABASE AdventureWorks2012
SET TARGET_RECOVERY_TIME = 90 SECONDS
GO
SELECT name, target_recovery_time_in_seconds
FROM sys.databases
```

Results | Messages

| | name | target_recovery_time_in_seconds |
|---|---|---|
| 1 | master | 0 |
| 2 | tempdb | 0 |
| 3 | model | 0 |
| 4 | msdb | 0 |
| 5 | AdventureWorks2012 | 90 |
| 6 | SSISDB | 0 |
| 7 | sqlnexus | 0 |
| 8 | HeapsDB | 0 |
| 9 | MDW | 0 |

# Demo

## LOG NATURE
## DIGGING IN THE LOG



# Tools for Transaction Log

- DBCC SQLPERF(LOGSPACE)
- DBCC LOGINFO
- sys.fn_dblog()
- sys.fn_dump_dblog()

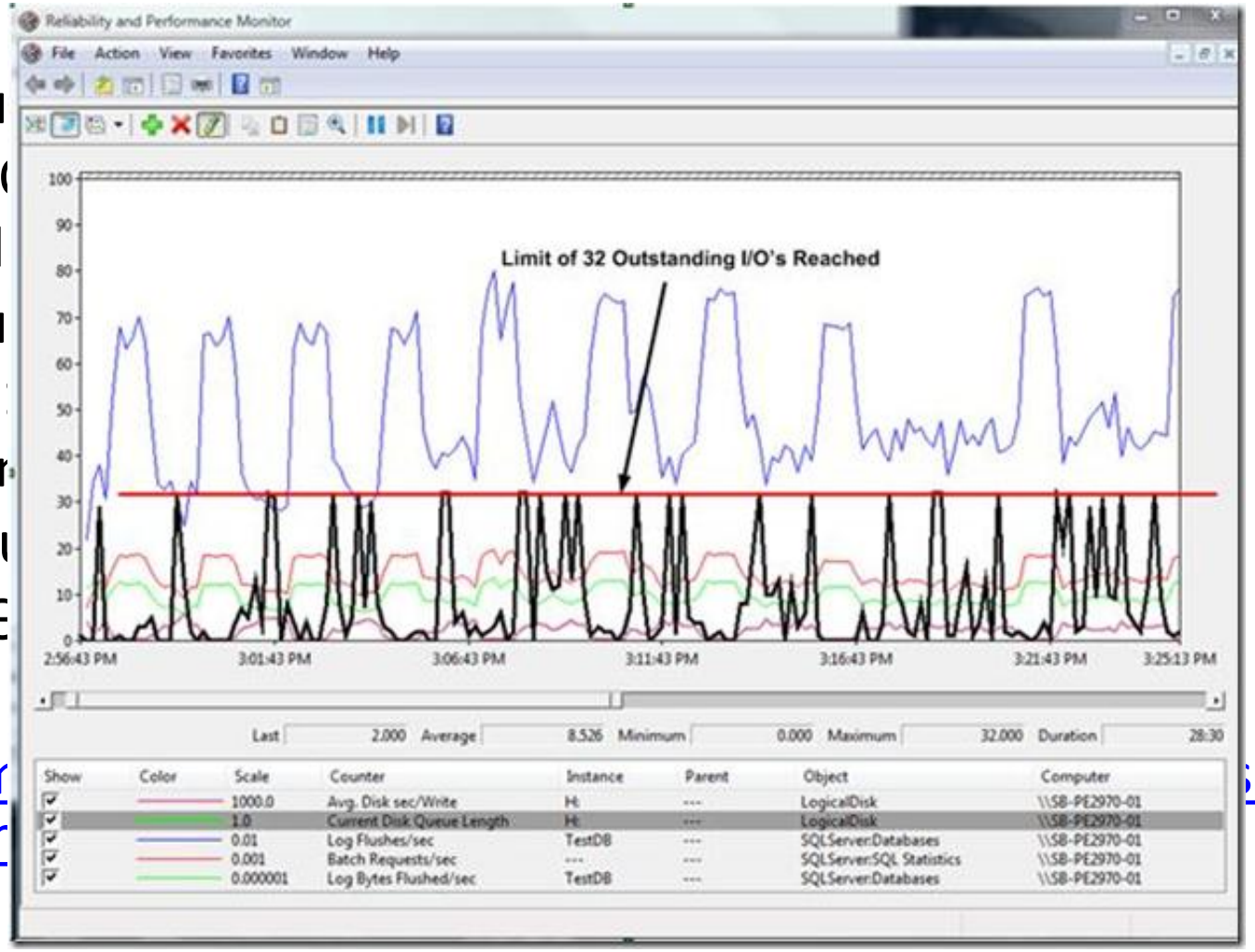
- DMVs & system views
- Extended Events
- Trace flag 3004 (info on file growth in SQL Server logs)

# Log Flushes

- On hidden scheduler
- Waiting on LOGMGR_QUEUE
- Processes completed log writes
- Background tasks LOG WRITER

# Transaction Log Manager Limits

- Number of ou
  - SQL 2012/S
  - SQL 2005 - I
- Number of ou
  - SQL 2012 –
  - Prior versio
- No new IO issu
- Wait type LOG
- http://blogs.m sing-transaction-log-performar



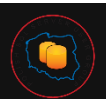
# Log Wait Types

| WRITELOG | LOGBUFFER | LOGMGR_RESERVE_APPEND | ASYNC_IO_COMPLETION |
|---|---|---|---|
| • Waiting on IO to Log file<br>• Slow disk possible<br>• Check log disk counters<br>• sys.dm_io_pending_io_requests | • Waiting on a free log buffer<br>• All log buffers in use<br>• This causes flushing not fast enough | • Couldn't grow the log<br>• Waiting on log truncation<br>• Check sys.databases log_reuse_wait_desc | • Can be for "zeroing" out a transaction log file during log creation or growth |

https://sqlserverperformance.wordpress.com/tag/dmv-queries/

# Demo

## TRANSACTIONS, WAITS, MONITORING



# SQL Server 2014: In-memory OLTP

- All logging for memory-optimized tables is logical
  - No log records for physical structure modifications
  - No index-specific / index-maintenance log records
- Log Records
  - Contains a log record header followed by memory optimized-specific log content
  - Are written only on a commit, no UNDO information is logged
  - None for Non-durable tables (SCHEMA_ONLY)
  - No Minimal –logging, everything is fully logged
  - Each physical log record is 24 KB
  - Logical records can exceed that limit and are chained
- New function sys.fn_dblog_xtp()

# SQL Server 2014: Delayed Durability

- Transaction commits logged asynchronously
- Set at database devel or ATOMIC block
  - DISABLED – normal behavior durability guaranteed
  - ALLOWED – allowed at the database level,
    - Transaction has to specify durability options, default is a durable transaction
    - COMMIT TRAN……. WITH (DELAYED_DURABILITY=ON)
  - FORCED – changes default durability for the database to „delayed”

```
CREATE PROCEDURE MyProc
WITH NATIVE_COMPILATION, SCHEMABINDING, EXECUTE AS OWNER
AS BEGIN ATOMIC WITH
(DELAYED_DURABILITY = ON,
TRANSACTION ISOLATION LEVEL = SNAPSHOT,
LANGUAGE = N'us_english')
          -- Insert T-SQL here…
END
```

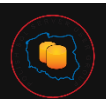
```
ALTER DATABASE Hekaton
SET DELAYED_DURABILITY = FORCED
GO
```

# Recap

- Log file performance is critical to SQL Server performance
- Always have your backup strategy ready
- There are tools helpful for log maintenance
- Be aware of the new features in SQL Server 2014

# Questions?

**PAWELPO@MICROSOFT.COM**





## Strategic Sponsor

Microsoft

## Gold Sponsors

PYRAMID ANALYTICS

KRAJOWY REJESTR DŁUGÓW www.krd.pl

devart Database Interaction Solutions

## Silver Sponsors

Profisee®

SQL EXPERT.pl

CREDIT SUISSE

## Technical Partners

Microsoft

PTI POLSKIE TOWARZYSTWO INFORMATYCZNE ODDZIAŁ GÓRNOŚLĄSKI

## Academic Partners

Wyższa Szkoła Zarządzania i Bankowości w Krakowie

Wyższe Szkoły Bankowe

## Media Partners

SPECJALISTA IT PROGRAMISTA DESIGNER GEEKCLUB.pl wss.pl

ERP24.pl

4programmers.net

IT professional

IN FACT PR

itnews24

virtual-it.pl

IT NEWS POLSKA

VirtualStudy

MAGAZYN IT PL

Made in Wro Do IT here!

ERP-view.pl Portal rozwiązań IT w biznesie