More Data, Less Chatter: Improving Performance on z/OS with IBM zHPF

Holden O’Neal
Session 17576
August 14, 2015
Overview

1. What problem does zHPF solve
2. Overview of zHPF and History
3. How zHPF is used by zOS
4. SAS internal test results
5. How do you get zHPF
Lets plan Lunch!

Person 1

Do you want pizza? 

How about Chinese? 

What would you like? 

Hotcakes? 

…

Person 2

No I just had Pizza last night 

No I just had Chinese for lunch yesterday 

I don’t know 

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Let's try that again...

Person 1:

We have three lunch options:
- Pizza, Chinese, or
- Holden's House of Hotcakes

Person 2:

Hotcakes sounds good.
How are we getting there?

Person 1:

Let's just walk, it'll be easier than catching a cab

Person 2:

Agreed, meet you in the lobby in ten minutes
What is zHPF?

- **Hiper Performance Ficon for SystemZ**
- zHPF is an extension to FICON Architecture
- Improve Small Block I/O
History of zHPF and FICON

- Oct 2008: Initial announcement of System z10 with FICON Express4 and FICON Express2 channels
  - Single track limit for zHPF data transfers
  - Max of 31k zHPF IOs/sec
- July 2009: FICON Express8 channels on System z10
  - 64k byte limit for zHPF data transfers
  - Max of 52k zHPF IOs/sec
- July 2010: Additional Support on zEnterprise 196
  - Extension to multi-tracks of zHPF data transfers
- July 2011: Introduced FICON Express8S Channel
  - Hardware data router introduced
  - 100% DB2 I/O converted to zHPF
FICON and zHPF performance on z Systems

Photo and tests credited to IBM

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
ESCON

- Original System/360 disk drive controllers lacked on-board processors
- Supports only CCW
- Must wait for Channel End/Device End
- Jobs monopolize channel
- More channel paths required

FICON

- Support previous CCW & zHPF
- CCWs can be transferred without waiting for command response
- Multiple I/O operations concurrently on a channel
- Shares the pipe
- zHPF reduces connection time to I/O device
Questions?

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Channel Architecture Overview

Figure 2 - Relationship of Channel Subsystem to Other Processor Components

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Channel Command Word Program (CCP)

Figure 1. CCW Channel Program Example

Photo credited to IBM

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Transport Command Word (TCW)

Figure 2. Example of TCW Channel Program Example
           Designating I/O

Photo credited to IBM

Complete your session evaluations online at www.SHARE.org/Orlando-Eval

8/21/2015
Questions?

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
zHPF vs FICON

Single FICON Express8S channel: zHPF vs FICON READ 4k bytes/IO
Total I/O Response Time vs IO/sec

Figure 2

Photo credited to IBM

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
zHPF and CCP Mixtures

Photo credited to IBM

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
DB2 10

- Disorganized Indexes
- Sorted Row ID (RID) Lists
  - No more “death by table scan”
- RUNSTATS
  - Page sampling reduces CPU time and uses less buffers
  - Page sampling becomes faster than Row Sampling
QSAM Benefits

FEx8S channel with QSAM max MB/sec measurement results using FICON and zHPF protocols

Photo credited to IBM

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
SAS Internal Testing Results

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
Parameters for Testing

- SAS Version 9.4 3rd Maintenance
- Little to no computational processing
- 27K Block Size
- SAS Direct Access Bound Libraries
- Tall and Wide

<table>
<thead>
<tr>
<th>Member</th>
<th>Page Size</th>
<th>Number of Pages</th>
<th>Size in Mbytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tall</td>
<td>27648</td>
<td>240176</td>
<td>6333</td>
</tr>
<tr>
<td>Wide</td>
<td>55296</td>
<td>147061</td>
<td>7755</td>
</tr>
</tbody>
</table>
## Increased I/O Throughput
### BLKSIZE = 27K

<table>
<thead>
<tr>
<th>Type of Test</th>
<th>Member</th>
<th>Elapsed Seconds CCW</th>
<th>Elapsed Seconds zHPF</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Read Tall</td>
<td>32.7</td>
<td>22.2</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>Random Read Wide</td>
<td>39.3</td>
<td>29.1</td>
<td>0.74</td>
<td></td>
</tr>
<tr>
<td>Random Write Tall</td>
<td>129.5</td>
<td>89.7</td>
<td>0.69</td>
<td></td>
</tr>
<tr>
<td>Random Write Wide</td>
<td>124.2</td>
<td>89.0</td>
<td>0.72</td>
<td></td>
</tr>
<tr>
<td>Sequential Read Tall</td>
<td>57.7</td>
<td>48.0</td>
<td>0.83</td>
<td></td>
</tr>
<tr>
<td>Sequential Read Wide</td>
<td>35.8</td>
<td>25.4</td>
<td>0.71</td>
<td></td>
</tr>
</tbody>
</table>

Median Elapsed Time for CCW versus zHPF
Channel Programs – Library BLKSIZE = 27K

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
zHPF Improves I/O Rate
BLKSIZE = 27K

<table>
<thead>
<tr>
<th>Type of Channel Program</th>
<th>Elapsed Seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCW</td>
<td>419.2</td>
</tr>
<tr>
<td>zHPF</td>
<td>303.4</td>
</tr>
</tbody>
</table>

Ratio (zHPF/CCW) 0.72

Improvement 28%

- Using recommended Library Block Size of 27K
- 28% less elapsed time

Complete your session evaluations online at www.SHARE.org/Orlando-Eval
# Increased I/O Throughput

**BLKSIZE = 6K**

<table>
<thead>
<tr>
<th>Type of Test</th>
<th>Member</th>
<th>Elapsed Seconds CCW</th>
<th>Elapsed Seconds zHPF</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Read</td>
<td>Tall</td>
<td>41.82</td>
<td>25.5</td>
<td>0.61</td>
</tr>
<tr>
<td>Random Read</td>
<td>Wide</td>
<td>48.06</td>
<td>32.41</td>
<td>0.67</td>
</tr>
<tr>
<td>Random Write</td>
<td>Tall</td>
<td>145.97</td>
<td>103.01</td>
<td>0.71</td>
</tr>
<tr>
<td>Random Write</td>
<td>Wide</td>
<td>171.1</td>
<td>102.04</td>
<td>0.60</td>
</tr>
<tr>
<td>Sequential Read</td>
<td>Tall</td>
<td>78.93</td>
<td>48.97</td>
<td>0.62</td>
</tr>
<tr>
<td>Sequential Read</td>
<td>Wide</td>
<td>61.79</td>
<td>30.65</td>
<td>0.50</td>
</tr>
</tbody>
</table>
zHPF Improves I/O Rate  
BLKSIZE = 6K

<table>
<thead>
<tr>
<th>Type of Channel Program</th>
<th>Elapsed Seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCW</td>
<td>547.67</td>
</tr>
<tr>
<td>zHPF</td>
<td>342.58</td>
</tr>
<tr>
<td>Ratio (zHPF/CCW)</td>
<td>0.63</td>
</tr>
<tr>
<td>Improvement</td>
<td>37%</td>
</tr>
</tbody>
</table>

- Using Library Block Size of 6K
- 37% less elapsed time
- Strength of zHPF
### zHPF Improves Channel Utilization – 8 Way

<table>
<thead>
<tr>
<th>Type of Channel Program</th>
<th>Channel Utilization</th>
<th>Percentage of Bus Cycles Busy</th>
<th>Data Transfer Rate (MB/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCW</td>
<td>67%</td>
<td>28%</td>
<td>440</td>
</tr>
<tr>
<td>zHPF</td>
<td>16%</td>
<td>44%</td>
<td>706</td>
</tr>
</tbody>
</table>

- 8 way Sequential Read
- Dedicated System and 2 Dedicated Channels
- zHPF had higher I/O rate but used less channel capacity.
### zHPF Improves Channel Utilization – 64 Way

<table>
<thead>
<tr>
<th>Type of Channel Program</th>
<th>Channel Utilization</th>
<th>Percentage of Bus Cycles Busy</th>
<th>Data Transfer Rate (MB/s)</th>
<th>Avg Connect Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCW</td>
<td>90%</td>
<td>25%</td>
<td>401</td>
<td>1.121</td>
</tr>
<tr>
<td>zHPF</td>
<td>41%</td>
<td>50%</td>
<td>797</td>
<td>0.135</td>
</tr>
</tbody>
</table>

- 64 way Random Read
- zHPF aggregate data rate almost 2xCCW
- zHPF has a significantly lower average connect time
# IBM zHPF Support Requirements

<table>
<thead>
<tr>
<th>Component</th>
<th>Requirements and Recommendations</th>
</tr>
</thead>
</table>
| **z/OS**          | One of the following Levels of z/OS:  
|                   | • V1R12 (with APAR OA38185)  
|                   | • V1R13 (with APAR OA38185)  
|                   | • V2R1  
|                   | - ZHPF = YES must be set in the IECIOSxx member of  
|                   | SYS1.PARMLIB (default for this parameter is ‘NO’)  
|                   | - z/OS must NOT be running as a guest under z/VM  
|                   | - We recommend that the fix be applied for APAR OA45589 – An incorrect  
|                   | zHPF Channel Program might be generated in certain retry situations |
| **Processor**     | See paper reference “Which servers support zHPF?” in Burgess (2014) |
| **DASD Controller** | One of the following models in the IBM DS8000 series:  
|                   | • DS8700 with maintenance bundle 76.20.90.0 or higher  
|                   | • DS8800 with maintenance bundle 86.31.110.00 or higher  
|                   | • DS8870 |
How to use zHPF in SAS

• Implement the appropriate level of zHPF support for your System Z hardware

• Release SAS 9.4 Maintenance 2:
  – Read operations

• Release SAS 9.4 Maintenance 3:
  – Read and Write operations
Evaluating Performance – DLDEBUG=1X

libname mylib '.mylib.saslib' dldebug=1x; run;
libname mylib clear; run;

Channel Program Generation Statistics for Library HOONEA.MYLIB.SASLIB:

<table>
<thead>
<tr>
<th>Method</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NumOperTotal</td>
<td>3</td>
</tr>
<tr>
<td>NumOperErrSchd</td>
<td>0</td>
</tr>
<tr>
<td>NumOperErrComp</td>
<td>0</td>
</tr>
<tr>
<td>NumOperRead</td>
<td>0</td>
</tr>
<tr>
<td>NumOperWrite</td>
<td>3</td>
</tr>
</tbody>
</table>
Evaluating Performance – LIBNAME LIST

- LIBNAME LIST statement display channel program type in 9.4 maintenance release 3

```
libname sashelp list;
```

NOTE: Libref= SASHELP

......

Channel Program Type= zHPF
Evaluating Performance – DLDEBUG=20000x

- How to ensure a library is using CCW and not zHPF
  
  libname x '.base.saslib' dldebug=20000x;
  libname x list; run;
  libname x list;

  NOTE: Libref= X

  Physical Name= HOONEA.BASE.SASLIB
    Unit= DISK
    Device= 3390
    Blocksize= 27648
    Blocks per Track= 2
    Total Library Blocks= 30
    Total Used Blocks= 12
    Percent Used Blocks= 40.0%
    Members= 0
    DSNTYPE= BASIC
    Data Representation= MVS_32
    Channel Program Type= CCW
Acknowledgements

• Lewis King, SAS (co-author)
• Fred Forst, SAS (co-author)
• Dale Riedy, IBM z/OS IOS Design and Development
Thank you for your time

• Holden O’Neal
• Email : Holden.O’Neal@sas.com
• Phone : 919-531-2312
• 100 SAS Campus Drive, Cary NC 27513
• Feel free to contact me if you have any questions or concerns