



Faster Db2 Active Log Writes with Media Manager Parallel Write Support using zHyperLink

Performance measurements using zHyperLink writes for active logs show that substantial reductions in log write I/O wait time have significantly improved transaction latency for update workloads.



By: Pooja Bhargava, Julie Chen, Neena Cherian, Brian Lee

Published: August 10th, 2021

Read time: 5 minutes

Background

zHyperLink is a low-latency short-distance connection between a z14 or later IBM Z and IBM storage controllers that allows for I/O to be performed synchronously¹. zHyperLink support for reads was delivered in 2017 and exploited by Db2 for database synchronous read. This article focuses on zHyperLink write enhancements. In 2019, zHyperLink write support was introduced for simplex and Metro Mirror-only environments, including multi-target. In 2020 for z/OS V2R3 and later, zHyperLink write support was provided for all forms of asynchronous mirroring in supported configurations such as Global Mirror with the exception of Extended Remote Copy (XRC). With the DFSMS Media Manager support for zHyperLink writes, Db2 12 for z/OS can improve log throughput by exploiting zHyperLink for active log writes. Media Manager traditionally allowed callers to perform I/O against a single data set. In 2021 for z/OS V2R3 and later, Media Manager was enhanced to provide support for parallel writes (also known as Media Manager Dual Logging). Callers can simultaneously perform update writes to two data sets in one request by using zHyperLink. Db2 exploits this enhancement to write to dual active logs, which is beneficial for most Db2 for z/OS configurations (see Figure 1).

Db2 active logs using zHyperLink with Media Manager dual logging

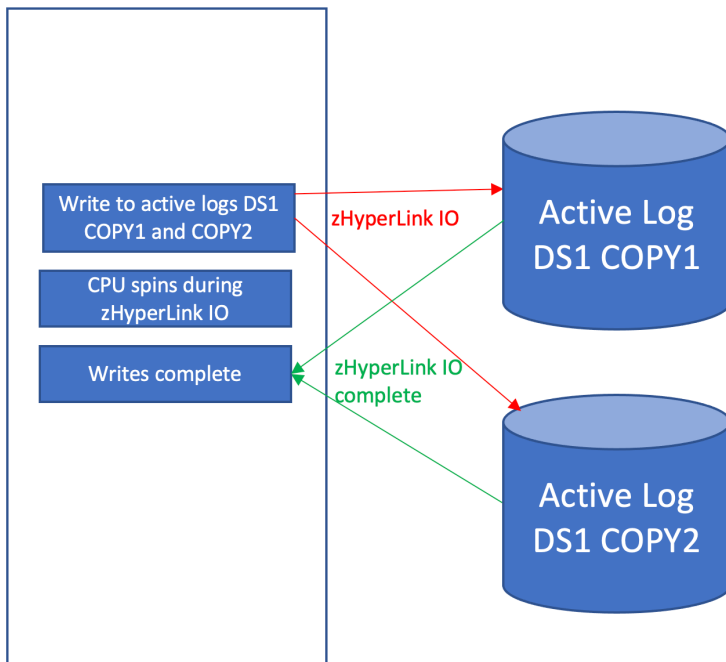


Figure 1: Db2 Active Logs using zHyperLink with Media Manager Parallel Write Support.

zHyperLink write processing speeds up commit by making the log write faster, which improves transaction response time, reduces the latency for log writes (Class 3), and significantly speeds up commits of the transactions. Because Db2 now performs log writes to the two active logs in parallel using zHyperLink synchronous I/O, the lock and latch held times are reduced without requiring any application changes.

Configuration

The online changeable ZHYPERLINK subsystem parameter specifies the scope of the zHyperLink protocol for I/O requests². To enable the zHyperLink protocol for I/O requests for active logs, specify either ENABLE or ACTIVELOG. The default value is DISABLE.

ENABLE: The zHyperlink protocol is implemented for all eligible I/O requests from Db2.

ACTIVELOG: The zHyperlink protocol is used only for write I/O requests from Db2 to active logs.

The active log data set must be associated with an SMS storage class where zHyperLink write is enabled. Both copies of the active log data sets must reside on IBM DS8880/DS8900 DASD subsystems with zHyperLink-enabled volumes in order to exploit the Media Manager parallel write feature.

Monitoring

To check whether Db2 active logs are exploiting the zHyperLink feature, issue the Db2 Display Log command. For example:

```
-DC1N DIS LOG
DSNJ370I  -DC1N DSNJCO0A LOG DISPLAY 639
CURRENT COPY1 LOG = DB2C1N0.DC1N.LOGCOPY1.DS01 IS 7% FULL
CURRENT COPY2 LOG = DB2C1N0.DC1N.LOGCOPY2.DS01 IS 7% FULL
          H/W RBA = 000000000D7CC1C598A8
          H/O RBA = 00000000000000001DFFF
          FULL LOGS TO OFFLOAD = 0 OF 6
          OFFLOAD TASK IS (AVAILABLE)
          SOFTWARE ACCELERATION IS DISABLED
          ZHYPERLINK WRITE IS ENABLED
          LAST LOG WRITE FOR COPY1 USED ZHYPERLINK
          LAST LOG WRITE FOR COPY2 USED ZHYPERLINK
DSNJ371I  -DC1N DB2 RESTARTED 07:38:15 FEB  5, 2021 640
          RESTART RBA 000000000D7CBE0A9000
          CHECKPOINT FREQUENCY 5 MINUTES
          LAST SYSTEM CHECKPOINT TAKEN 07:38:17 FEB  5, 2021
DSN9022I  -DC1N DSNJCO01 '-DIS LOG' NORMAL COMPLETION
```

Figure 2: Db2 Log Display Command Output.

The Media Manager support for parallel writes added additional zHyperLink write statistics. They can be found in both the output from the display command `D SMS,DSNAME(dsn),STATS(ZHLWRITE{,RESET})` and in SMF type 42 subtype 6 synchronous I/O section 2. For the details of the command output and the SMF 42 subtype 6 fields, see z/OS MVS System Messages³ and z/OS MVS System Management

Facilities (SMF)⁴. Use the display command to display the percentage of synchronous I/O being done. For example:

```

IGW289I 662
D SMS,DSNAME,STATS(ZHLWRITE) Start of Report
DATA SET DB2C1N0.DC1N.LOGCOPY1.DS01.DATA
STATISTICS Since 02/05/2021 07:38:15.373500
SUMMARY
      TOTAL      %SYNC  -----%ASYNC-----
WRITE REQUESTS WRITES   SKIP LNKBSY   TEST   MISC DISABL
      51346   99.99    0.00   0.00  <0.01   0.00   0.00
      -----%ASYNC-----
                MISS DELAY   DUAL
                0.00   0.00   0.00

DEVICE STATISTICS
      TOTAL      %SYNC  -----%ASYNC-----
SSID DEVNO WRITES WRITES   SKIP LNKBSY   TEST   MISC   MISS DELAY
AD00 0B024  51826  99.99    0.00   0.00  <0.01   0.00   0.00  0.00
D SMS,DSNAME,STATS(ZHLWRITE) End of Report

```

Figure 3: Display zHyperLink Write Statistics Command Output.

In Db2, two new fields for zHyperLink activity for active logs were added to IFCID 1 section 12 (QJST).

1. QJSTSYCW: Total number of zHyperLink log write I/O requests.
2. QJSTSYCF: Total number of zHyperLink log write I/O requests that failed to be synchronous.

Performance

Writes to Db2 active log data sets are crucial to performance because each transaction waits for log writes to complete before it can proceed. Db2 exploits zHyperLink write capability for active log writes to reduce log write wait time in Db2 (Class 3) with the potential to improve transaction latency time.

Improvement in transaction latency time depends on the type of workload. Workloads that are update-intensive with frequent commits will benefit the most from implementing zHyperLink for Db2 active log writes.

The following example shows an insert-intensive workload in a one-way data sharing environment with 50 threads concurrently inserting ten rows per commit to different partitions in a table. zHyperLink was enabled in Db2 with the ZHYPERLINK subsystem parameter ACTIVELOG setting. When zHyperLink I/O was used to write to active dual logs, there was a 54% reduction in Class 3 active log write I/O wait time in Db2. This reduction improved transaction latency by 24% compared to traditional I/O using zHPF (z High Performance FICON) link (see Figure 4).

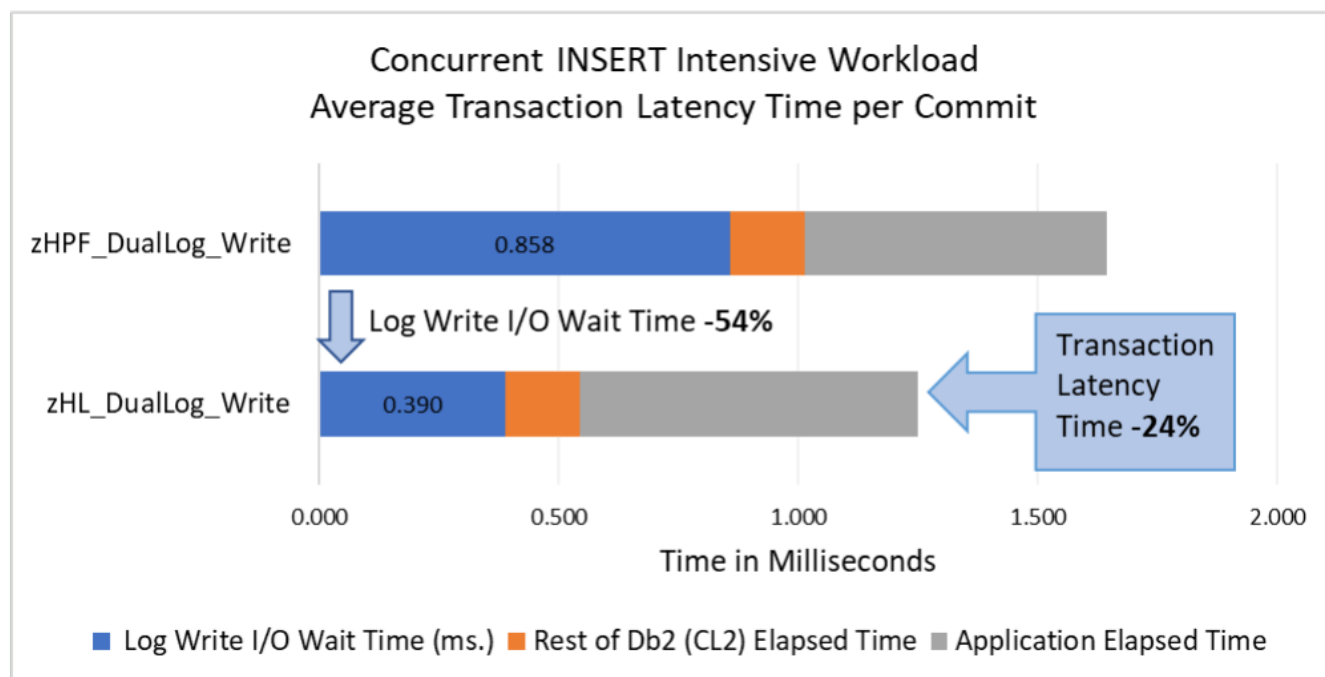


Figure 4: Comparing Transaction Latency Time for zHPF and zHyperLink Parallel Writes.

When Db2 writes to active log data sets using zHyperLink I/O, unlike traditional I/O, the CPU spins while waiting for the I/O to complete. The clock time of the CPU spin is charged to System Services (MSTR) CPU under SRB in Db2 and is zIIP-eligible. zIIP usage does not incur additional charges. For an insert-intensive workload, there was a substantial increase in zIIP CPU utilization with no increase in Class 2 CPU time (see Figure 5). You must consider the current usage of zIIP processors and plan for more capacity before implementing zHyperLink writes for active logs.

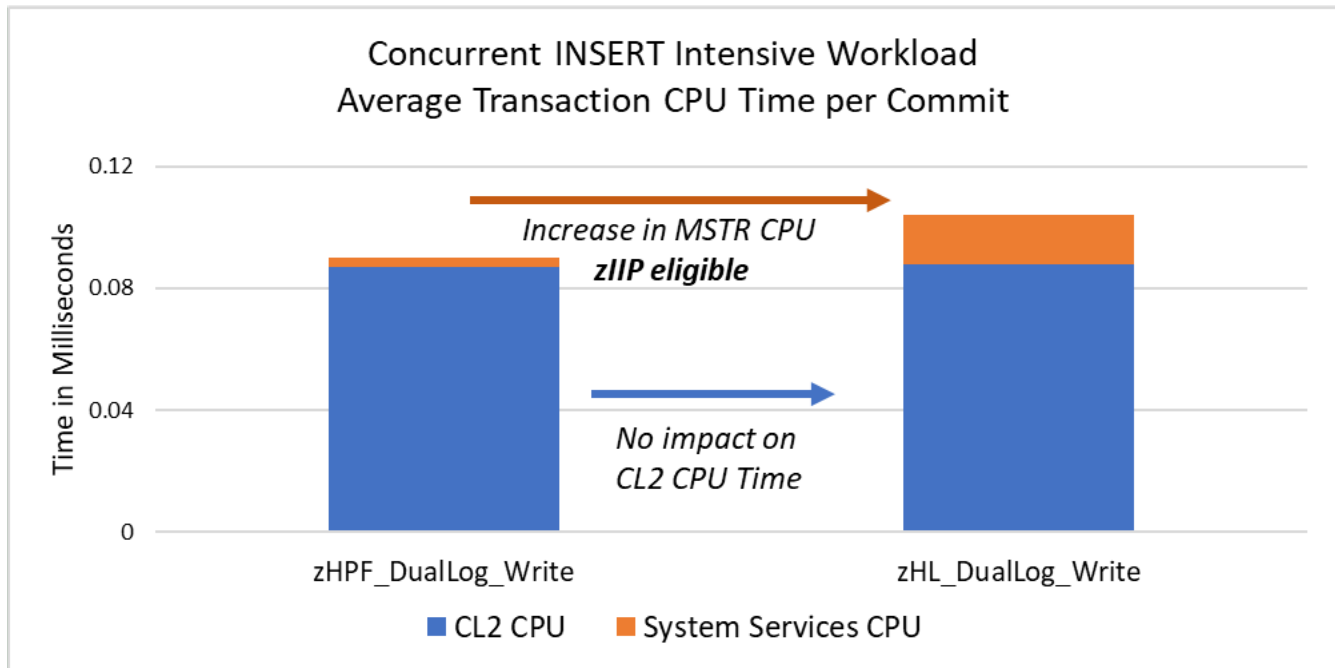


Figure 5: Comparing CPU usage for zHPF and zHyperLink Parallel Writes.

In addition to active log writes during the update commit, zHyperLink provides the same benefit to other types of Db2 log writes that are triggered by Db2. A good example is the active log force writes during an index tree modification against indexes that are dependent on group buffer pools. The modification can be an index-leaf page split from the insert operations or an index page consolidation from the deletes. During the tree modification, Db2 holds the internal latch against the index tree while writing to active log data sets. Holding the internal latch is essential to maintain the data integrity, but often causes a bottleneck in insert or delete-intensive applications in the data sharing environment. The low latency from zHyperLink I/O can significantly reduce the duration of log write and Db2 latch wait and also improve Db2 application throughput.

Workloads with a high read-to-write ratio may not always show significant improvement in transaction latency if log write I/O is not a major contributor to transaction-elapsed time.

Software Requirements

The following PTFs provide the Media Manager support for parallel writes.

OA57833: UJ04670 (V2R3), UJ04671 (V2R4)

OA58134: UJ04661 (V2R3), UJ04662 (V2R4)

OA59581: UJ04668 (V2R3), UJ04669 (V2R4)

The following PTF provides Db2 12 for z/OS exploitation of Media Manager parallel writes.

PH29407: UI74012

Find the PTFs for zHyperLink by searching for Fix Category IBM.Function.zHyperLink, APAR keyword HYPERL/K.

For a complete list of requirements for using zHyperLink in Db2, refer to these articles:

- [Lightning Fast I/O via zHyperLink and Db2 for z/OS Exploitation](#)
- [Lightning Fast I/O via zHyperLink and Db2 for z/OS Exploitation – Part 2](#)

Summary

Performance measurements using zHyperLink writes for active logs show that substantial reductions in log write I/O wait time have significantly improved transaction latency for update workloads. Be aware of hardware restrictions and plan for an increase in zIIP CPU before considering zHyperLink for Db2 active log writes.

Acknowledgments

We would like to thank Akiko Hoshikawa, Distinguished Engineer, Db2 for z/OS Development, for reviewing and providing valuable technical contribution to this article. We would also like to acknowledge Sharon Roeder, Db2 Developer, for her review and helpful comments.

About the authors

Pooja Bhargava worked on supporting zHyperLink and dual logging in Db2 development and is currently a z/OS DFSMS Developer.

Julie Chen is with Db2 for z/OS Log Manager development.

Neena Cherian is with Db2 for z/OS Performance.

Brian Lee is a Software Developer for z/OS DFSMS Media Manager.

Debbie Noll contributed to the editorial review of this article.

-
1. [IBM DS8000 Series V9.0.0 documentation: zHyperLink for the DS8880F overview](#)

[↩](#)

2. [Db2 12 for z/OS: DB2 zHyperLinks SCOPE field \(ZHYPERLINK subsystem parameter\)](#)

[↩](#)

3. [z/OS MVS System Messages, Vol 9 \(IGF-IWM\): IGW289I](#)[↩](#)

4. [z/OS MVS System Management Facility \(SMF\): Synchronous I/O Section 2](#)