

Getting Started with IBM zHyperLink for z/OS

Dale Riedy

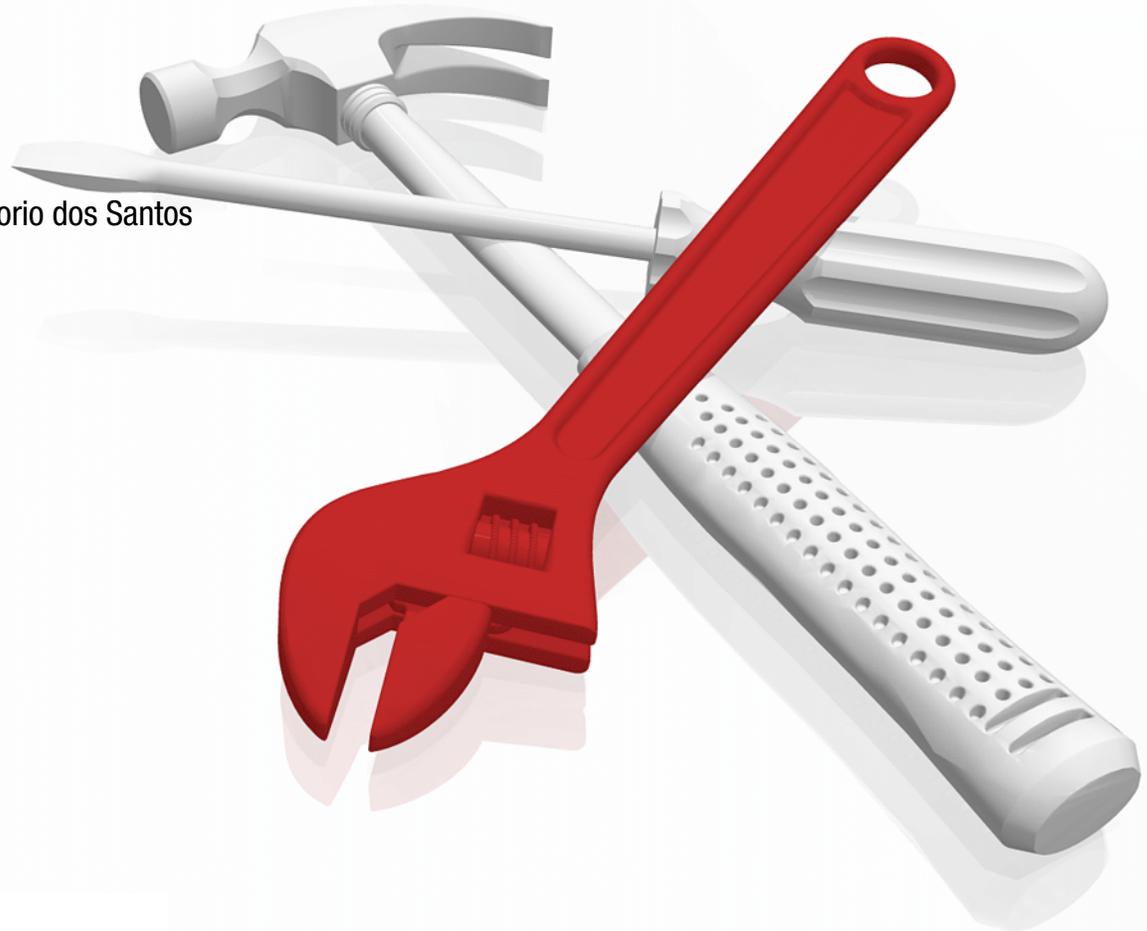
Preben Esbensen

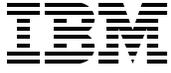
Marcelo Lopes de Moraes

Romoaldo Wellington Honorio dos Santos

Octavian Lascu

Ewerson Palacio





International Technical Support Organization

Getting Started with IBM zHyperLink for z/OS

April 2022

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

Second Edition (April 2022)

This edition applies to IBM Z®: IBM z15™.

This document was created or updated on May 7, 2024.

© Copyright International Business Machines Corporation 2018, 2022. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	vii
Trademarks	viii
Preface	ix
Authors	ix
Now you can become a published author, too!	x
Comments welcome	xi
Stay connected to IBM Redbooks	xi
Chapter 1. Introduction	1
1.1 Introduction to zHyperLink	2
1.1.1 Synchronous versus asynchronous I/O	2
1.2 Use cases	3
1.2.1 Avoiding financial losses	3
1.2.2 Growing business opportunities	4
1.2.3 Better client experience and revenue	4
1.2.4 Faster recovery from outages	4
1.2.5 Workload growth without redesigning applications	4
1.2.6 Reducing data sharing overhead	5
1.3 zHyperLink prerequisites	5
1.3.1 Hardware prerequisites	5
1.3.2 Storage systems zHyperLink support	7
1.3.3 Software requirements	8
1.4 Additional considerations	9
1.4.1 Site considerations	9
1.4.2 ECKD	12
1.4.3 DS8000 I/O priority manager	12
1.4.4 FICON I/O workload	13
Chapter 2. Planning	15
2.1 Identifying zHyperLink eligible workloads	16
2.1.1 zHyperLink in site configurations overview	17
2.1.2 Single-site	17
2.1.3 Multi-site with synchronous replication	18
2.1.4 Consistent read from secondary storage with synchronous replication	19
2.1.5 Multi-site with extended distance and asynchronous replication	21
2.1.6 IBM z Batch Network Analyzer	22
2.2 Sizing synchronous I/O for zHyperLink	27
2.2.1 Number of connections	27
2.2.2 Using Resource Measurement Facility data for sizing	28
2.2.3 I/O connectivity planning	29
2.3 zHyperLink hardware planning	29
2.3.1 Microcode for zHyperLink	29
2.3.2 IBM z14 and IBM z15 microcode change level	30
2.3.3 IBM DS8000 microcode change level	31
2.3.4 IBM z14 and IBM z15 feature codes	32
2.3.5 DS8000 feature codes	32
2.3.6 zHyperLink cables	32
2.3.7 I/O configuration	34

2.3.8 Access limitations	36
2.4 Software planning	36
2.4.1 Levels and requirements	36
2.4.2 Software parameters for z/OS	37
2.4.3 zHyperLink Storage Class Granularity activation plan	38
2.4.4 Software activation plan for Db2	40
2.5 Performance planning	41
2.5.1 CPU considerations	41
2.5.2 Db2 performance considerations	41
2.5.3 Measuring before deployment	42
2.6 Monitoring	44
2.6.1 Monitoring PCIe	44
2.6.2 Monitoring subsystem	44
2.6.3 Monitoring devices	45
Chapter 3. Deployment	47
3.1 Hardware deployment	48
3.1.1 Ordering features	48
3.1.2 Updating microcode	49
3.1.3 Enabling the DS8000 zHyperLink feature	49
3.1.4 Installing features	50
3.1.5 Activating zHyperLink on the DS8000	50
3.1.6 Updating I/O configuration	52
3.1.7 Verifying zHyperLink cables	58
3.2 Software upgrade and maintenance	60
3.2.1 Security profiles	60
3.2.2 Product upgrades	60
3.2.3 Product maintenance	61
3.3 zHyperLink enablement	62
3.3.1 Verifications before activation	62
3.3.2 Activating zHyperLink	64
3.4 Monitoring	68
3.4.1 System automation	68
3.5 Activating zHyperLink for VSAM data sets	69
3.5.1 zHyperLink for SMS data sets	69
3.5.2 zHyperLink for non-SMS data sets	70
3.6 Activating zHyperLink for Db2	71
3.7 Measuring after deployment	72
3.7.1 Measuring by using Resource Monitoring Facility	72
3.7.2 Measuring using zBNA	77
Chapter 4. Operations	79
4.1 System commands	80
4.1.1 Display commands	80
4.1.2 Modify commands	88
4.1.3 Configuration commands	90
4.2 System automation	91
4.2.1 Error messages	91
4.2.2 Informational messages	92
4.3 System messages	93
4.3.1 Error messages	94
4.3.2 Informational messages	94
4.4 Hardware Management Console	99

4.5 Troubleshooting	104
4.5.1 zHyperLink port in CNFG status	104
4.5.2 zHyperLink port in STNB status	105
4.5.3 zHyperLink is disabled	106
4.5.4 zHyperLink is degraded	107
4.5.5 zHyperLink cabling issue	108
4.5.6 Other reasons that zHyperLink is not used.....	108
Appendix A. zHyperLink checklist	109
zHyperLink implementation checklist	110
Appendix B. Problem determination	113
Where do I look to see if zHyperLink is being used?	114
Why is zHyperLink not used?	114
Why are zHyperLink I/Os failing?	115
zHyperLink cable problems	116
Commands to use for problem determination	116
Appendix C. Sample IOCDS	121
Sample IOCDS	122
Appendix D. Mapping DCOLLECT data	125
DCOLLECT mapping	126
Appendix E. Consistent Read from Metro Mirror Secondary	127
Overview	128
Consistent reads from Metro Mirror Secondary	129
Consistent read management	129
z/OS requirements and behavior	129
Supported data sets	131
Consistent read from secondary support for zHyperlink	131
Extended Long Busy	131
Enabling a device for Consistent Read from Secondary	132
How it works	132
Related publications	135
IBM Redbooks	135
Other publications	135
Online resources	135
Help from IBM	136

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

Db2®	HyperSwap®	Redbooks (logo)  ®
DB2®	IBM®	Resource Link®
DS8000®	IBM Z®	System z®
Easy Tier®	IBM z Systems®	z Systems®
FICON®	IBM z14®	z/OS®
FlashCopy®	Interconnect®	z/VM®
GDPS®	POWER®	z15™
Global Technology Services®	Redbooks®	

The following terms are trademarks of other companies:

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, or service names may be trademarks or service marks of others.

Preface

With the pressures to drive transaction processing 24/7 because of online banking and other business demands, IBM® zHyperLink® for IBM z/OS® is making it easy to accelerate transaction processing for the mainframe.

This IBM Redpaper™ publication helps you to understand the concepts, business perspectives, and reference architecture of installing, tailoring, and configuring zHyperLink in your own environment.

Authors

This paper was produced by a team of specialists from around the world working at the International Technical Support Organization, Poughkeepsie Center.

Dale Riedy is a Senior z/OS Software developer. He works in IBM Poughkeepsie Center with over 40 years of experience and specializes in z/OS I/O.

Preben Esbensen is an IBM Certified Consulting IT Specialist at IBM Global Technology Services®, Solutions, Delivery & Transformation in Denmark. Preben has 30 years experience as a z/OS Systems Programmer, and has been responsible for z Server management and design for 20 years. Preben's other areas of expertise include disaster recovery, IBM® GDPS, parallel sysplex, zSeries solution design, and Linux on IBM Z. Preben also teaches customers and at IBM internally. He had a lead role in several hardware projects, including data center migrations, infrastructure, standardization, and hardware consolidation projects. He joined IBM in 2004.

Marcelo Lopes de Moraes is an IBM Certified IT Specialist in Integrated Technology Delivery, Server Systems Operations/Storage Management in IBM Brazil. He started at IBM in 2004 working with systems z on Mainframe System Operations. Marcelo has been working on storage management for eight years and worked on big tape refreshes and data center move projects to attend customer needs, which reduced data center floor space, power, and cooling. Although his experience is mostly with IBM products (DFSMSdfp, DFSMSdss, DFSMSshm, and DFSMSrmm), he also works with other equipment manufacturer (OEM) software products. Marcelo's areas of expertise include installing, customizing, and maintaining storage products, and developing automation processes.

Romoaldo Wellington Honorio dos Santos is an IBM Z® consultant in IBM Brazil. He has 38 years experience with the mainframe and is currently working on virtualization and data center platforms. Romoaldo is responsible for hardware configurations in IBM Global Accounts and supports GDPS® environments in EMEA and the US. He also works on high availability and business continuance projects. Other areas of expertise include IBM Z I/O Supervisor, Assembler, Sysplex problem determination, performance and tuning, and z/OS® internals.

Octavian Lascu is a project leader working for IBM Redbooks®. He has over 30 years of experience in IT Infrastructure, specializing in designing and implementing complex solutions that are based on IBM Z and IBM Power platforms.

Ewerson Palacio is an IBM Redbooks publications project leader. He holds Bachelors' degree in Math and Computer Science. Ewerson worked for IBM Brazil for over 40 years and retired in late 2016 as an IBM Distinguished Engineer. Ewerson co-authored many IBM Z Redbooks publications, and created and taught IBM Z technical workshops and seminars for IBM's International Technical Support Organization (ITSO) to customers, IBMers, and Business Partners around the globe.

Many thanks to the authors of the first edition of this paper:

- ▶ Jose Gilberto Biondo Junior
- ▶ Preben Esbensen
- ▶ Marcelo Lopes de Moraes

Thanks to the following people for their contributions to this project:

Robert Haimowitz
International Technical Support Organization, Poughkeepsie Center

Thomas Ambrosio
William G Lamastro
CPO Poughkeepsie Competitive Center

John Paveza
Peter G Sutton
Susan Candelaria
Matt Kalos
Wayne Rhoten
Tim Raley
IBM US

Renato Carmesim de Oliveira Bueno
IBM Brazil

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks publications in one of the following ways:

- ▶ Use the online **Contact us** review Redbooks form found at:

ibm.com/redbooks

- ▶ Send your comments in an email to:

redbooks@us.ibm.com

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- ▶ Look for us on LinkedIn:

<http://www.linkedin.com/groups?home=&gid=2130806>

- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>

- ▶ Stay current on recent Redbooks publications with RSS Feeds:

<http://www.redbooks.ibm.com/rss.html>



Introduction

The business requirements for faster transactions and lower response times for applications drive new technologies. These technologies are designed to reduce latency that is related to processing data from backend storage, such as IBM Easy Tier® and flash storage.

Although these solutions help to improve (shorten) the time that is required to process data from the actual physical media, other parts of your I/O processing also use valuable time and affect storage access latency.

This chapter provides an introduction to IBM zHyperLink technology and use cases to demonstrate the benefits of the zHyperLink in your enterprise. Also included in this chapter are the hardware and software prerequisites that must be met for a successful implementation.

This chapter includes the following topics:

- ▶ 1.1, “Introduction to zHyperLink” on page 2
- ▶ 1.2, “Use cases” on page 3
- ▶ 1.3, “zHyperLink prerequisites” on page 5
- ▶ 1.4, “Additional considerations” on page 10

1.1 Introduction to zHyperLink

When an I/O request is performed by an application, several operations are performed on the pathway that are required to complete the request. All of these processes and devices on the storage pathway contribute to the overall I/O latency. The I/O latency can be a significant part of application response time; for example, an I/O service time can represent up to 65% of the latency for an IBM Db2 transaction.

zHyperLink technology is designed to reduce the I/O latency by providing a fast, reliable, and direct communication path between the CPU and the I/O device (backend storage).

zHyperLink is intended to speed up Db2 for z/OS transaction processing and improve active log throughput. This goal is accomplished by installing zHyperLink adapters on the z/OS host (IBM z14®, and IBM z15™), and select IBM Storage (IBM DS8880 or IBM 8900F) hardware, and connecting them using zHyperLink cables. This configuration creates a point-to-point connection between the CPU and I/O device (no intermediary switching devices), which reduces the I/O response time by up to 10 times compared to IBM z High-Performance FICON® (zHPF). Such low response time is achieved by using synchronous I/O requests.

Synchronous I/O requests reduce the time that is required for some functions, such as I/O interrupts and z/OS dispatch time. The difference between synchronous (sync) and asynchronous (async) I/O is explained next.

1.1.1 Synchronous versus asynchronous I/O

Asynchronous I/O (async I/O) is a form of input/output processing that permits other processing to continue before the transmission on the storage pathway completes. When the operating system processes a block storage request, the CPU is released to perform other tasks, while the hardware device completes the storage operation.

Standard I/O processing that is available, such as zHPF, requires I/O operations to perform a series of time-consuming tasks, including the following examples:

- ▶ z/OS dispatching
- ▶ Interrupt handling
- ▶ CPU queue time
- ▶ Reload L1/L2 processor cache

These tasks and others that are required for I/O processing cause the I/O response time to be relatively long compared to transferring data within virtual storage, with response times of 130+ microseconds plus interrupt handling and CPU dispatch time.

Synchronous I/O (sync I/O), means that the entire path that handles an I/O request stays within the process context that started the I/O. When synchronous I/O is performed, the CPU waits (or “spins”) until the I/O is completed, or the timeout value is reached.

zHyperLink can significantly reduce the time that is required to complete the I/O because the dispatching, interrupt handling, CPU queue time, and CPU cache reload activities are no longer necessary. In addition to the I/O response time improvement, more value for zHyperLink is realized in that the I/O remains connected; therefore, no z/OS redispatch time exists.

To achieve shorter response times, synchronous I/O read operations are performed on random 4 KB read requests when the data is in the DS8000 cache. If the data is not in the storage cache, a notification is returned to the host, and the standard (traditional) asynchronous I/O (read) is performed. In the meantime, the requested information is pre-fetched, which might reduce the response time for the asynchronous I/O operation.

For writes, DS8000 cannot meet the latency requirements of a random write because too much overhead exists in the DS8000 to acquire lock tracks, assign NVS space, and so forth. Therefore, only writes that follow a log-like pattern are supported.

Records can be written more than once; however, they are always going forward in the data set. When the end of the data set is reached, writes can wrap around to the beginning. However, IBM DB2® does not wrap, it starts using a new active log data set.

zHyperLink write operations require that z/OS establishes a zHyperLink session with the IBM DS8000®, which consists of issuing a special channel program to establish zHyperLink write access. The channel program returns a token that is used for subsequent I/Os to that data set and causes the DS8000 to assign NVS space and set up descriptors to move data.

DS8000 also pre-locks the next set of tracks that are written to (write set). As zHyperLink writes are issued, the DS8000 releases tracks as they are completed and locks more tracks to stay ahead of the I/O requestor. This process is done to ensure that when the zHyperLink write occurs, it can be satisfied immediately. Contrast this with reads, which lock tracks on-demand and only for the time to perform the I/O. If for any reason the I/O cannot be completed, it is rejected and is retried by using FICON.

1.2 Use cases

You can benefit from the use zHyperLink in various situations. In this section, we provide several examples that show how the implementation of zHyperLink can introduce new business opportunities or reduce your application design complexity.

1.2.1 Avoiding financial losses

Mainly in the financial sector, fraud attempts that occur daily affect a broad range of businesses. Insurance, government agencies, and financial services are the main targets of fraud activities, which makes security a high priority for those companies. With such relevant risk to the core business, every transaction must be checked and validated. In this case, time matters.

Data mining applications and statistics help to anticipate and quickly detect fraud, take immediate action to minimize losses, and reduce risks. However, those efforts can increase response time. If the I/O does not complete within the time that is expected by the core business application, it might abandon the validation process, which increases the risks of processing fraudulent operations.

zHyperLink enables much faster read and write operation with synchronous I/O and low latency access to the databases with fraud detection rules. This ability allows an effective fraud detection process with high I/O performance.

1.2.2 Growing business opportunities

With numerous channels available to create business every day, industries cannot miss a new opportunity to offer more services to their clients. The process of adding or improving services must be instantly efficient. The number of transactions in e-commerce increased significantly and the reason why it increased also changed. Because more customers are influenced by online advertising, that advertising is structured with fresh data from the customers (feedback), which makes it that much more effective.

To help offer faster and smarter services, zHyperLink improves I/O response time significantly. This ability allows companies to process transactions much faster, which gives them enough time to process data from the clients to gain more insights and deliver more and improved services or finer targeted advertisements.

Providing these services or advertisements creates business opportunities. For example, a financing company might use zHyperLink advantages to show more favorable mortgage refinancing options to a client while they are checking their account balance without affecting response time.

1.2.3 Better client experience and revenue

Poor load speeds and inadequate performance on online services usually lead users to not wait for a transaction to complete, which affects critical financial business and negatively affects your brand. Clients who are dissatisfied with online services might take their business elsewhere. As a result, you are not losing only one customer, but possibly a customer whenever your company's performance is affected. This situation can worsen during peak time periods, which limits and compromises your marketing campaign.

Driven by market requirements, zHyperLink delivers the leading I/O latency for transaction processing on IBM Db2® for z/OS applications. It also improves customer satisfaction, which often generates revenue and new business opportunities.

1.2.4 Faster recovery from outages

Hardware and software outages usually drive recovery processes to run. Recovery is not considered complete while every pending transaction in the work queue is not fully processed.

zHyperLink enables a synchronous I/O for that operation and allows the work queue to be reduced much more quickly than heritage FICON I/O. This ability significantly improves the Mean Time to Recovery (MTTR).

1.2.5 Workload growth without redesigning applications

The growth in the amount of data that is generated by mobile, social media, or analytics applications, augmented security requirements, and cloud usage drive new requirements to run transactions with improved level of service and an enterprise class quality of service.

Applications must constantly be redesigned to avoid performance degradation, while the core business must remain in service always. However, any type of development or redesign work can be time-consuming and costly to the business.

Improving I/O latency mitigates the need to reengineer applications to scale up as the transaction rates and business grow. zHyperLink provides lower risk than application development (redesign or reengineer) to achieve better response times and keep business running.

1.2.6 Reducing data sharing overhead

Concurrent read and write operations can be done from different applications that are accessing the same data. Better I/O latency allows clients to save money by continuing to grow and add data sharing instances. When the I/O operation completes faster, it reduces the possibility of multiple accesses to the data, which mitigates or delays system upgrades.

1.3 zHyperLink prerequisites

Before you plan and deploy zHyperLink in your environment, you must understand the basic prerequisites that must be satisfied before zHyperLink is enabled. The hardware and software prerequisites are described next.

1.3.1 Hardware prerequisites

This section provides information about the hardware requirements for zHyperLink. The IBM z14 Central Processor Complex (CPC), IBM z15 Central Processor Complex, zHyperLink connectivity, and the DS8880 and DS8900F storage subsystems (the storage control units) are described.

Naming: Unless specific model information is presented, we use the generic names *IBM Z CPC* for supported IBM Z models and *DS8000* for supported DS8880 or DS8900F models in the remainder for this chapter.

IBM Z hardware and firmware

zHyperLink is a point-to-point connection between the zCPC and the Storage Controller, which provides extreme low latency for random reads and Db2 log writes. A sample zHyperLink connection between a IBM Z CPC and a DS8000 Storage Control Unit, which is limited to a distance of 150 meters (492 feet), is shown in Figure 1-1.

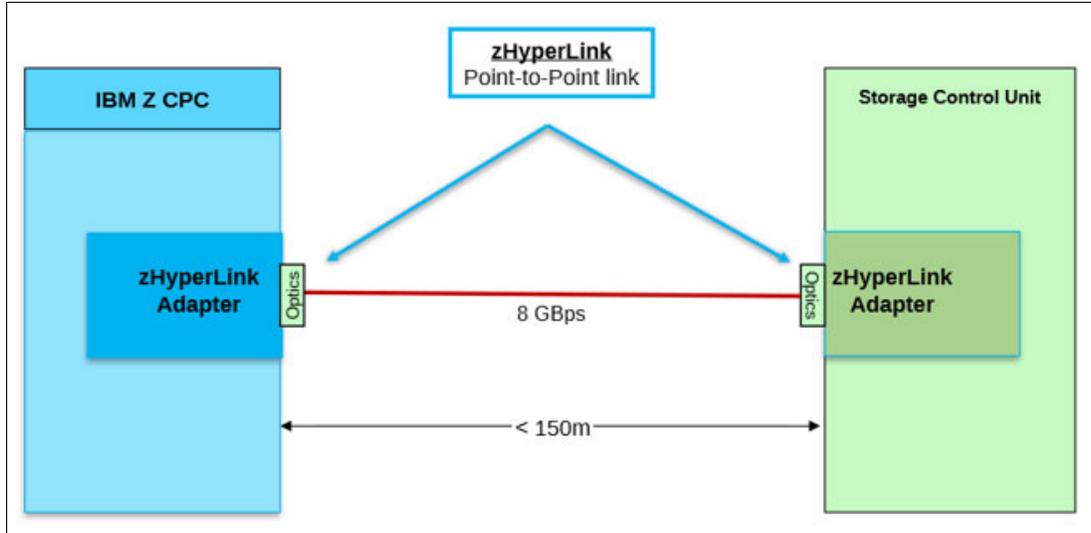


Figure 1-1 zHyperLink physical connectivity

zHyperLink Express includes the following hardware requirements:

- ▶ IBM z14, IBM z14 ZR1 with zHyperLink Express adapter (FC #0431), and Driver 36C at its Minimum Ship Level (MSL)
- ▶ IBM z15 models T01 and T02 with zHyperlink Express 1.1 adapter (FC #0451) and Driver 41C MSL
- ▶ FICON channel as a driver. At least two 24x MTP-MTP cables to connect to the Storage Controller (DS8880 or DS8900F); that is, two zHyperLink links.
- ▶ DS8880 with I/O Bay Planar board and firmware level 8.3 or later
- ▶ DS8900F with I/O Bay Planar board and firmware level 9.0 or later
- ▶ DS8880 or DS8900F zHyperLink adapters

The zHyperLink adapters (FC 0431 for IBM z14 and FC 0451 for IBM z15) are installed in the Peripheral Component Interconnect® Express (PCIe) I/O drawer.

The zHyperLink Express features directly connects the IBM z14 and IBM z15 Central Processor Complex (CPC) to the I/O Bay enclosure of the DS8000. The IBM zHyperLink Express is a two-port feature that is installed in the PCIe I/O drawer. Up to 16 features with up to 32 zHyperLink Express ports are supported in an IBM z14 or IBM z15 CPC.

The zHyperLink Express feature uses PCIe Gen3 technology, with x16 lanes that are bifurcated into x8 lanes for storage connectivity. It is designed to support a link data rate of 8 GB per second (GBps).

Note: The link data rates do not represent the link performance. The performance is dependent upon many factors, including latency through the adapters, cable lengths, and the type of workload.

On the IBM Storage Controllers, the fiber optic cable connects to a zHyperLink PCIe interface in I/O Bay enclosure. Each zHyperLink connection requires a zHyperLink I/O adapter to connect zHyperLink cable to the storage system. Each zHyperLink I/O adapter features one port.

For availability, the adapters must be ordered in sets of two. Depending on the system configuration, the zHyperLink I/O adapters for your storage system can be ordered in quantities of 4, 6, and 12 (see Table 1-1 on page 8).

For availability and performance, zHyperLink connections are required to be configured in pairs. zHyperLink connections work together with FICON zHPF channels; they do not replace them. FICON channels are still required to “drive” zHyperLink to each storage system. FICON is required for IPL and the following components:

- ▶ Initialization of zHyperLink connection
- ▶ I/Os that are not eligible for zHyperLink
- ▶ Fallback when a zHyperLink request fails (for example, cache miss and busy condition)

A 24x MTP-MTP cable is required for each port of the zHyperLink Express feature. It is a single 24-fiber cable with Multi-fiber Termination Push-on (MTP) connectors. Internally, the single cable houses 12 fibers for transmit and 12 fibers for receive.

The following fiber type options are available with specifications that support different distances for zHyperLink Express:

- ▶ Up to 150 meters (492 feet): OM4 50/125 micrometer multimode fiber optic cable with a fiber bandwidth wavelength of 4.7 GHz-km @ 850 nm.
- ▶ Up to 100 meters (328 feet): OM3 50/125 micrometer multimode fiber optic cable with a fiber bandwidth wavelength of 2.0 GHz-km @ 850 nm.

Note: The minimum requirement for zHyperLink Express connectivity between the IBM z14 or IBM z15 CPCs and the DS8880 or DS8900F subsystems is two links.

1.3.2 Storage systems zHyperLink support

This section lists the IBM Storage systems that support IBM zHyperlink.

DS8900F 9.0 or later (inherited support from DS8880 R8.3.1)

The following features are supported:

- ▶ Db2 V11 and V12 Read Support (batch and transactional workloads)
Synchronous Db2 database reads: 4 K CI¹/page sizes
- ▶ VSAM Read Support (batch and transactions):
 - APAR OA52941 (fix available as of April 29, 2019)
 - All VSAM record type data sets (that is, KSDS, ESDS, RRDS, and VRRDS) support zHyperLink, along with all types of access (except NSR Sequential)
- ▶ Ability to turn zHyperLink on and off at the data set level
By way of Storage Class policy or the **VARY SMS** command

¹ CI - VSAM Control Interval

DS8900F 9.0 or later (inherited support from DS8880 R8.5.1)

Db2 active log features the following support:

- ▶ zHyperLink write exploitation (supports Metro Mirror with zHyperWrite and full duplex)
- ▶ Fixes for APAR PH05030 (available since January 2019)
- ▶ Fixes for z/OS APAR OA56575 (available since March 2019)

The number of zHyperLink ports, cores per node, and available processor memory on DS8900F models are listed in Table 1-1.

Table 1-1 zHyperLink connections, cores, and memory available on DS8900F models

DS900F system models	IBM POWER® 9 cores per DS8900F system (min/max)	Processor memory	Max zHyperlink connections (increments of 2)
DS8910F Rackless	16/16	FC 4450 - 192 GB FC 4451 - 512 GB	4
DS8910F Racked	16/16	FC 4450 - 192 GB FC 4451 - 512 GB	4
DS8950F	20/40	FC 4452 - 512 GB	6
		FC 4453 - 1024 GB FC 4454 - 2048 GB FC 4455 - 3456 GB	12
DS8980F	44	FC 4456 - 4352 GB	12

Use the feature codes that are listed in Table 1-2 to order zHyperLink I/O adapters for DS8900F Storage Systems.

Table 1-2 DS8900F zHyperLink related Features

Feature code	Feature	Description
1303	Gen3 I/O enclosure pair	An I/O enclosure pair supports: <ul style="list-style-type: none"> ▶ Up to two HPFE Gen2 adapter cards ▶ Up to two HPFE Gen2 enclosure pairs ▶ Up to four zHyperLink adapters ▶ Up to eight Host Adapters
1450	zHyperLink cable	<ul style="list-style-type: none"> ▶ 40m (131 ft) OM4 50/125 micrometer, multimode MTP connectors ▶ Requires FC 3500 to be present in the storage system
1451	zHyperLink cable	<ul style="list-style-type: none"> ▶ 150 m (492 ft), OM4 50/125 micrometer, multimode, and MTP connectors ▶ Requires feature code 3500 for storage systems
3500	zHyperLink I/O-adapter	<ul style="list-style-type: none"> ▶ Provides a single port zHyperLink adapter ▶ Orderable in increments of two ▶ Up to eight zHyperLink adapters are supported in a single frame system ▶ Up to 12 zHyperLink adapters are supported in a two frame system (996 and 998 models)

For more information about zHyperLink features, see the following publications:

- ▶ *IBM Z Connectivity Handbook*, SG24-5444
- ▶ *IBM z14 Technical Guide*, SG24-8451
- ▶ *IBM z15 Technical Guide*, SG24-8851
- ▶ *IBM DS8900F Version 9 Introduction and Planning Guide*, SC27-9560

zHyperLink Express is supported on DS8880 Storage systems with 12-core and more processors. DS8884 with six cores can be concurrently upgraded to 12 cores with feature code (FC) #4425.

DS8880 with I/O Bay Planar board and firmware level 8.3 or later plus the z-synergy Services license is required. Also, I/O Priority Manager must be disabled. The DS8880 microcode level must be at least 8.3.1 or later.

The DS8900F requires microcode level at R9.0 or later. The z-synergy Services license is required, and I/O Priority Manager must be disabled.

1.3.3 Software requirements

zHyperLink includes the following software requirements:

- ▶ zHyperLink Express is supported on z/OS 2.3² or later. Plan for required z/OS software maintenance.
- ▶ The z/OS image must run in an LPAR, not as a guest under IBM z/VM®.
- ▶ zHyperLink requires IBM z High-Performance FICON (zHPF) to be enabled.
- ▶ The minimum required version for IBM Db2 is version 12. Plan for required Db2 software maintenance.

Note: Db2 V11 supports zHyperLink for Synchronous database reads – 4 K Cl/page sizes (Batch and Transactions) only.

zHyperLink supports Db2 random 4 KB reads and writes. As of this writing, only Db2 log data set writes are supported. VSAM read support is provided with APAR OA52941. All VSAM record type data sets support zHyperLink.

z/OS V2R2 and later, with APARs, support for zHyperLink Write with GM primary volume support is provided. (MGM/MTMM also is supported when MM uses zHyperWrite). The following software levels are required:

- ▶ z/OS 2.2 (HBB77A0) - KA56723 RW20109
- ▶ z/OS 2.3 (HBB77B0) - BA56723 RW20109
- ▶ z/OS 2.4 (HBB77C0) - CA56723 RW20109

In addition to software levels, updates to z/OS and Db2 system parameters are required for zHyperLink enablement and monitoring. For more information about maintenance and parameter requirements, see 2.4, “Software planning” on page 36.

² At the time of this writing, z/OS V2R2 is at end of support (EoS).

1.4 Additional considerations

In this section, the following considerations that are related to zHyperlink implementation, such as limitations and the use potential, are described:

- ▶ Site considerations
- ▶ IBM ECKD
- ▶ DS8000 I/O Priority manager
- ▶ IBM FICON Utilization
- ▶ z Batch Network Analyzer (zBNA)

1.4.1 Site considerations

zHyperLink hardware is designed for short distance communication of up to 150 meters (492 feet). Therefore, your Storage Controller must be connected up to this (cable) distance from your CPC.

zHyperLink is a point-to-point connection (IBM Z CPC to DS8000 controller); no switching is supported. The distance and point-to-point requirements along with your site availability strategy play a key role in implementing zHyperLink. For more information about planning zHyperlink implementation, see Chapter 2., “Planning” on page 15.

Each customer might use a different site configuration that is based on application and business requirements. For some clients, it might be suitable to use a single data center location while other business might be required by law to use multiple production and disaster recovery sites to ensure high availability and prevent data loss.

In this IBM Redbooks publication, we consider the following data center scenarios and describe how the applications can benefit from zHyperLink in each scenario:

- ▶ Single site
- ▶ Multi-site
- ▶ Multi-site with extended distance

These scenarios are described next.

Single-site

A single-site basic mainframe configuration features all production data and processing units in the same physical location. In this scenario, a secondary copy of your data might exist. The secondary DASD controller also can be attached to the z/OS host by zHyperLink if the distance between the machines is less than 150 meters (492 feet).

A single site configuration with a secondary DASD controller is shown in Figure 1-2.

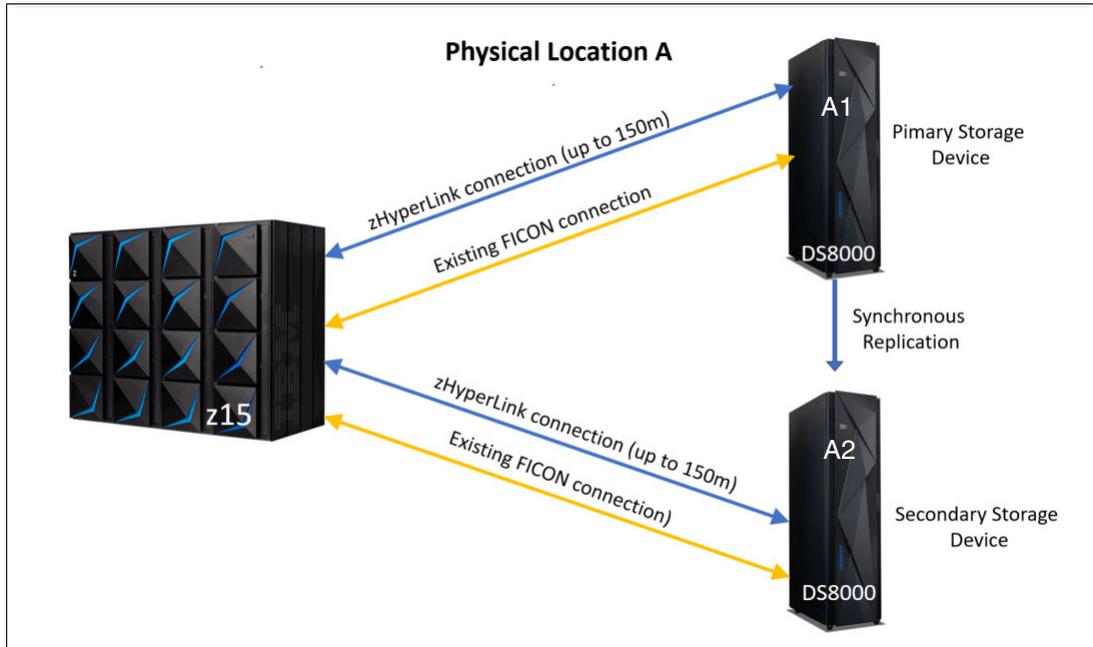


Figure 1-2 Sample single site configuration

For more information about the implementation of zHyperLink in a single-site environment, see Chapter 2., "Planning" on page 15.

Multi-site with synchronous storage replication

Many customers find that a multi-site configuration provides many benefits to their business by allowing the workload to be spread across multiple sites, or by providing disaster recovery capabilities that allow near zero downtime or data loss.

These sites might be up to 10 km (6.21 miles) apart, which allows the use of synchronous write operations. In such scenarios, the benefits of zHyperLink depend on how your system is designed.

A two-site configuration with IBM HyperSwap® capabilities is shown in Figure 1-3.

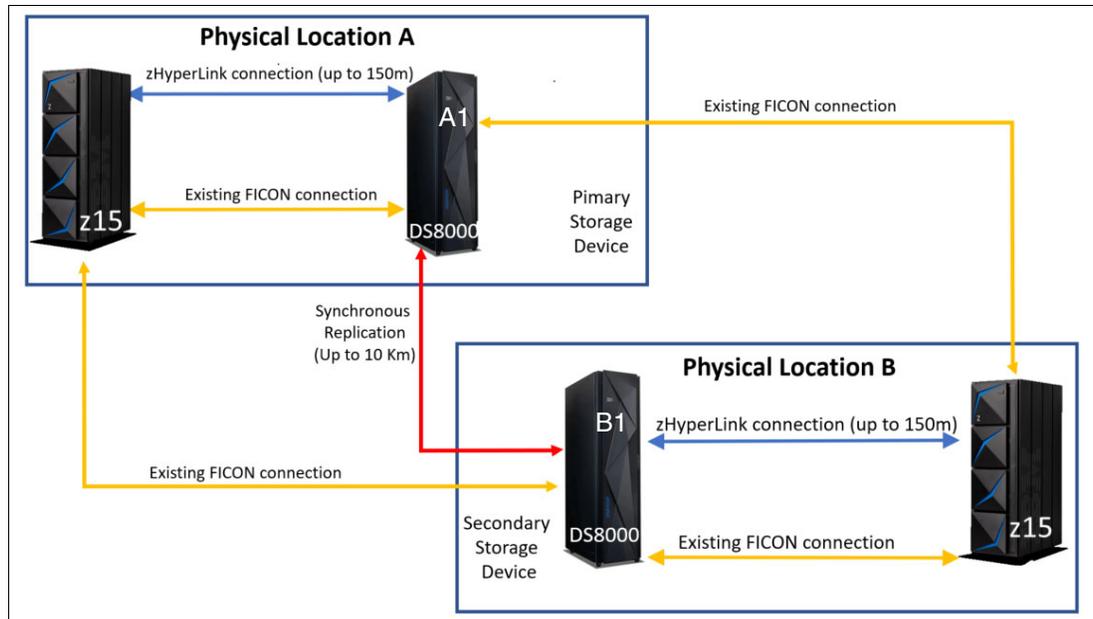


Figure 1-3 Sample two site configuration

For more information about the implementation of zHyperLink in a multi-site environment, see Chapter 2., "Planning" on page 15.

Multi-site with extended distance

Other customers might require extended distance recovery capabilities, including situations where a large area can be affected by events, such as earthquakes or flooding. These companies often opt to use a multi-site configuration, which involves synchronous and asynchronous replication, which results in improved system performance while maximizing system availability and resilience.

A two-site configuration with synchronous replication within each site, and asynchronous replication between sites, is shown in Figure 1-4.

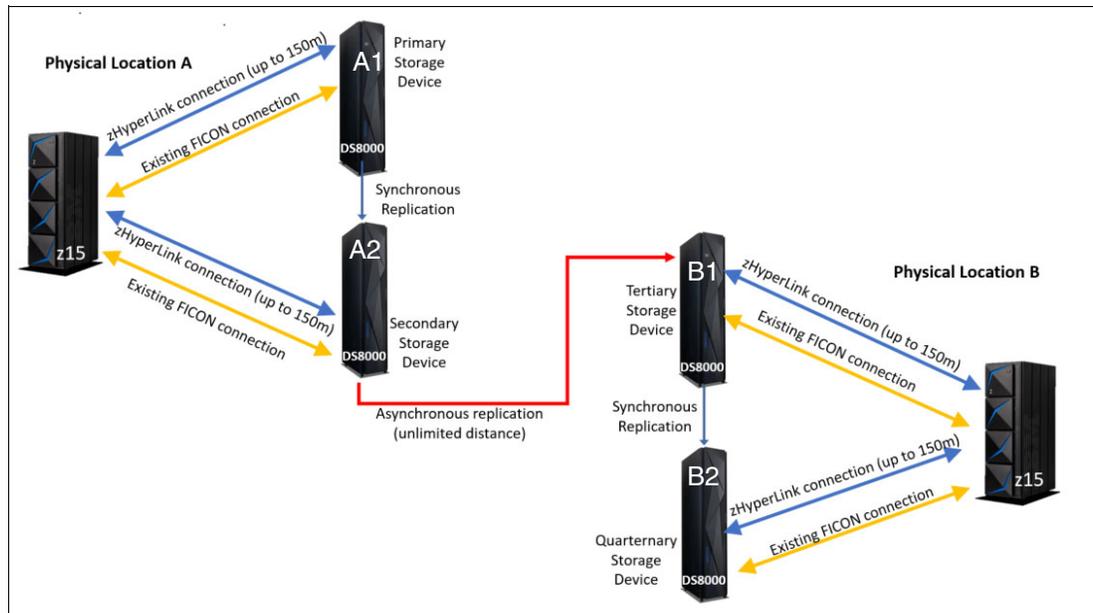


Figure 1-4 Sample multi-site configuration with extended distance

For more information about the implementation of zHyperLink an extended distance multi-site environment, see Chapter 2., “Planning” on page 15.

1.4.2 ECKD

zHyperLink supports only the traditional mainframe Extended Count Key Data (ECKD). The transport protocol is defined for accessing ECKD data records.

1.4.3 DS8000 I/O priority manager

The DS8000 I/O Priority Manager prioritizes access to system resources to achieve the wanted quality of service that is based on defined performance goals (high, medium, or low) for a volume or a single I/O request. The Priority Manager constantly monitors and balances DS8000 system resources to help applications meet their performance targets automatically, without operator intervention.

DS8000 I/O Priority Manager works with z/OS Workload Manager to add granularity that enables the I/O prioritization at single I/O operation level. When enabling zHyperLink on the DS8000, I/O Priority manager must be turned off at the disk subsystem level. This feature is turned off for all data, even if the data is not eligible for zHyperLink. Therefore, I/O prioritization for IBM IMS data (and other data) might be affected.

1.4.4 FICON I/O workload

Considering that IBM zHyperLink does not replace zHPF but works in tandem with it instead, the workload that is transferred by zHPF reduces when zHyperLink is implemented; however, zHyperLink Express does not reduce the physical number of current zHPF connections.

An example in which zHPF is the only communication protocol handling 100% of I/Os is shown in Figure 1-5.

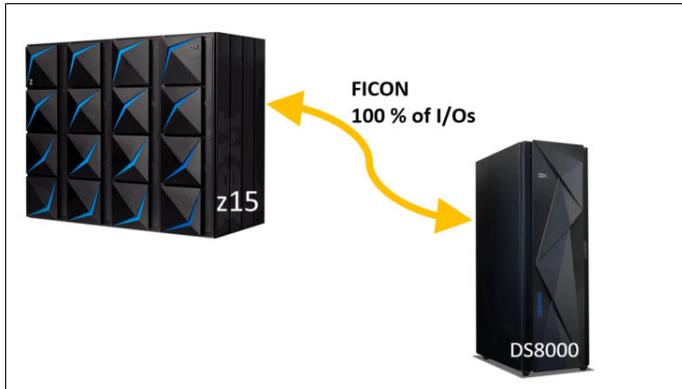


Figure 1-5 zHPF is the only communication protocol handling 100% of I/Os

How traffic on zHPF was directly affected after enabling zHyperLink is shown in Figure 1-6.

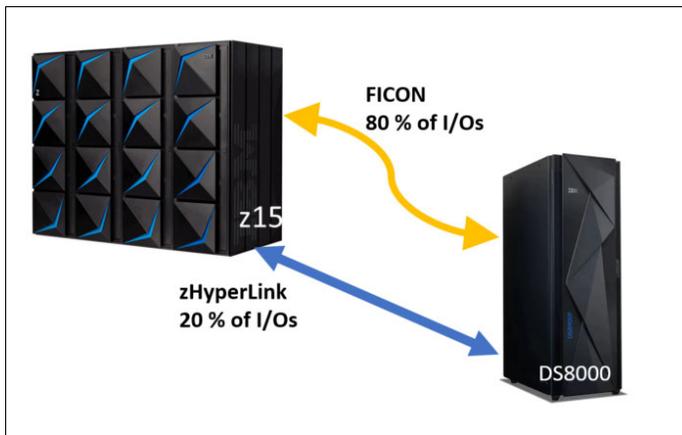


Figure 1-6 How traffic on zHPF was directly affected after enabling zHyperLink

Note: The numbers that are shown in Figure 1-5 and Figure 1-6 are for example purposes only.



Planning

Every business industry has their own IT workload requirements. Some applications might be designed to make heavy use of database read operations, while others require many inserts or updates to data files. Some workloads might be a mixture of both scenarios. These particularities must be assessed before zHyperLink is implemented.

Understanding your business requirements and how your applications are designed is vital to gain the most benefit from a zHyperLink implementation.

In this chapter, we describe the necessary considerations when planning for zHyperLink.

This chapter includes the following topics:

- ▶ 2.1, “Identifying zHyperLink eligible workloads” on page 16
- ▶ 2.2, “Sizing synchronous I/O for zHyperLink” on page 27
- ▶ 2.3, “zHyperLink hardware planning” on page 29
- ▶ 2.4, “Software planning” on page 36
- ▶ 2.5, “Performance planning” on page 41
- ▶ 2.6, “Monitoring” on page 44

2.1 Identifying zHyperLink eligible workloads

The first issue to consider when planning for zHyperLink is how your critical applications perform I/O operations. zHyperLink is not a substitute for other I/O protocols, such as z High Performance FICON (zHPF) because only a subset of all I/O operations is eligible for zHyperLink processing.

At the time of this writing, the following workloads use zHyperLink to perform I/O operations.

- ▶ Db2 database reads
- ▶ VSAM data set reads
- ▶ Db2 sequential log writes

Note: If your environment is identified as eligible for zHyperLink usage, and the DS8000 I/O Priority Manager is enabled, it *must* be disabled before zHyperLink can be enabled. In such cases, a further analysis of possible effects on applications that do not benefit from zHyperLink also is necessary.

By providing low latency disk access, zHyperLink improves application response time. This improvement reduces I/O sensitive workload overall response time by up to 50% without requiring application changes.

The improved performance of zHyperLink allows the processor to make a synchronous request for the data that is in the DS8000 cache. This feature eliminates the undispach of the running request, the queuing delays to resume the request, and the PU cache disruption.

The following situations benefit most from zHyperLink enablement:

- ▶ Db2 random reads

Db2 workloads are between the applications that can achieve extreme low latency times when zHyperLink is used. By eliminating time-consuming tasks, such as undispatching the running process, CPU queuing, and cache reload, you can achieve up to 10x better response time on cache hit read requests.

- ▶ VSAM read requests on cache

You can also configure your system to use zHyperLink for VSAM read processing. zHyperLink can be used for all types of VSAM access (NSR, LSR, and RLS). Set your VSAM data sets that require low latency and perform many read I/Os to use zHyperLink.

- ▶ Db2 sequential writes

zHyperLink feature also supports Db2 log writes after it follows a log-like pattern and writes and made sequentially. Random writes are not supported because they are not good candidates for zHyperLink.

Although zHyperLink can provide huge performance benefits to some types of I/O operations, cases exist in which its benefits are limited. The following examples are not good candidates for zHyperLink:

- ▶ Read requests that are not likely to find the data in the DASD cache:

Because zHyperLink is designed to provide low response times, only data that is stored in the DASD cache is eligible for zHyperLink I/O processing. If your application data is not likely to be on DASD cache or the cache hit ratio is below 80%, the benefits from zHyperLink can be limited.

- ▶ Read ahead or prestaging of data:

When read ahead or prestaging are performed, several I/O requests are sent to the DASD controller before the data is required by the application. The extended wait time that is related to the multiple I/O requests result in an extended CPU spinning time while the operations are being processed. For that reason, read ahead and prestaging of data are not good candidates for zHyperLink processing.

- ▶ Random Writes:

Before a zHyperLink write operation can be done to a data set, z/OS must establish a zHyperLink write session with the DS8000 for that data set. Establishing a zHyperLink session is a special channel program that returns a token that enables zHyperLink write access to the DS8000 and is used for all subsequent I/Os for that specific data set. It causes DS8000 to pre-lock the next set of tracks to ensure that zHyperLink write can be satisfied immediately. This process works well with sequential writes; however, DS8000 cannot meet zHyperLink latency requirements of a random write. For that reason, random writes are not good candidates for zHyperLink.

2.1.1 zHyperLink in site configurations overview

As described in 1.4, “Additional considerations” on page 10, different site layouts also can affect the applicability and potential performance benefits of the zHyperLink implemented configuration.

In this section, we describe three different site scenario examples and how they can benefit from zHyperLink.

2.1.2 Single-site

Single-site configurations are the simplest configurations. In this scenario, all computing and storage devices are in a single physical location. No DASD replication to other remote sites occurs (synchronous nor asynchronous).

In a single site configuration, a secondary DASD subsystem for replication and high availability can be included if a DASD subsystem failure occurs. zHyperLink can provide great benefits to your system in single site configurations. By connecting your IBM Z Central Processor Complex (CPC) to the primary and secondary DASD subsystems by using zHyperLink, you can reduce eligible I/O response time latency by up to 10x for eligible operations.

An example of implementation in a single site configuration with a primary and secondary DASD subsystem is shown in Figure 2-1 on page 18. In this example, both DS8000 subsystems are connected by using zHyperLink, which enables the use of zHyperLink if a planned or unplanned DASD subsystem failover to secondary occurs.

This configuration supports zHyperLink reads and writes. zHyperWrite must be enabled if zHyperLink writes are required.

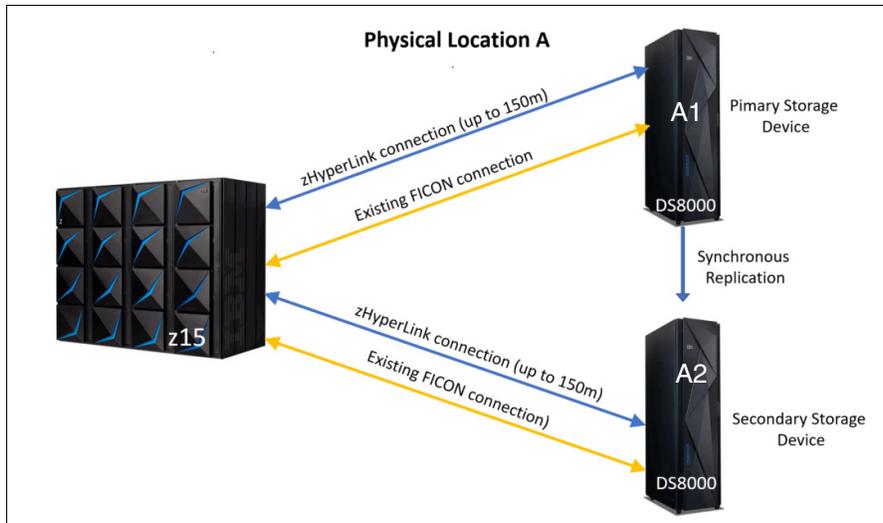


Figure 2-1 Sample single site configuration

Note: In the configuration that is shown in Figure 2-1, both DASD subsystems must be no more than 150^a meters (492.12 feet) from the CPC.

a. Direct connection to storage system, maximum fiber cable length (OM4 fiber)

2.1.3 Multi-site with synchronous replication

To achieve higher levels of system availability and reduce recovery time (in case of a site failure) to near zero, customers might run their systems in more than one physical location (site). The data also must be replicated between sites synchronously (see Figure 2-2). Although longer distances are supported¹, these sites are apart from each other usually by no more than 10 kilometers (6.21 miles).

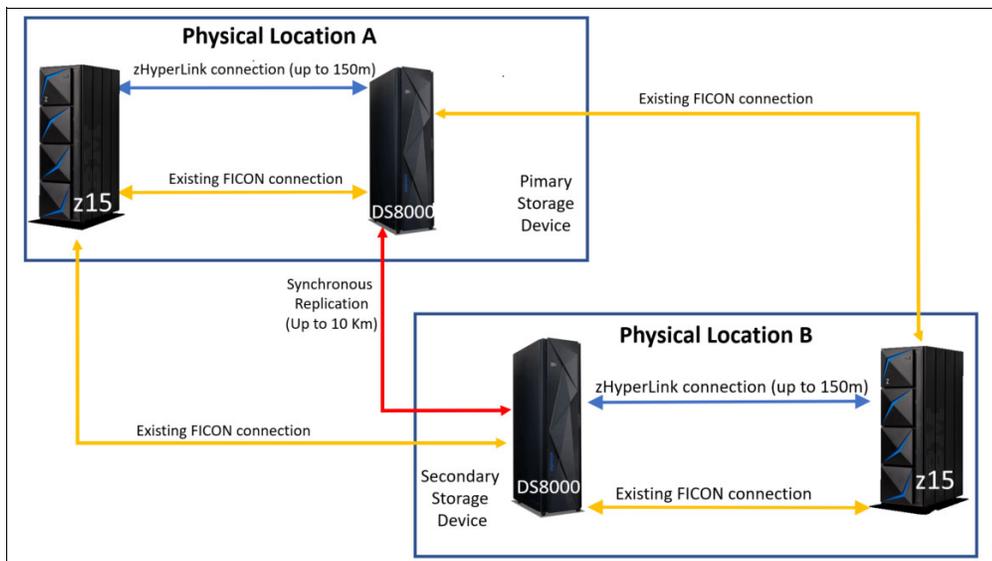


Figure 2-2 Multi-site configuration with synchronous replication

¹ When using DWDM (Dense Wave Division Multiplexing) distance extenders in supported configurations

A sample multi-site implementation in which each CPC is connected to the DASD subsystem in the same site by zHyperLink and FICON adapters, and to the remote site that is existing FICON only technology, is shown in Figure 2-2 on page 18.

Note: The configuration that is shown in Figure 2-2 on page 18 supports zHyperLink read operations only. zHyperLink write with synchronous replication requires the distance between the CPC and both storage subsystems to be up to 150 meters (492.12 feet), as shown on Figure 2-1 on page 19.

Because zHyperLink is designed to work within a range of 150 meters (492.12 feet), the distance between the sites (less than 150 meters) prevents the remote CPC from being connected to the primary and secondary DASD subsystems by zHyperLink connections. Therefore, only the CPC that is in the same physical location with the production DASD subsystem (more than 150 meters) can be connected by using zHyperLink.

zHyperLink read considerations

Such a configuration might affect the performance of zHyperLink *read* I/O because only the workload that is running in the local site benefits from lower I/O response times. Also, this configuration can result in uneven application read response times for dual site workloads or disaster recovery. In normal situations, this configuration does not affect zHyperLink read eligibility.

zHyperLink write considerations

zHyperLink write operations are supported only when primary and secondary DS8000s are within 150 meters (492.12 feet) from the CPC because zHyperLink connections to both subsystems must exist.

The reason for the requirement of having zHyperLinks to both primary and secondary DS8000 is that the DS8000 firmware overhead for mirroring the data is too high for zHyperLink.

Also, zHyperWrite must be enabled and the write operations performed synchronously by the zHyperLink connections to both DASD subsystems (primary and secondary). Write operations are never be possible with zHyperLink if primary or secondary storage is not within supported distance from its associated IBM Z CPC.

You can configure your environment so that the workloads that benefit most from zHyperLink run on the CPC that is connected to the primary DASD subsystem by using zHyperLink adapters. You can plan to direct your zHyperLink eligible batch workload to the CPC closer to the primary DASD subsystem, if the resources satisfy your batch needs.

2.1.4 Consistent read from secondary storage with synchronous replication

Note: In this section, *Primary* and *Secondary* refer to storage system or volume, depending on the context. When referring to the physical aspect (processor or link), it is primary storage and secondary storage. When referring to the logical aspect (that is, z/OS HyperSwap), it is primary volumes and secondary (target) volumes).

Consistent Read from Secondary (CRS) is a performance improvement in a z/OS Metro Mirror (MM) environment.

Single-site with HyperSwap considerations

In a single site workload with Metro Mirror configuration (see Figure 2-1 on page 18), the host processor and primary storage (A1) are in the same site (≤ 150 meters - fiber length) with the secondary storage (A2), which is at some distance from the host processor.

Under normal conditions, all read and write operations are performed with the primary storage, while writes are mirrored to the secondary storage (Metro Mirror A1 \rightarrow A2). Because the primary storage is within 150 meters (492.12 feet) of the processor (fiber length), zHyperLink can be used for eligible read operations.

After a HyperSwap is performed and the mirroring direction is reversed (Metro Mirror A2 \rightarrow A1), the secondary storage is now active for I/O to the processor. The primary is at distance, which means that reads and writes must travel over distance and zHyperLink cannot be used. I/O latency improves if read operations can be issued to the secondary storage (A1).

Multi-site considerations

Another use case is in a multi-site workload configuration where host processors exist in both sites that are accessing the storage systems that are in MM relationship (see Figure 2-2 on page 18).

The processor at site one can perform all of the read/write I/Os by using the local disk. However, the I/Os from second site must travel over a distance to site one. I/O latency improves if read operations can be issued to the secondary.

Consistent reads

Consistent read means that the data is read from the secondary copy and it is the same as if it was read from the primary storage. The following prerequisites must be met for this condition to be true:

- ▶ The application or middleware does not rely on behavior within the storage system to serialize concurrent read and write requests. That is, the application or middleware includes some type of serialization mechanism to ensure that read and write requests to the same set of tracks do not occur concurrently and request the bypass extent checking attribute when issuing their I/Os. This process must be transparent to applications that use Db2, IMS, or VSAM directly or indirectly, because many of the I/Os that are issued by these components \ specify bypass extent checking.

Note: Bypass extent checking is a prerequisite for data sets that use zHyperLink or zHyperWrite.

- ▶ The MM pairs are in full-duplex state; that is, the primary and secondary are identical and kept in sync, and must be part of a HyperSwap configuration.

Consistent read from Metro Mirror secondary

With DS8000 Licensed Machine Code (LMC) R9.2, IBM introduces consistent read from the Metro Mirror secondary site. This capability provides improved performance for read operations from a z/OS system where the Metro Mirror secondary includes closer or better connectivity than the Metro Mirror primary.

This capability also enables reads for supported IO types to be performed from a Metro Mirror secondary volume over FICON and zHyperLink, which delivers reduced latency and improved workload balance compared to reading from the primary only.

Consistent read from Metro Mirror secondary is supported for Metro Mirror and multi-target Metro Mirror, and the Metro Mirror relationship in three-and four-site Metro Global Mirror configurations.

This consistent read from Metro Mirror requires that the Metro Mirror environment is HyperSwap-enabled with Copy Services Manager or GDPS and that the z/OS system includes the required software support installed. Supported IO types include Db2 and Virtual Storage Access Method reads, and require that the software supports the Bypass Extent Serialization function.

For more information, see Appendix E, “Consistent Read from Metro Mirror Secondary” on page 127, and IBM Hardware Announcement [121-039](#).

2.1.5 Multi-site with extended distance and asynchronous replication

Some enterprises require the highest levels of system availability in specific situations; for example, a site or hardware device failure, or a natural disaster that can affect a large area. Such high levels of service availability are achieved by creating a multi-site configuration with local and remote copies to ensure short recovery time objective (RTO) and minimize data loss (recovery point objective [RPO]).

A sample implementation of a multi-site configuration with extended distance between sites is shown in Figure 2-3. This configuration enables the full benefits from zHyperLink at the primary (Physical Location A) and secondary (disaster recovery - DR, physical location B) sites.

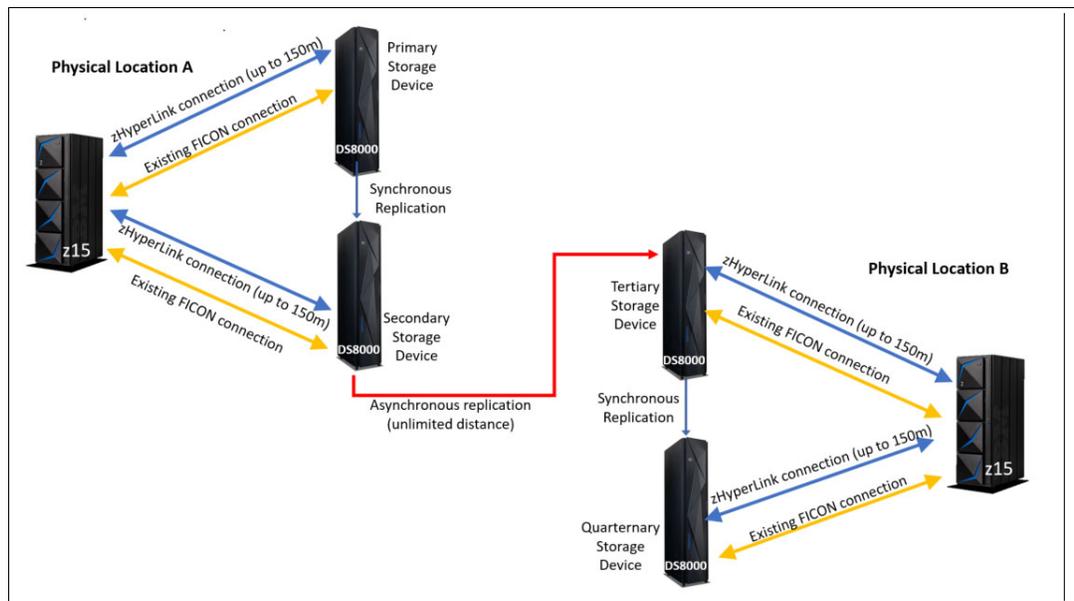


Figure 2-3 Multi-site configuration with extended distances

In the primary site, the primary and secondary DASD subsystems are replicated synchronously to provide high availability if a hardware failure occurs. The CPC system must have zHyperLink connectivity to the primary and the secondary DASD subsystems to ensure zHyperLink write availability and if a planned or unplanned switch to secondary DASD subsystem occurs.

The replication between the primary and secondary (DR) site is performed asynchronously to allow almost unlimited distances between sites. This configuration reduces the risks of regional disasters that might affect both sites.

At the DR site, a primary and auxiliary storage subsystem also can be available to provide high availability. The replication between the primary and secondary DASD subsystems at the DR site can be performed synchronously. zHyperLink connections from the CPC are configured to both DASD subsystems.

zHyperLink connectivity to the primary and the secondary DASD subsystems at the DR site are required to ensure that your application response times are not affected if a disaster occurs at the primary site and the workload is performed in the DR site.

2.1.6 IBM z Batch Network Analyzer

The use of tools to identify workloads that can benefit from zHyperLink implementation can reduce the amount of time that is necessary to identify such workloads, and size the possible benefits.

By using IBM z Batch Network Analyzer (zBNA), you can see more information about I/O response times, including expected benefits from zHyperLink implementation. It is designed to help identify technology options (zEDC, DFSMS Encryption, CF Encryption, and zHyperLink) to reduce the batch window.

The zBNA is a PC-based productivity tool that provides a means of estimating the elapsed time for batch jobs that is solely based on the differences in CPU speeds for a base processor and a target processor, the number of engines on each system, and system capacities (data sharing is not considered). zBNA provides powerful, graphics-based demonstration of the z/OS batch window.

The zBNA tool requires the IBM CPS Java Runtime Environment v8 64-bit and runs on the Windows 7, 8, or 10 64-bit platform.

Input for the zBNA is SMF data that is extracted on the z/OS system by using the z/OS Data Extraction Program (CP3KEXTR). The CP3KEXTR program reads SMF records and extracts data that is needed as input to IBM's Processor Capacity Reference (zPCR) and zBNA tools.

The zBNA and CP3KEXTR programs are available for download at no charge from the IBM Techdocs website. The latest versions that must be used are [zBNA version 2.2.4](#) and [CP3KEXTR version 4.21](#).

A zBNA Users Guide and Education materials, including FAQs, a demonstration video, and lab exercises, also are available at the IBM Techdocs page for zBNA. A Users Guide for CP3KEXTR is available on the IBM Techdocs page for CP3KEXTR.

CP3KEXTR output files are a flat Enterprise data file (.edf format file) plus a flat text DAT file. Because the output files from the CP3KEXTR extract tool can be large (even for a single day for one LPAR), the capability to terse the resulting output was added.

We suggest to always use the terse option. zBNA automatically unterses the files. It also provides you with the option of saving the untersed versions, which avoids the processing time to unterse it again.

The process of SMF data being collected, tersed, and transferred to the PC is shown in Figure 2-4.

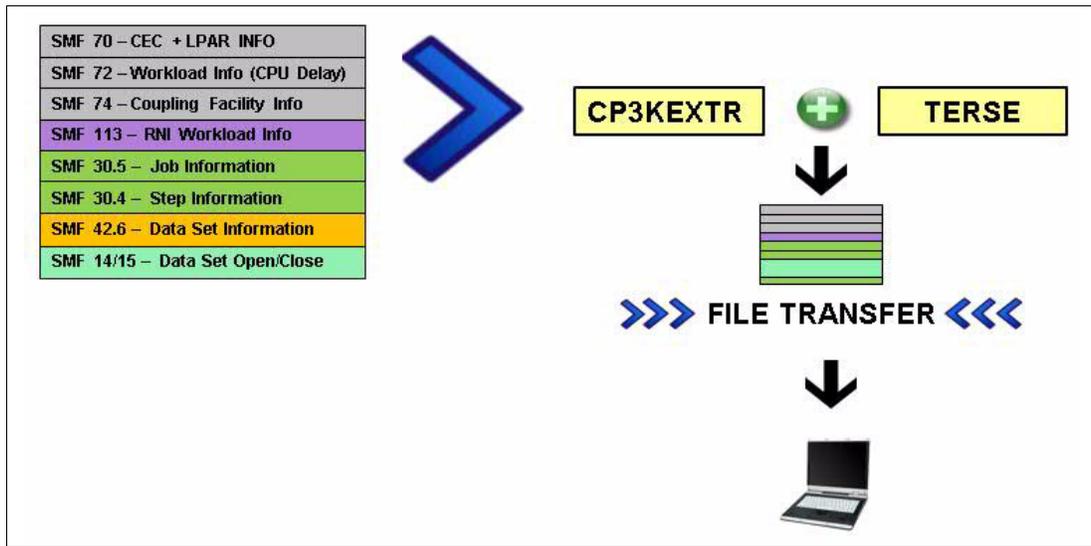


Figure 2-4 SMF data being collected, tersed, and transferred to the PC

zBNA supports estimating benefits of zHyperLink by providing reports with text and graphics, including a list of top data sets candidates. Data can be filtered by date and time. It also includes support for aggregating zBNA LPAR results into CPC level views.

An example of zHyperLink report created by zBNA after running the process is shown in Figure 2-5.

Data Set Name	Storage Class	Type	Extended Format	Block Size	Total IO Number	Read %	zHL Active	% zHL Eligible	% zHL Hit	Avg. IO Response Time	Estimated Response Time with zHL	% Improvement	Average Re-Dispatch + Response	Estimated Re-Dispatch + Response with zHL	% Improvement w/ Re-Dispatch
DB2DBSG.DSNDBD.BOOKXINVENT.J0001.A001	DB2DATA	Linear	Yes	4,096	528,472	100.0%	No	100.0%	99.6%	0.228 ms	0.055 ms	76.0%	0.428 ms	0.055 ms	87.0%
DB2DBSG.DSNDBD.BOOK.TSCUSACT.J0001.A001	DB2DATA	Linear	Yes	4,096	252,450	61.5%	No	61.5%	100.0%	0.285 ms	0.175 ms	38.6%	0.485 ms	0.252 ms	48.1%
DB2DBSG.DSNDBD.BOOK.TSINVENT.J0001.A001	DB2DATA	Linear	Yes	4,096	64,426	100.0%	No	100.0%	42.9%	4.096 ms	4.020 ms	1.8%	4.296 ms	4.134 ms	3.8%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A001	DB2DATA	Linear	Yes	4,096	540,723	84.8%	No	11.0%	100.0%	5.307 ms	5.239 ms	1.3%	5.507 ms	5.417 ms	1.6%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A002	DB2DATA	Linear	Yes	4,096	536,487	84.3%	No	10.4%	100.0%	9.952 ms	9.886 ms	0.7%	10.152 ms	10.065 ms	0.9%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A003	DB2DATA	Linear	Yes	4,096	538,606	83.8%	No	10.1%	100.0%	11.690 ms	11.620 ms	0.6%	11.890 ms	11.800 ms	0.8%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A005	DB2DATA	Linear	Yes	4,096	565,922	83.9%	No	9.6%	100.0%	7.895 ms	7.805 ms	1.1%	8.095 ms	7.986 ms	1.4%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A004	DB2DATA	Linear	Yes	4,096	537,177	84.0%	No	9.8%	100.0%	0.379 ms	0.354 ms	6.4%	0.579 ms	0.535 ms	7.6%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A009	DB2DATA	Linear	No	4,096	48,180	100.0%	No	95.4%	100.0%	0.209 ms	0.033 ms	84.0%	0.409 ms	0.043 ms	89.6%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A010	DB2DATA	Linear	No	4,096	48,006	100.0%	No	94.8%	100.0%	0.209 ms	0.035 ms	83.5%	0.409 ms	0.045 ms	89.1%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A006	DB2DATA	Linear	Yes	4,096	538,019	84.0%	No	7.9%	99.8%	4.493 ms	4.450 ms	0.9%	4.693 ms	4.634 ms	1.2%
DB2DBWG.DSNDBD.DBSCTL80.TSPART80.I0001.A009	DB2DATA	Linear	No	4,096	44,496	100.0%	No	95.0%	100.0%	0.223 ms	0.035 ms	84.3%	0.423 ms	0.045 ms	89.3%
DB2DBWG.DSNDBD.DBSCTL80.TSPART80.I0001.A010	DB2DATA	Linear	No	4,096	41,599	100.0%	No	93.7%	100.0%	0.211 ms	0.037 ms	82.6%	0.411 ms	0.049 ms	88.0%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A008	DB2DATA	Linear	No	4,096	40,748	99.9%	No	94.6%	100.0%	0.217 ms	0.036 ms	83.6%	0.417 ms	0.046 ms	88.9%
DB2DBWG.DSNDBD.DBSCTL80.TSVPEND80.I0001.A001	DB2DATA	Linear	No	4,096	3,185,263	100.0%	No	1.2%	100.0%	0.593 ms	0.590 ms	0.5%	0.793 ms	0.788 ms	0.7%
DB2DBWG.DSNDBD.DBSCTL80.TSPART80.I0001.A008	DB2DATA	Linear	No	4,096	40,274	100.0%	No	94.6%	100.0%	0.216 ms	0.035 ms	83.6%	0.416 ms	0.046 ms	88.9%
DSNDBSG.DSNDBD.DSNDB01.DSNACTP01.I0001.A001	DB2DATA	Linear	Yes	4,096	37,863	100.0%	No	100.0%	100.0%	0.200 ms	0.025 ms	87.5%	0.400 ms	0.025 ms	93.8%
DB2DBWG.DSNDBD.DBSCTL80.TSPART80.I0001.A011	DB2DATA	Linear	No	4,096	40,317	100.0%	No	93.1%	100.0%	0.210 ms	0.038 ms	82.0%	0.410 ms	0.052 ms	87.4%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A011	DB2DATA	Linear	No	4,096	38,665	100.0%	No	92.9%	100.0%	0.212 ms	0.038 ms	81.9%	0.412 ms	0.053 ms	87.2%
DB2DBWG.DSNDBD.DBSCTL80.TSPART80.I0001.A007	DB2DATA	Linear	No	4,096	37,927	100.0%	No	94.3%	100.0%	0.219 ms	0.036 ms	83.4%	0.419 ms	0.048 ms	88.6%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A007	DB2DATA	Linear	No	4,096	37,503	99.9%	No	94.1%	100.0%	0.219 ms	0.037 ms	83.2%	0.419 ms	0.049 ms	88.4%
DB2DBWG.DSNDBD.DBSCTL80.TSPART80.I0001.A012	DB2DATA	Linear	No	4,096	36,089	100.0%	No	91.9%	100.0%	0.212 ms	0.040 ms	81.0%	0.412 ms	0.056 ms	86.3%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A012	DB2DATA	Linear	No	4,096	33,313	100.0%	No	91.7%	100.0%	0.213 ms	0.041 ms	80.9%	0.413 ms	0.057 ms	86.1%
DSNDBWG.DSNDBD.DSNDB06.SYSTSPKG.I0001.A001	DB2DATA	Linear	Yes	4,096	29,521	100.0%	No	99.4%	100.0%	0.200 ms	0.027 ms	86.4%	0.400 ms	0.028 ms	92.9%
DB2DBWG.DSNDBD.DBITRK95.TSACTI95.I0001.A015	DB2DATA	Linear	No	4,096	27,877	100.0%	No	100.0%	100.0%	0.193 ms	0.025 ms	87.0%	0.393 ms	0.025 ms	93.6%
DB2DBWG.DSNDBD.DBITRK95.TSACTI95.I0001.A005	DB2DATA	Linear	No	4,096	27,495	100.0%	No	100.0%	100.0%	0.192 ms	0.025 ms	86.9%	0.392 ms	0.025 ms	93.6%
DB2DBWG.DSNDBD.DBITRK95.TSACTI95.I0001.A012	DB2DATA	Linear	No	4,096	27,403	100.0%	No	100.0%	100.0%	0.191 ms	0.025 ms	86.9%	0.391 ms	0.025 ms	93.6%
DB2DBWG.DSNDBD.DBITRK95.TSACTI95.I0001.A011	DB2DATA	Linear	No	4,096	27,392	100.0%	No	100.0%	100.0%	0.191 ms	0.025 ms	86.9%	0.391 ms	0.025 ms	93.6%

Figure 2-5 zBNA report for zHyperLink statistics

Note: Redispach time is a relevant part of the total I/O response time. The columns that reference this metric are important to determine the benefits of implementing zHyperLink.

From the report that is generated, a graphical view can be created with zHyperLink expected improvements. An example of the zBNA graphical view of the Average response time that estimates the benefit of zHyperLink implementation for all data sets is shown in Figure 2-6.

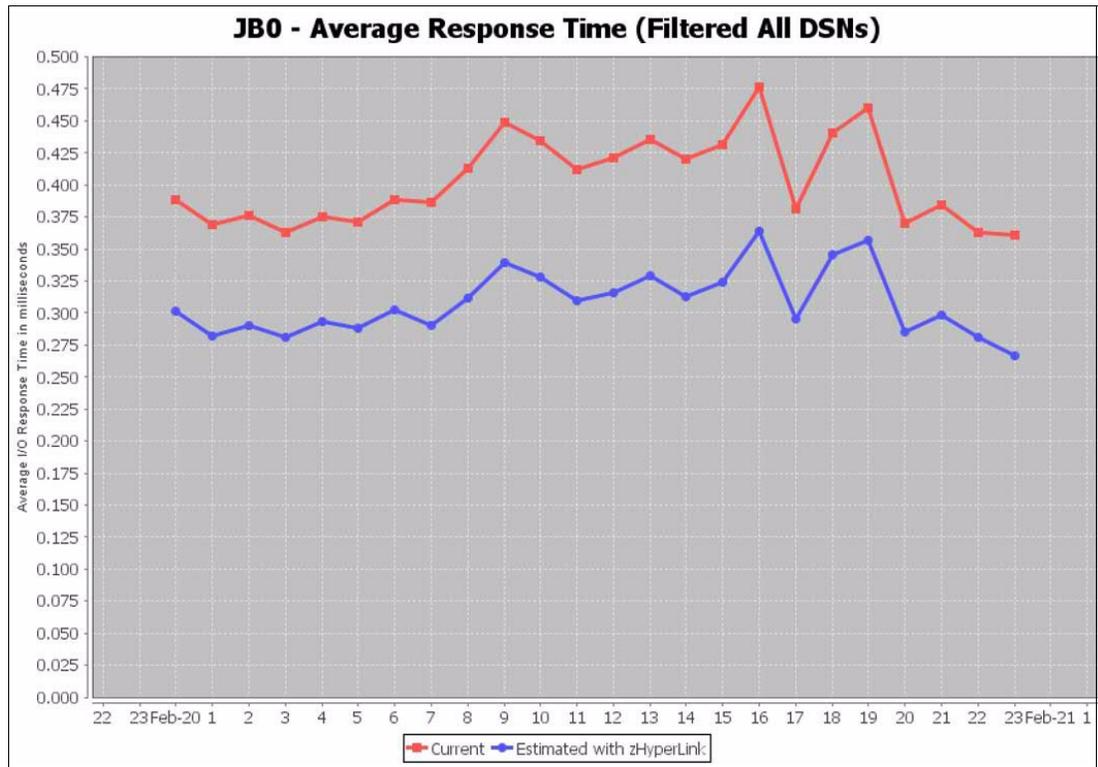


Figure 2-6 zBNA average response time

zBNA provides a chart with I/O that can be eligible for zHyperLink but was not yet implemented. The chart also provides information about overall potential zHyperLink I/O rate for the data sets included in your loaded file, as shown in Figure 2-7.

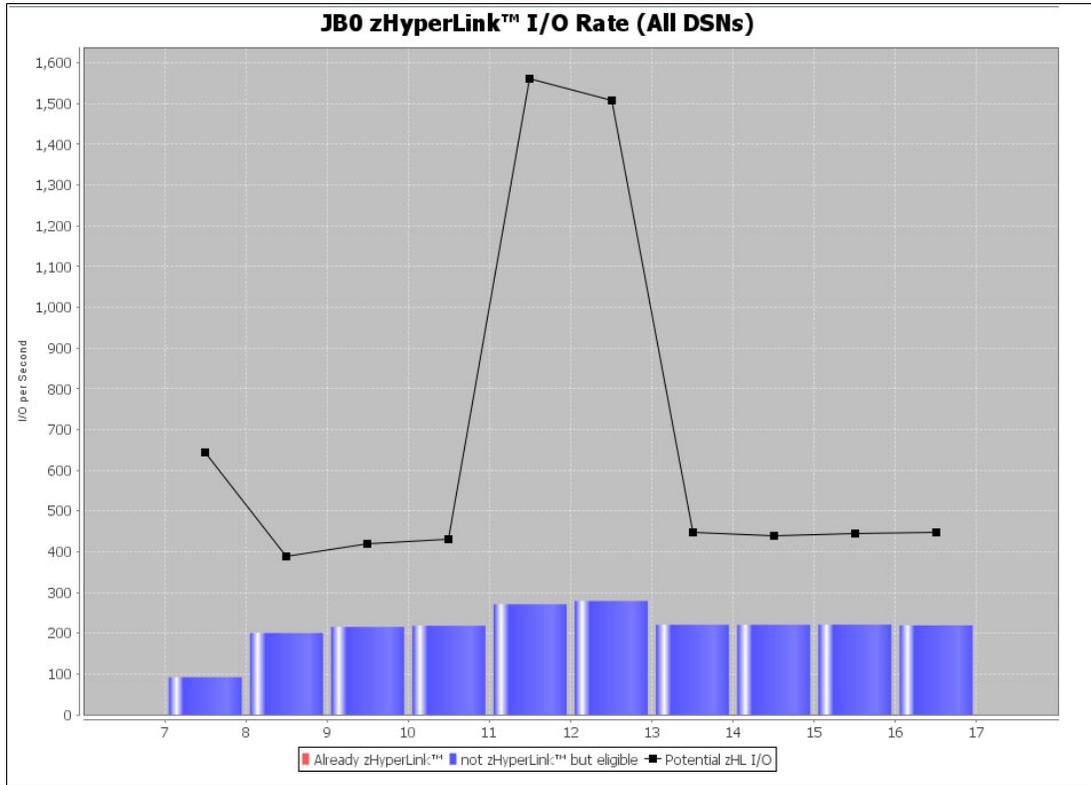


Figure 2-7 zBNA I/O Rate for zHyperLink

Note: The x-axis shows the hour of the day when the I/O occurred. The timestamp that is shown on the x-axis is governed by the time filter that is used when the chart was generated (07:00 - 17:00).

As shown in Figure 2-8, a summary report also can be made available that features relevant statistics about zHyperLink. It can assist with studying zHyperLink and determine how much the performance can be improved with its implementation.

Number of zHyperLink™ Read Eligible Data Sets	13,196
Total IO Number	24,200,950
Total Read Percent	87.6%
Total % zHL Eligible	32.8%
Total % zHL Hit	97.7%
Total Avg. IO Response Time	2.816
Total Estimated Response Time with zHL	2.732
Total % Improvement	3.0%
Total Average Re-Dispatch + Response Time	3.016
Total Estimated Re-Dispatch + Response Time with zHL	2.868
Total % Improvement with Re-Dispatch	4.9%

Figure 2-8 zBNA Summary Report

In addition to the zHyperLink I/O response time, the time that is spent with redispatch process is an important variable for zHyperLink. In comparison with a normal I/O, zHyperLink provides a considerable improvement with elimination of redispatch process,

zBNA provides the current time of average redispatch and response time, which is the sum of the average I/O response and specified redispatch time.

Figure 2-9 displays the estimates of the New zHyperLink Re-Dispatch and Response times after implementing zHyperLink. It considers zHyperLink read eligible DASD cache hits and the elimination of redispatch time for those hits. The last column of the sample in Figure 2-9 shows the percentage Improvement with redispatch.

Data Set Name	Data Set Type	Extended Format	Total IO Number	Read %	zHL Active	% zHL Eligible	% zHL Hit	Avg. IO Response Time	Estimated Response Time with zHL	% Improve	Average Re-Dispatch + Response Time	Estimated Re-Dispatch + Response Time with zHL	% Improve with Re-Dispatch
DB2DBSG.DSNDBD.BOOK.XINVENT.J0001.A001	Linear	Yes	528,472	100.0%	No	100.0%	99.6%	0.228	0.055	76.0%	0.428	0.055	87.0%
DB2DBSG.DSNDBD.BOOK.TSCUSACT.J0001.A001	Linear	Yes	252,450	61.5%	No	61.5%	100.0%	0.285	0.175	38.6%	0.485	0.252	48.1%
DB2DBSG.DSNDBD.BOOK.TSINVENT.J0001.A001	Linear	Yes	64,426	100.0%	No	100.0%	42.9%	4.096	4.020	1.8%	4.296	4.134	3.8%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A001	Linear	Yes	540,723	84.8%	No	11.0%	100.0%	5.307	5.239	1.3%	5.507	5.417	1.6%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A002	Linear	Yes	536,487	84.3%	No	10.4%	100.0%	9.952	9.886	0.7%	10.152	10.065	0.9%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A003	Linear	Yes	538,606	83.8%	No	10.1%	100.0%	11.690	11.620	0.6%	11.890	11.800	0.8%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A005	Linear	Yes	565,922	83.9%	No	9.6%	100.0%	7.895	7.805	1.1%	8.095	7.986	1.4%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A004	Linear	Yes	537,177	84.0%	No	9.8%	100.0%	0.379	0.354	6.4%	0.579	0.535	7.6%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A009	Linear	No	48,180	100.0%	No	95.4%	100.0%	0.209	0.033	84.0%	0.409	0.043	89.6%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A010	Linear	No	48,006	100.0%	No	94.8%	100.0%	0.209	0.035	83.5%	0.409	0.045	89.1%
DSNDBSG.DSNDBD.WRKDBS6.DSN4K04.I0001.A006	Linear	Yes	538,019	84.0%	No	7.9%	99.8%	4.493	4.450	0.9%	4.693	4.634	1.2%
DB2DBWG.DSNDBD.DBSCTL80.TSPART80.I0001.A009	Linear	No	44,496	100.0%	No	95.0%	100.0%	0.223	0.035	84.3%	0.423	0.045	89.3%
DB2DBWG.DSNDBD.DBSCTL80.TSPART80.I0001.A010	Linear	No	41,599	100.0%	No	93.7%	100.0%	0.211	0.037	82.6%	0.411	0.049	88.0%
DB2DBWG.DSNDBD.DBSCTL81.TSPART81.I0001.A008	Linear	No	40,748	99.9%	No	94.6%	100.0%	0.217	0.036	83.6%	0.417	0.046	88.9%
DB2DBWG.DSNDBD.DBSCTL80.TSVEND80.I0001.A001	Linear	No	3,185,263	100.0%	No	1.2%	100.0%	0.593	0.590	0.5%	0.793	0.788	0.7%
Totals			7,510,574	91.8%		17.5%	97.0%	3.194	3.145	1.5%	3.394	3.311	2.5%

Figure 2-9 Top 15 candidate data sets

More information about the top 15 candidate data sets for the use of zHyperLink also is shown in Figure 2-9.

Note: The table in Figure 2-9 is determined by the total number of zHyperLink read eligible I/Os. By default, the list with the top 15 candidate data sets is displayed in descending order.

For z/OS 2.3, before collecting data, z/OS fix for APAR OA53110 must be installed to perform the analysis.

The PTF is an enhancement to DFSMS Common Measurement Manager. It adds DFSMS statistics to the following SMF Records:

- ▶ Type 42 SubType 5:
 - Storage class response time
 - VTOC data component I/O statistics
 - VTOC index component I/O statistics
 - VVDS component I/O statistics
- ▶ Type 42 SubType 6 Data Set I/O statistics

For Db2, before collecting data, the following APARs must be installed to perform the analysis:

- ▶ Db2 v11 PTF for APAR PI99235
- ▶ Db2 v12 PTF for APAR PI89828

The PTFs add data that identifies I/Os that are candidates for zHyperLink read to SMF type 42 SubType 6 data set I/O Statistics.

2.2 Sizing synchronous I/O for zHyperLink

In this section, we describe the maximum number of supported ports, how to determine the number of required zHyperLinks, and considerations for future growth.

2.2.1 Number of connections

The IBM z14 (FC #0431) and IBM z15 (FC #0451) for zHyperLink Express are two-port features that are installed in the PCIe I/O drawer. Up to 16 features with up to 32 zHyperLink Express ports are supported on IBM z14, IBM z14 ZR1, IBM z15 T01, and IBM z15 T02 CPCs.

On the DS8000 DASD subsystem, the number of zHyperLink ports that can be installed varies, depending on the number of available cores per CPC and the number of I/O bay enclosures. The number of zHyperLinks that can be installed based on the number of cores available is listed in Table 2-2.

Table 2-1 zHyperLink availability by DS8000 model

DS8000 Model	Cores per CPC	zHyperLink support	Max. zHyperLink connections (increments of 2)
DS8882F	6	No	N/A
DS8884/DS8884F	6	No	N/A
	12	Yes	4
DS8886/DS8886F	8	No	N/A
	16	Yes	8
	24	Yes	12
DS8888/DS8888F	24	Yes	8
	48	Yes	16
DS8910F - 993	16	Yes	4
DS8910F - 994	16	Yes	4
DS8950F - 996	20	Yes	6
DS8950F - 996/E96	40	Yes	12
DS8980F - 998/E96	44	Yes	12

On the DS8000, DASD subsystems can have up to eight ports in a single frame, or 16 ports if an extension frame also exists. Each zHyperLink I/O adapter features one port, but you must order them in pairs (x2). Depending on the system configuration, zHyperLink I/O adapters can be ordered in quantities of 2, 4, 6, and 8. Each I/O Bay (Enclosure) can have two ports and are upgraded in pairs.

Important: A zHyperLink connection is point-to-point, which means switching is *not* supported.

zHyperLink allows the following interconnections:

- ▶ One IBM z14 to one DS8880/DS8900 by using multiple zHyperLink connection pairs
- ▶ One IBM z15 to one DS8880/DS8900 by using multiple zHyperLink connection pairs
- ▶ Multiple IBM z14 CPCs to a single DS8880/DS8900
- ▶ Multiple IBM z15 CPCs to a single DS8880/DS8900
- ▶ Multiple DS8880/DS8900s to a single IBM z14 CPC
- ▶ Multiple DS8880/DS8900s to a single IBM z15 CPC

Any interconnection between CPCs and DS8000 is supported if connections are in pairs and within the maximum number of ports.

2.2.2 Using Resource Measurement Facility data for sizing

The IBM Resource Measurement Facility (RMF) provides a wide range of information about your system usage, including I/O information that can be used for sizing your I/O system.

From RMF, you can use the Channel Path Activity report (CHANNEL) to display information about all channel paths in the system. From this panel, you can access the following channel information:

- ▶ Percentage utilization
- ▶ Read (Bytes per second)
- ▶ Write (Bytes per second)
- ▶ FICON operations rate
- ▶ zHPF operations rate

To request the Channel Path Activity report, select **3 - Monitor III** from the primary menu and then, select **12 - Channel** in the Resource Report Selection menu. You also can use the **CHANNEL** command.

Locate the FICON channels that are used for the DS8000 disk subsystem in question and analyze the usage.

Also, analyze information regarding cache reads. You can run an RMF Postprocessor report that specifies `REPORTS(CACHE(SUBSYS))` to extract cache read information. To request the postprocessor report, select **1 - Postprocessor** from the primary menu or run it by using a batch job of your own.

Note: CACHE must be enabled in the ERBRMFxx parmlib member.

For more information about zHyperLink and RMF, see 3.7, “Measuring after deployment” on page 73.

2.2.3 I/O connectivity planning

Standard recommendations for channels, such as FICON and OSA, also apply to zHyperLink. Ensure that connectivity is spread out on as many different I/O cards for the IBM z14/IBM z15 CPC and the DS8000 system. This configuration limits consequences that can result from component failures.

When planning zHyperLink implementation, plan for technology refresh and future growth for the IBM z14/IBM z15 CPC and the DS8000 system.

The IBM zHyperLink Express feature is a two-port feature. All other connectivity for IBM Z future system updates or changes to the configuration must be considered when planning port allocation.

Connectivity must be spread out on the available cards. If you are connecting an IBM Z CPC with four features to two DS8000s with four ports each, we suggest the use of one port at each IBM Z CPC for the first DS8000 and the other port of each IBM Z CPC for the second DS8000.

Consider ordering enough ports for future growth. If all installed ports are in use and you need to add two ports for connecting to a new DS8000, this configuration causes a single point of failure (SPOF) situation if both of the new ports are on a single feature in the IBM Z CPC. This issue can be avoided by adding ports zHyperlink features) or by relocating connectivity for some on the current ports.

Note: Unlike FICON, swapping a zHyperLink within a CPC does not require I/O configuration updates. You can swap links by configuring them offline, swapping the physical cables, and then, configuring them back online. It is important to check the workload and verify the links after swapping them.

For more information about commands that are used for configuring online, offline, and verifying zHyperLink features and ports, see 4.1, “System commands” on page 80.

Also, ensure that your systems have enough zHyperLink connections to each box to manage possible maintenance activities, such as replacing a zHyperLink adapter on the CPC or the DASD subsystem.

2.3 zHyperLink hardware planning

In this section, we describe the microcode and how to plan for zHyperLink hardware updates, including microcode and ordering required features.

2.3.1 Microcode for zHyperLink

Planning for implementing zHyperLink includes verifying the necessary levels of microcode at both ends of the point-to-point connection; that is, the IBM Z CPC and the DS8000 DASD subsystem. The microcode must be updated if it is not at the required level.

The following microcode levels are required:

- ▶ IBM z14 mainframe at level GA2 D36C bundle S08 or higher
- ▶ DS8880 is at least 8.3.2 (88.32.6.0) for read operations
- ▶ DS8880 is at least 8.5.1 (88.55.9.1) for writes with no or synchronous mirroring

- ▶ DS8900 is at least 9.1 (89.10.92.0) for writes with synchronous or asynchronous mirroring

Note: IBM z15 has no specific level required for zHyperLink. Any level is supported.

In addition to the DS8000 microcode level, the DS8000 z-synergy Services license is needed.

2.3.2 IBM z14 and IBM z15 microcode change level

Regular installation of Microcode Change Levels (MCLs) is key for reliability, availability, and serviceability (RAS), and optimal performance. We recommend that the following tasks are performed:

- ▶ Install and activate MCLs on a quarterly basis, at a minimum.
- ▶ Review hiper MCLs continuously to decide whether to wait for the next scheduled fix application session or to schedule a fix earlier based on risk assessment.

Tip: For the IBM z14/IBM z15, IBM Resource Link® provides access to the system information for your IBM Z system according to the system availability data that is sent on a scheduled basis and information about the MCL status of your IBM z14/IBM z15.

For more information, see the [IBM Resource Link website](#).

IBM z14 and IBM z15 microcode terms

The microcode features the following characteristics:

- ▶ The driver contains engineering change (EC) streams.
- ▶ Each EC stream covers the code for a specific component of IBM Z CPCs. It includes a specific name and an ascending number.
- ▶ The EC stream name and a specific number are one MCL.
- ▶ MCLs from the same EC stream must be installed in sequence.
- ▶ MCLs can include installation dependencies on other MCLs.
- ▶ Combined MCLs from one or more EC streams form one bundle.
- ▶ An MCL contains one or more Microcode Fixes (MCFs).

A bundle is a set of MCLs that is grouped during testing and released as a group on the same date. You can install an MCL to a specific target bundle level.

On the IBM Z interface, the System Information window is enhanced to display a summary bundle level for the activated level, as shown in Figure 2-10.

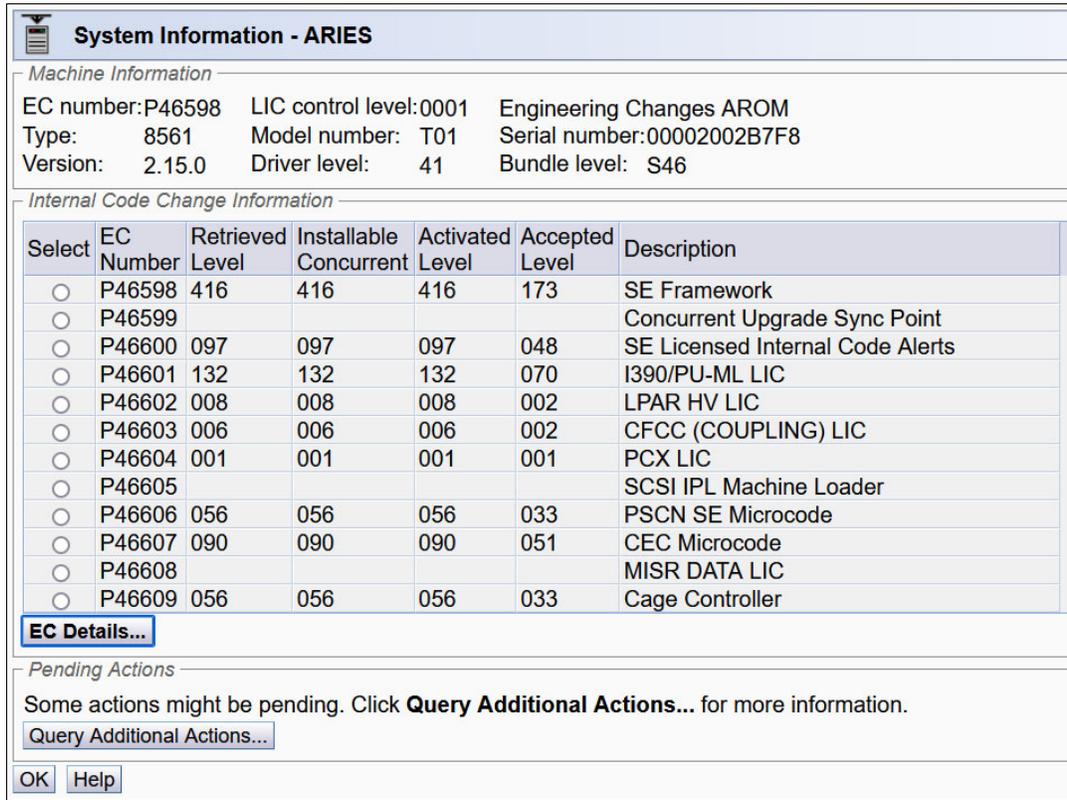


Figure 2-10 System Information: Bundle level

2.3.3 IBM DS8000 microcode change level

The DS8000 z-synergy Services license is required for zHyperLink. The initial enablement of any optional IBM DS8000 licensed function is a concurrent activity (assuming that the suitable level of microcode is installed on the machine for the specific function).

When IBM releases new microcode for the DS8000, it is released in the form of a bundle. The term *bundle* is used because a new code release can include updates for various DS8000 components. These updates are tested together, and then, the various code packages are bundled into one unified release.

In general, use the term *bundle* when referring to the code level that is used on a DS8000. Components within the bundle each feature their own revision levels.

For a DS8000 cross-reference table of code bundles, see [this website](#).

At the website, search for DS8800 code bundle or DS8900F code bundle, and then, click **DS8880/DS8900F code bundle information**. The levels of code for released bundles are listed on the web page. The information is updated as new code bundles are released. It is important to maintain a current version of the data storage command-line interface (DS CLI).

For the DS8000, the naming convention of bundles is PR.MM.FFF.E, where:

- ▶ P: Product (8 = DS8000)
- ▶ R: Release Major (X)
- ▶ MM: Release Minor (xx)
- ▶ FFF: Fix Level (xxx)
- ▶ E: EFIX level (0 is base, and 1.n is the interim fix build above base level.)

For more information about DS8880 and DS8900 Microcode updates, see *IBM System Storage DS8000: Architecture and Implementation*, SG24-8456.

2.3.4 IBM z14 and IBM z15 feature codes

The IBM zHyperLink Express is a two-port feature that is installed in the PCIe I/O drawer. You can install a maximum of 16 features with two zHyperLink Express ports each on the IBM z14 or IBM z15. The feature code to order is FC #0431 for IBM z14 and FC #0451 for IBM z15. Installing PCIe zHyperLink features on an IBM Z CPC is a non-disruptive operation.

zHyperLink connections are required to be configured in pairs. Therefore, the minimum number of features that is required to avoid single point of failure (SPOF) is two.

zHyperLink Express feature works as a native PCIe adapter that can be shared by multiple LPARs. Each port can support up to 127 Virtual Functions (VFs), with one or more VFs/PFID²s being assigned to each LPAR. This configuration supports a maximum of 254 VFs per adapter. The maximum of 127 Virtual Functions per port means that a link can be shared by all LPARs in the CPC (85 LPARs per CPC for IBM z14/IBM z15-T01 and 40 LPARs per CPC for IBM z14 ZR1/IBM z15-T02).

2.3.5 DS8000 feature codes

The zHyperLink feature on DS8000 systems sits on the I/O Bay Enclosure. The feature includes one port each, although the minimum increment is two ports on the I/O Bay enclose, which also is always installed in pairs and results in increments of four ports.

The DS8880 and DS8900 feature code for zHyperLink adapter is #3500. Depending on your configuration, other feature codes might be required; for example, FC #4235 for 256 GB system memory and FC #4425 for 12 core POWER processors (DS8880).

Installing zHyperLink features on a DS8000 system is non-disruptive.

The maximum number of zHyperLink features (FC #3500) on a DS8000 is shown in Table 2-1 on page 27.

For more information about DS8000 zHyperLink features, see 1.3.1, “Hardware prerequisites” on page 5.

2.3.6 zHyperLink cables

A 24x MTP-MTP cable is required for each port of the zHyperLink Express feature.

Cables can be ordered as features codes for the DS8000, with lengths of 40 meters (131.23 feet) or 150 meters (492.12 feet). Other length options also are available. The cables are identical to the cables that are used for the ICA SR feature (coupling for IBM Z - FC #0172 for IBM z14 or FC #0176 for IBM z15).

² PCIe Function ID

The HyperLink Express and ICA SR features are designed to drive distances up to 150 meters (492.12 feet) and support a link data rate of 8 GBps by using customer-supplied OM4 (4.7 GHz-km @ 850 nm) MTP-MTP fiber optic cables. With OM3 (2.0 GHz-km @ 850 nm) fiber optic cables, zHyperLink Express and ICA SR distance is limited to 100 meters (328.08 feet).

An OM4 24-fiber cable with Multi-fiber Terminated Push-on (MTP) connectors for zHyperLink is shown in Figure 2-11.

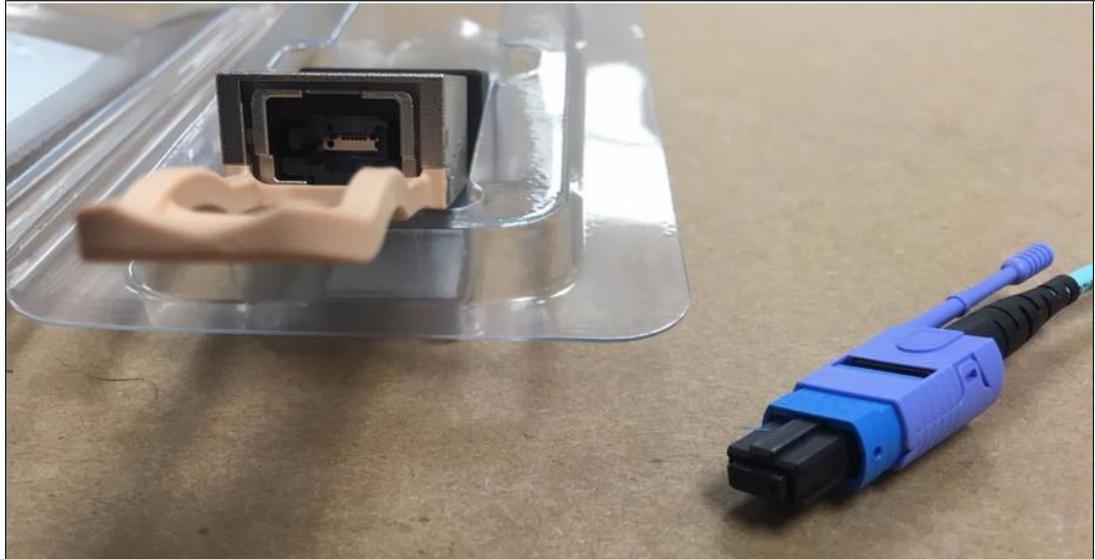


Figure 2-11 Sample zHyperLink cable and connector

The custom or standard cable lengths that are listed in Table 2-2 are available from IBM or other vendors. Similar lengths are available as OM3 options, although those lengths are limited to 100 meters (328.08 feet).

Table 2-2 Single MTP 24 to single MTP 24 fiber optic ASM (OM4)

P/N	Length (m)
00JA683	1.0
00JA684	2.0
00JA685	3.0
00JA686	5.0
00JA687	8.0
00LU282	10.0
00LU283	13.0
00JA688	15.0
00JA689	20.0
00LU284	40.0
00LU285	80.0
00LU286	120.0

P/N	Length (m)
00LU287	150.0
00LU288	Custom

For more information, see *IBM Z Planning for Fiber Optic Links (FICON/FCP, Coupling Links, Open Systems Adapters, and zHyperLink Express)*, GA23-1408.

2.3.7 I/O configuration

PCIe function for zHyperLink also must be defined to enable zHyperLink. Plan to update the I/O configuration with zHyperLink. An example of Hardware Configuration Definition (HCD) panel is shown in Figure 2-12.

```

+----- Add PCIe Function -----+
|
| Specify or revise the following values.
|
| Processor ID . . . . . : CPC1SIT1      IBM z15 CPC Site 1
|
| Function ID . . . . . 1001
| Type . . . . . ZHYPERLINK      +
|
| Channel ID . . . . . 140      +
| Port . . . . . 1              +
| Virtual Function ID . . . . . 1    +
| Number of virtual functions . . 4
| UID . . . . . _____
|
| Description . . . . .
|
+-----+

```

Figure 2-12 Add PCIe function panel

The number of simultaneous I/O operations that can be active on a zHyperLink is determined by the number of virtual functions that is defined in the I/O configuration for that link. The virtual function number is a 1- to 3-digit number in the range 1-n (zHyperLink range is 1-127). zHyperLink adapters feature multiple ports and the virtual function values are on a port basis rather than an adapter basis.

We recommend defining four virtual functions per LPAR for each link for optimal performance. The limit of four is caused by storage controllers capability to handle up to four simultaneous requests per link.

An example of defined PCIe functions, including zHyperLink, is shown in Figure 2-13. Two ports are defined for each zHyperLink CHID.

PCIe Function List Row 47 of 66 More:						
Command ==>			_____		Scroll ==> CSR	
Select one or more PCIe functions, then press Enter. To add, use F11.						
Processor ID : CPC1SIT1 IBM z15 CPC Site 1						
/ FID	CHID+	P+	VF+	Type+	UID	Description
_ 1001	140	1	1	ZHYPERLINK	_____	LPAR LPZOSAPR _____
_ 1002	140	1	2	ZHYPERLINK	_____	LPAR LPZOSAPR _____
_ 1003	140	1	3	ZHYPERLINK	_____	LPAR LPZOSBPR _____
_ 1004	140	1	4	ZHYPERLINK	_____	LPAR LPZOSBPR _____
_ 1041	140	2	1	ZHYPERLINK	_____	LPAR LPZOSAPR _____
_ 1042	140	2	2	ZHYPERLINK	_____	LPAR LPZOSAPR _____
_ 1043	140	2	3	ZHYPERLINK	_____	LPAR LPZOSBPR _____
_ 1044	140	2	4	ZHYPERLINK	_____	LPAR LPZOSBPR _____

Figure 2-13 PCIe Function List panel

Virtual functions also must be defined for connectivity to the secondary DS8000 if you use metro-mirror and are planning to use zHyperLink writes.

The IOCP definition for zHyperLink as in HCD defines both ports on a single zHyperLink card (same PCHID but different port numbers), as shown in Example 2-1.

Example 2-1 Defining both ports on a single zHyperLink card

```

FUNCTION FID=1001,VF=1,PCHID=140,                                     *
    PART=((LPZOSAPR),(=),TYPE=HYL,PORT=1
FUNCTION FID=1001,VF=2,PCHID=140,                                     *
    PART=((LPZOSAPR),(=),TYPE=HYL,PORT=1
FUNCTION FID=1001,VF=3,PCHID=140,                                     *
    PART=((LPZOSBPR),(=),TYPE=HYL,PORT=1
FUNCTION FID=1001,VF=4,PCHID=140,                                     *
    PART=((LPZOSBPR),(=),TYPE=HYL,PORT=1
FUNCTION FID=1041,VF=1,PCHID=140,                                     *
    PART=((LPZOSAPR),(=),TYPE=HYL,PORT=2
FUNCTION FID=1041,VF=2,PCHID=140,                                     *
    PART=((LPZOSAPR),(=),TYPE=HYL,PORT=2
FUNCTION FID=1041,VF=3,PCHID=140,                                     *
    PART=((LPZOSBPR),(=),TYPE=HYL,PORT=2
FUNCTION FID=1041,VF=4,PCHID=140,                                     *
    PART=((LPZOSBPR),(=),TYPE=HYL,PORT=2

```

Important: zHyperLink connections cannot be used while the I/O Priority Manager is enabled on storage system. However, the I/O Priority Manager is disabled on DS8000 Storage Management only, but it can still operate normally on z/OS through WLM.

2.3.8 Access limitations

Important: When determining the number of features to be installed and defining links and virtual functions, consider the zHyperLink read and write access limitations that are set by the DS8000 storage controller or z/OS. The limitations are caused by the amount of non-volatile storage (NVS) in the DS8000:

- ▶ A maximum of four concurrent I/O requests are supported by the DS8000 on a single link
- ▶ A read request needs a single virtual function ID and is limited to 4K by z/OS.
- ▶ A write request needs one virtual function ID if the write is for a single track, or two requests if the write crosses a track boundary.
- ▶ A striped data set needs one virtual function ID per stripe and dual logging needs one virtual function for each of the logs. z/OS issues these write requests in parallel.

The DS8000 subsystem supports up to 64 concurrent write sessions. The number of active sessions can be displayed by using the **DS QD, MACH=nnnn-xxxxx, ZFL** command. For more information about the **DS QD, MACH=nnnn-xxxxx, ZFL** command, see 4.1, “System commands” on page 80.

The following example of calculating the number of virtual functions for zHyperLink is provided for a write to a dual logging data set with three stripes crossing a track boundary:

2 (dual logging) * 3 (striping) * 2 (crossing track boundary) = 12 virtual function IDs

The same links are used for read and write operations and no prioritization or pooling of those links exists. Also, LPARs compete for the links when those links are defined as shared between multiple LPARs. Requests are handled on a first-come, first-served basis.

2.4 Software planning

IBM zHyperLink is supported on z/OS operating system images that are running in an LPAR only. It is not supported on guest LPARS under z/VM or other operating systems. It was developed for IBM relational database Db2 for z/OS.

2.4.1 Levels and requirements

The following software prerequisites must be met to enable zHyperLink:

- ▶ z/OS 2.3³ or later and PTFs for the following FMIDs:
 - z/OS 2.3 BCP FMID HBB77B0
 - z/OS 2.3 HCD FMID HCS77B0
 - z/OS 2.3 DFSMS FMID HDZ2230
- ▶ Db2 v11 or v12 and PTFs for the following FMIDs:
 - DB2 v11 FMID HDBBB10
 - DB2 V12 FMID HDBCC10

³ Support was made available in z/OS V2R2; however, z/OS V2R2 is End of Support as of Sept. 30, 2020.

- ▶ IBM z High Performance FICON (zHPF)
- ▶ IBM zHyperWrite: Enabled for zHyperLink write support

2.4.2 Software parameters for z/OS

Verify z/OS I/O-related system parameters that are required for zHyperLink and plan for adding parameters that enable zHyperlink.

The IBM zHyperWrite data replication option is required for zHyperLink write. The High Performance FICON for IBM z Systems® (zHPF) option is required for both zHyperLink read and write.

D IOS, HYPERWRITE

Use the **D IOS, HYPERWRITE** command to verify status of the zHyperWrite option. The option is a prerequisite for zHyperLink write. The option does not need to be enabled for zHyperLink read. An example of the **D IOS, HYPERWRITE** command is shown in Example 2-2.

Example 2-2 Verifying zHyperwrite status

```
D IOS, HYPERWRITE
IOS634I 08.24.48 IOS SYSTEM OPTION 465
HYPERWRITE IS ENABLED
```

D IOS, ZHPF

Use the **D IOS, ZHPF** command to verify status of the zHPF option. The option is a prerequisite for zHyperLink. An example of the **D IOS, ZHPF** command is shown in Example 2-3.

Example 2-3 Verifying ZHPF status

```
D IOS, ZHPF
IOS630I 08.25.20 ZHPF FACILITY 467
HIGH PERFORMANCE FICON FACILITY IS ENABLED
```

D IOS, ZHYPERLINK

By default, the zHyperLink facility is disabled in z/OS. Use the **D IOS, ZHYPERLINK** command to verify the status of zHyperLink. If zHyperLink is not yet implemented, the **D IOS, ZHYPERLINK** command displays the status of **DISABLED**, as shown in Example 2-4.

Example 2-4 Verifying zHyperLink status

```
D IOS, ZHYPERLINK
IOS634I 14.34.43 IOS SYSTEM OPTION 268
ZHYPERLINK IS DISABLED
```

If zHyperLink is enabled on the system, one of the following statuses can be displayed:

- ▶ zHyperLink is enabled for read operations
- ▶ zHyperLink is enabled for write operations
- ▶ zHyperlink is enabled for read and write operations

When verifying zHyperLink status on an IBM Z CPC that does not support zHyperLink, the status is displayed as: zHyperLink is not supported by the processor.

For more information about how to enable options that are required for zHyperLink but are not yet enabled, see 3.3.2, “Activating zHyperLink” on page 65.

For more information about the **D IOS** MVS command, see Chapter 4 of *z/OS MVS System Commands*, SA38-0666⁴.

IECIOSxx parmlib member

An alternative to displaying the status of zHyperWrite, zHPF, and zHyperLink information is to verify the status by browsing the IECIOSxx parmlib member.

If the HYPERWRITE parameter is not specified in IECIOSxx, the option that is enabled as the z/OS default for zHyperWrite is YES. For the zHyperLink option, the z/OS default is NONE; therefore, if the parameter is not specific, zHyperLink is disabled for read and write operations.

An example of an IECIOSxx parmlib member where zHyperWrite and zHPF are enabled and zHyperLink is disabled is shown in Example 2-5.

Example 2-5 Sample IECIOSxx parmlib member before enabling zHyperLink

```
HYPERWRITE=YES  
ZHPF=YES  
ZHYPERLINK,OPER=NONE
```

Attention: Growing the IECIOSxx parmlib member might not reflect the status of the I/O-related features because those I/O-related features might be updated dynamically by using system commands.

For more information about the IECIOSxx PARMLIB member, see Chapter 57 of *z/OS MVS Initialization and Tuning Reference*, SA23-1380⁴.

2.4.3 zHyperLink Storage Class Granularity activation plan

Storage Class Granularity is a single policy that covers all users so that zHyperLink can be managed across all applications. The policy turns zHyperLink *off* and *on* for specific non-SMS data sets. It also overrides zHyperLink specification for SMS-managed data sets.

By using Storage Class Granularity, the system administrator can enable or disable synchronous I/O to the data sets that are members of a specific storage class (SC), and enable or disable synchronous I/O for specific data sets by using an operator command. The ability to dynamically enable or disable zHyperLink also facilitates testing this facility.

⁴ Match the edition of this manual to your running z/OS version.

With Storage Class Granularity support, two new SC attributes were added and can be found in DCOLLECT Record Type 'SC'. The Storage Class definition is shown in Figure 2-14.

Guaranteed Space	N	(Y or N)
Guaranteed Synchronous Write . . .	N	(Y or N)
Multi-Tiered SG		(Y, N, or blank)
Parallel Access Volume Capability	N	(R, P, S, or N)
CF Cache Set Name		(up to 8 chars or blank)
CF Direct Weight		(1 to 11 or blank)
CF Sequential Weight		(1 to 11 or blank)
CF Lock Set Name		(up to 8 chars or blank)
Disconnect Sphere at CLOSE	N	(Y or N)
<u>zHyperLink Eligible for read</u> . . .	Y	(Y or N)
<u>zHyperLink Eligible for write</u> . .	N	(Y or N)

Figure 2-14 ISMF Storage Class panel

Without requiring applications to restart, systems can start the use of synchronous I/O with zHyperLink for eligible data sets with Storage Class Granularity policy. Verify SMS-managed data sets that were unable to use zHyperLink before and can use zHyperLink after the SC definition is activated.

This facility also provides a **VARY SMS** command that is used to toggle zHyperLink on an individual data set basis:

```
V SMS,DSNAME(dsname)
```

The use of the **V SMS** command overrides the SC definition. Media Manager checks the eligibility flags, and after the command is issued, all current and subsequent (open) operations use the new specified value.

Note: The use of the **VARY SMS** command is the only way to enable zHyperLink processing for a non-SMS data set.

The following examples describe how to dynamically activate or deactivate synchronous I/O (SIO) by using the **VARY SMS** command:

- ▶ **VARY SMS,DSNAME(dsname),ZHLREAD=YES**
Enables synchronous I/O reads for data set. After the **vary** command overwrites the SC definition, the reads use synchronous I/O, even if SC is not eligible for reads.
- ▶ **VARY SMS,DSNAME(dsname),ZHLREAD=NO**
Overrides SC eligibility for reads and issues reads to the data set by using asynchronous I/O. Reads use zHPF because the data set is not enabled for Synchronous I/O reads.
- ▶ **VARY SMS,DSNAME(dsname),ZHLWRITE=YES**
Enables synchronous I/O writes for data set. After the **vary** command overwrites the SC definition, the writes use synchronous I/O, even if SC is not eligible.
- ▶ **VARY SMS,DSNAME(dsname),ZHLWRITE=NO**
Overrides SC eligibility for writes and issues writes to the data set by using asynchronous I/O. Writes use zHPF because the data set is not enabled for Synchronous I/O writes.
- ▶ **VARY SMS,DSNAME(dsname),ZHLWRITE=DEFAULT**
Sets zHyperLink writes to default value.

2.4.4 Software activation plan for Db2

For activation of zHyperLink I/O protocol on Db2, the ZHYPERLINK subsystem parameter is specified to determine the default scope of zHyperLink I/O requests. The following values are acceptable for zHyperLink on Db2:

- ▶ **DISABLE**
Db2 does not use zHyperLink for any I/O requests. This value is the default.
- ▶ **ENABLE**
All eligible I/O requests that Db2 makes request the zHyperLink protocol.
- ▶ **DATABASE**
Db2 requests the zHyperLink protocol only for database synchronous read I/Os.
- ▶ **ACTIVELOG**
Db2 requests the zHyperLink protocol only for active log write I/Os.

Note: zHyperLink Db2 read supports up to 4 KB page sizes, in multiples of 512 bytes whereas zHyperLink write supports multi-track I/Os.

The entries in the DSNTIPA3 panel define storage management preferences for the storage groups and data sets that are created during installation and migration. The zHyperLink parameter is included in that session, as shown in Figure 2-15.

```

DSNTIPA3          INSTALL DB2 - DATA PARAMETERS PANEL
===> _

Check parameters and reenter to change:
 1 PERMANENT UNIT NAME  ==> 3390      Device type for MVS catalog
                                and partitioned data sets
 2 TEMPORARY UNIT NAME  ==> SYSDA      Device type for
                                temporary data sets
 3 DB2 zHyperLinks SCOPE ==> ENABLE    Scope of zHyperLinks I/O connections:
                                (DISABLE, ENABLE, DATABASE)

                                ----- SMS -----
                                VOL/SER  DATA CLASS  MGMT CLASS  STOR CLASS
                                -----
 4 CLIST ALLOCATION      ==>           ==>           ==>           ==>
 5 NON-VSAM DATA       ==>           ==>           ==>           ==>
 6 VSAM CATALOG, DEFAULT, AND WORK FILE DATABASE ==>           ==>           ==>
 7 LOG COPY 1, BSDS 2   ==>           ==>           ==>           ==>
 8 LOG COPY 2, BSDS 1   ==>           ==>           ==>           ==>

```

Figure 2-15 Db2 parameters panel

Note: When Metro Mirror is used, it is recommended to enable the REMOTE_COPY_SW_ACCEL parameter, as shown in the following example (default value is **DISABLE**):

```
Db2 REMOTE_COPY_SW_ACCEL=ENABLE
```

For more information about how to configure your Db2 to explore zHyperLink, see [IBM Documentation](#).

2.5 Performance planning

In this section, we describe performance considerations when zHyperLink is activated. System specialists must identify applications that can use zHyperLink and decide to explore (and eventually use) it.

Thorough performance planning is helpful for the following reasons:

- ▶ The installation might determine that the potential incremental CPU cost that is associated with zHyperLink cannot be justified, even with the reduced latency that synchronous I/O provides.
- ▶ The installation might want to disable zHyperLink because the cache hit ratio for a class of data sets does not warrant its use.

2.5.1 CPU considerations

zHyperLink can reduce transaction time by half (up to 50%). It is highly advised to be used when critical business depends on transaction time. However, the CPU cost is not neutral for some environments, which requires performance planning to project what customers might observe with Db2 or VSAM.

As of this writing, the system settings allow any VSAM application or Db2 table space to read by using zHyperLink. By introducing fine-grained controls (for example, at the Db2 table space or a VSAM data set), users can choose the CPU consumption versus the response time trade-off that is wanted.

2.5.2 Db2 performance considerations

Db2 APARS for performance add indicators on Db2, which improves performance capacity planning. The following counters were created in the Instrumentation Facility Component Identifier (IFCID) 003 to represent DASD cache hits and successful zHyperLink read I/Os:

- ▶ QWACAWTD
Wait time for synchronous read I/O operations that resulted in disk cache hits.
- ▶ QWACAWCD
Number of synchronous read I/O operations that resulted in disk cache hits.

Messages DSNB455I and DSNB411I are changed to display statistics that show read I/O delays when zHyperLink is used.

The DSNB411I message is the result of the **DISPLAY BUFFERPOOL DETAIL** command, which displays detailed information about the buffer pool, including synchronous I/O reads that are performed by zHyperLink. A sample output from the **DISPLAY BUFFERPOOL DETAIL** command is shown in Example 2-6.

Example 2-6 DISPLAY BUFFERPOOL DETAIL command output

```
DSNB455I - SYNCHRONOUS I/O DELAYS -
  AVERAGE DELAY          = xx
  MAXIMUM DELAY          = xx
  TOTAL PAGES             = xx
SYNCHRONOUS I/O DELAYS WITH ZHYPERLINK
  AVERAGE DELAY          = xx
  MAXIMUM DELAY          = xx
```

```
TOTAL PAGES          = xx

DSNB411I - RANDOM GETPAGE = xx
          SYNC READ I/O (R) = xx
          SEQ. GETPAGE = xx
          SYNC READ I/O (S) = xx
          SYNC READ I/O (ZHL) = xx
          DMTH HIT = xx
          PAGE-INS REQ = xx
          SEQUENTIAL = xx
          VPSEQT HIT = xx
          RECLASSIFY = xx
```

Note: To reduce I/O latency, it is advised that I/Os are limited to a Control Interval (CI) size of 4 KB.

2.5.3 Measuring before deployment

In this section, we describe measuring performance by using RMF and the estimation of future performance by using the zBNA tool.

When implementing major updates to the system or configuration, performance must be measured before and after the update. A similar situation applies to implementing zHyperLink. However, because zHyperLink is intended to improve synchronous I/O, focus must be on the eligible workload and not the overall performance.

In general, performance comparisons are always be done on comparable workloads, such as the same day of the week and month or a batch window on the weekend.

Note: zHyperLink easily can be turned on and off by using system commands; therefore, you can compare performance with and without zHyperLink repeatedly, if needed. For more information about which commands to use, see 4.1, “System commands” on page 80.

Measuring by using RMF

We focus on measuring the DASD subsystem cache and the FICON Channel usage. The Subsystem Cache is a focus because that is the target for future zHyperLink I/Os. The FICON Channel usage also is a focus because it can detect the change that is caused by some I/Os to being off-loaded from FICON to zHyperLink.

RMF Monitor III offers the following reports that assist you in monitoring the performance of your cache subsystem. They present the combined I/O activity that is coming from any z/OS system (in the same or in different sysplexes) from or to the controller:

- ▶ Cache Summary Report (CACHSUM)
- ▶ Cache Detail Report (CACHDET)

By using the reports, you can analyze cache rates at the Storage Subsystem ID (SSID) and volume level.

An alternative to the interactive analysis with RMF Monitor III is the RMF Postprocessor Cache Subsystem Activity Report. You can run the report by using the **REPORTS(CACHE(SUBSYS))** sysin statement. A JCL example is provided later in this section (see Figure 2-16 on page 43).

By using the z/OS Workload Manager (WLM), you can define a service and report class for each Db2 region. The use of a separate report class for the Db2 region for which you are about to implement zHyperLink helps you limit reporting to the workload that is to benefit from zHyperLink implementation. A similar report is available in the RMF Spreadsheet reporter.

The Workload Activity report (WLMGL) can be used to request various reports, including the Service Class report and the Report Class report, which are nearly identical. The WLMGL report shows the response time distribution and the goal versus actual.

The RMF WLMGL report also is available in XML output format. For more information about how to produce and view XML reports, see “How to work with Postprocessor XML reports” in the *IBM z/OS RMF Users Guide*, [SC27-4936](#).

A sample JCL that is used to print a Workload Activity report is shown in Figure 2-16.

```
//RMFPP EXEC PGM=ERBRMFPP,REGION=OM
//*STEPLIB DD DISP=SHR,DSN=CEE.SCEERUN
//MFPINPUT DD DISP=SHR,DSN=*.RMFSORT.SORTOUT
//MFPMSGDS DD SYSOUT=*
//*****
//*
//* RMF POSTPROCESSING OPTIONS GENERATED FROM
//* POSTPROCESSOR OPTIONS PANEL INPUT ONLY.
//*
//*****
//SYSIN DD *
        SYSOUT(0)
        SYSRPTS(WLMGL(RCLASS(report_class)))
//
```

Figure 2-16 Sample JCL to print the RMF Workload Activity Report

To measure the FICON channel that is used for the DASD subsystem in question, you can use the RMF Channel Path Activity report (CHANNEL). This report provides more information about channel path activity for all channel paths in the system.

As with the Cache subsystem reporting, the use of the RMF Postprocessor alternative is available. Specify **REPORTS(CHAN)** as sysin for a Channel Activity Report.

For more information about RMF reporting, see *Resource Measurement Facility Report Analysis*, [SC34-2665](#).

Measuring by using zBNA

The zBNA Average response Time Graph shows the current and estimated I/O response times. The report can be filtered by Service classes and job names.

The report provides a current time of Average Re-Dispatch and Response time, which is the sum of Average I/O Response and user-specified Re-Dispatch time. It also provides an estimate of zHyperLink Estimated Re-Dispatch and Response times. This report factors in zHyperLink Eligible DASD Cache Hits and the elimination of Re-Dispatch Time for those hits.

Another useful zBNA report is the zHyperLink I/O Rate report that documents I/Os per second for workloads that use zHyperLink and further eligible zHyperLink workloads. The report also documents the highest potential I/O rate that is estimated to be achievable if all zHyperLink workloads were enabled.

Use the reports for documenting current I/O response times and I/O rates before implementing zHyperLink.

Note: The zBNA tool is designed to estimate the potential effect of moving to new technologies can have on batch elapsed time. zBNA is not designed for measuring overall system performance. When creating a zBNA report multiple times, it cannot be guaranteed that all variables are the same for each report.

For more information about the zBNA tool, see 2.1, “Identifying zHyperLink eligible workloads” on page 16.

2.6 Monitoring

In this section, we describe how to monitor the operation of zHyperLink by using z/OS display commands over the devices that are enabled for zHyperLink, PCIe adapters, virtual functions, and DS800 subsystems. This task is important following zHyperLink activation after the system is activated and before production data is sent through zHyperLink, and also for monitoring further use of zHyperlink.

2.6.1 Monitoring PCIe

In this section, we describe how to display and monitor PCIe adapters. Some important display commands and new information were added to display PCIe commands.

D PCIE

The **D PCIE** command shows the information that is registered for the device type with PCIe. It displays all PCIe devices, including zHyperLink. Each PCIe Function Id (PFID) is displayed in a single line; that is, if you have multiple PFIDs that are defined for each physical connection, one line is displayed for each PFID.

This command also displays the device type name, status, ASID, and JOBNAME that is allocating the connection, Channel Id, Virtual Function Number, and Port Number. For more information about the fields, see 4.1, “System commands” on page 80.

D PCIE,PFID=pfid

The **D PCIE,PFID=pfid** command displays more information about a single zHyperLink connection, such as software status, port state, and CU Node description. For more information about the fields that are available for the **D PCIE,PFID=pfid** command, see 4.1, “System commands” on page 80.

2.6.2 Monitoring subsystem

First, identify the serial number of the subsystem to be monitored. Use the device services command **DS QD,nnnn** where *nnnn* is the device number to display the serial number.

Then, issue the **DS QD,MACH=nnnn-xxxxx,ZFL** command, where *nnnn-xxxxx* is the serial number to display the number of concurrent write sessions at subsystem level.

Note: The display is not limited to the system on which the command is issued. The maximum number of concurrent write sessions is 64.

2.6.3 Monitoring devices

In this section, we describe how to monitor devices that are enabled for zHyperLink. We also describe how to display more information about those devices.

D IOS,ZHYPERLINK | ZHL

The display IOS command with the **ZHYPERLINK** parameter shows if zHyperLink is **ENABLED** or **DISABLED**. When zHyperLink is **ENABLED**, it describes what I/O operation is available to use zHyperLink (**READ|WRITE**). In specific cases where hardware requirements were not met, it might display “**NOT SUPPORTED BY THE PROCESSOR**” instead.

For more information about the **D IOS,ZHYPERLINK** command, see 4.1, “System commands” on page 80.

D IOS,ZHYPERLINK | ZHL,LINK=ALL

The display IOS command with the **ZHYPERLINK, LINK=ALL** parameters provides an overview of PDHIDs, ports, port status, and the subsystem serials to which they are connected.

For more information about the **D IOS,ZHYPERLINK, LINK=ALL** command, see 4.1, “System commands” on page 80.

D IOS,ZHYPERLINK | ZHL,LINK=pchid.port

The display IOS command with the **ZHYPERLINK, LINK=pchid.port** provides the same information as when the **LINK=ALL** parameters are used, but is limited to a specific PCHID and port.

For more information about the **D IOS,ZHYPERLINK, LINK=pchid.port** command, see 4.1, “System commands” on page 80.

D IOS,ZHYPERLINK | ZHL,LINK=pchid.port,DETAIL

The display IOS command with the **ZHYPERLINK, LINK=pchid.port, DETAIL** provides more information about the port status; for example, if the port is degraded. Also counters for success, and link busy and time-out for both read and write operations are displayed; however, only if counters are nonzero.

For more information about the **D IOS,ZHYPERLINK, LINK=pchid.port, DETAIL** command, see 4.1, “System commands” on page 80.

D M=DEV(devno)

Use the **D M=DEV(devno)** command to display device-related information. The command output now also includes zHyperLink-related information. It can be used to check the number and status of paths to the device, and the number of connections and status of zHyperLink.

You also can check the functions that are enabled for the device, including the zHyperLink options Read, Write, and All.

For more information about the **D M=DEV(devno)** command, see 4.1, “System commands” on page 80.

D M=DEV(devno),ZHYPERLINK

The **D M=DEV(devno), ZHYPERLINK** command displays more information that is related to zHyperLink. The information that is available in this command also includes information about what functions are enabled, or the reason why they are disabled.

You can plan to periodically issue this command to monitor your zHyperLink status. For more information about the status, see 4.1, “System commands” on page 80.

D M=CU(cuno)

The **D M=CU(cuno)** command includes output that is similar to the **D M=DEV(devno)** command. It also includes zHyperLink function information, including the PFID, PCHID, LinkID, Software, and Port status.

For more information about each field, see 4.1, “System commands” on page 80.



Deployment

This chapter describes the deployment considerations when implementing zHyperLink. It also covers monitoring system availability and measuring the benefits of implementing zHyperlink.

This chapter includes the following topics:

- ▶ 3.1, “Hardware deployment” on page 48
- ▶ 3.2, “Software upgrade and maintenance” on page 61
- ▶ 3.3, “zHyperLink enablement” on page 63
- ▶ 3.4, “Monitoring” on page 69
- ▶ 3.5, “Activating zHyperLink for VSAM data sets” on page 70
- ▶ 3.6, “Activating zHyperLink for Db2” on page 72
- ▶ 3.7, “Measuring after deployment” on page 73

3.1 Hardware deployment

In this section, we describe how to implement the hardware part of zHyperLink on the IBM z15 connected to the DS8900 storage system. The physical hardware components, DS8900 settings, and IBM z15 I/O configuration also are covered.

In this section, we describe how to implement the hardware part of zHyperLink on the IBM z15 and on the DS8900. The physical hardware components, DS8900 settings, and IBM z15 I/O configuration also are covered.

Tip: The components and configuration for IBM z15 and DS8880 series are similar to IBM z15 and DS8900 series (feature codes might differ across server and storage generations).

3.1.1 Ordering features

Order features for the IBM Z CPC and the DS8000 plus cables from your IBM representative.

For the IBM z15, the IBM zHyperLink Express is a two-port feature in the PCIe I/O drawer. You can install a maximum of 16 features with two zHyperLink Express ports each (FC #0451). Because zHyperLink connections are required to be configured in pairs, the minimum number of required features to avoid a single point of failure (SPOF) is two (2 x 2 ports).

For the DS8000, the feature includes one port in an I/O Bay enclosure, and is always installed in pairs. Also, two I/O Bay enclosures are updated at a time, which results in increments of four ports. The DS8900 feature code for the zHyperLink adapter is FC #3500.

For the DS8880, depending on your configuration, other feature codes might be required (for example, FC #4235 for 256 GB system memory, and FC #4425 for 12 core POWER processors). The supported zHyperLink features for DS8000 models are listed in Table 2-1 on page 27.

Order 24x MTP-MTP cables for connecting the IBM Z CPC to the DS8900. Cables are available as OM3 and OM4 fiber optic cables.

The same operation also must be done on the secondary DS8000s if you are using metro-mirror and want to use zHyperLink writes.

Tip: Cables up to 150 meters (492 feet) are supported. Because the cable length affects latency, cables are generally ordered as short as possible.

After ordering the features, configuration files for the future update of IBM Z CPC and DS8000 are available. You need the configuration files for ordering connection of cables, as described in 3.1.4, “Installing features” on page 50, and for the I/O configuration update, as described in 3.1.6, “Updating I/O configuration” on page 52.

For more information about feature codes and cables, see 2.3, “zHyperLink hardware planning” on page 29.

3.1.2 Updating microcode

The microcode at both ends of the point-to-point connection (the IBM Z CPC and the DS8000 DASD subsystem) must be updated to support zHyperLink. If the microcode is not at the required level, contact your IBM support representative and request the microcode update.

The following microcode levels are required:

- ▶ For IBM z14, the recommended level for zHyperLink is GA2 D36C Bundle S08 or higher.
- ▶ For IBM z15, no recommended level that is specific to zHyperLink is available.

Updating microcode on the IBM Z CPC and the DS8000 are nondisruptive activities.

For writes, both DS8000 clusters must be quiesced or resumed to allocate NVS¹ for zHyperlink. This process is nondisruptive. I/O priority manager must be disabled (DS8000) and System z® CUIR support must be enabled.

For writes, asynchronous replication is supported on the DS8900 models only, starting with the 9.1 firmware level.

3.1.3 Enabling the DS8000 zHyperLink feature

The DS8000 z-synergy Service License must be available on the DS8000. You can obtain your license activation codes from the IBM data storage feature activation from the [\(DSFA\) website](#), or contact your IBM support representative and request that the license is enabled.

Note: The DS8000 z-synergy Service License also includes parallel access volumes (PAVs), HyperPAV, Transparent Cloud Tiering, and other functions; therefore, it is likely to be available on your DS8000 if you are implementing zHyperLink on a DASD subsystem that is used for IBM Z systems.

Then, verify that the zHyperLink feature is enabled on the DS8000. Feature enablement is done by using the GUI or the DS8000 Series command-line interface (DS CLI). In the following example, we use the GUI for verifying enablement of the zHyperLink function on the DS8000.

Complete the following steps:

1. Access the GUI by using your browser and pointing to the GUI URL `https://<HMC_IP>`, where `<HMC_IP>` is the DS8000 Hardware Management Console (HMC) IP address. This Storage Management GUI URL is configured by the IBM support representative during the installation process.
2. Log on by using your login credentials.
3. After logging on, select the **Settings** menu and then, the **System** option.

¹ NVS - nonvolatile storage (DS8000)

4. Select the **Licensed Functions** tab, which brings you to the list of functions, as shown in Figure 3-1. The IBM Z Synergy function must be enabled.

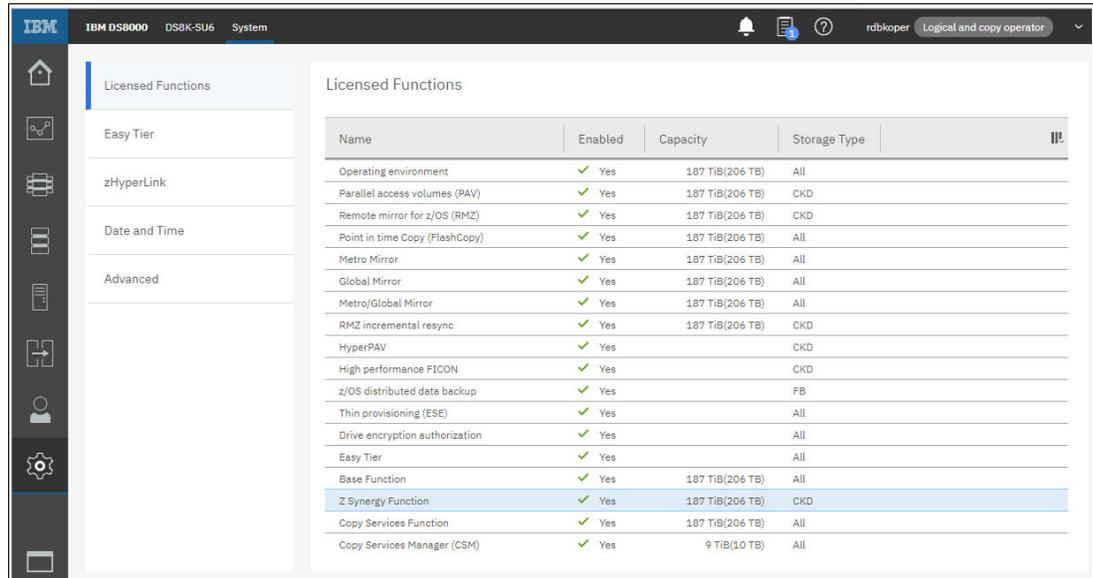


Figure 3-1 DS8000 GUI Licensed Functions window

Installing the DS8000 Z Synergy Service license and enabling the IBM Z Synergy Function are nondisruptive activities.

For more information about the GUI, see the *IBM DS8900F Architecture and Implementation Release 9.2*, [SG24-8456](#).

3.1.4 Installing features

When the ordered features are delivered, the IBM service representative installs the features. Detailed instructions for installing the features are provided for the IBM service representative as part of the feature delivery.

After installing the features in the IBM Z CPC and the DS8000, cables can be connected. Avoid SPOFs when deciding which ports to use on the IBM Z CPC and the DS8000. Spread connections over as many different cards on both devices as possible. For more information, see 2.2.3, “I/O connectivity planning” on page 29.

Tip: Documenting the specific ports that each cable is connected to at both ends makes future troubleshooting and updates to the setup easier.

Installing the features and connecting the cables are nondisruptive activities.

3.1.5 Activating zHyperLink on the DS8000

Complete the following steps to enable the zHyperLink option on the DS8000 DASD subsystem before enabling it on the z/OS side:

1. Log on to the DS8000 Storage Management GUI with administrator privileges.
2. Select the **Settings** menu and then, the **System** option.

3. Select the **Advanced** tab.
4. Disable the I/O Priority Manager mode (Function setting) and verify that the CUIR support (under IBM System z) is enabled, as shown in Figure 3-2.

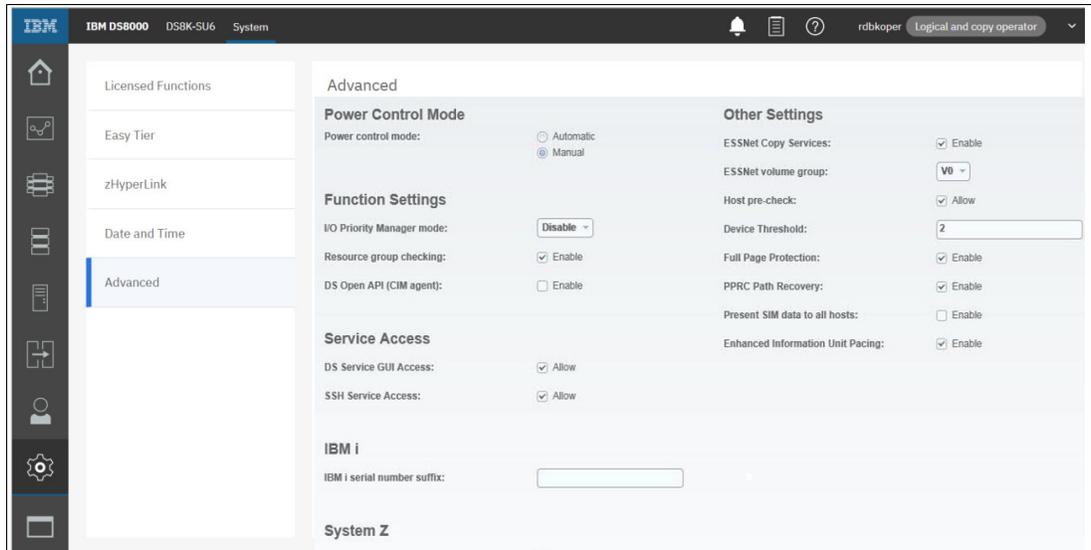


Figure 3-2 DS8000 GUI Advanced System Settings

If I/O Priority Manager is enabled on your storage system, you cannot use a zHyperLink connection. CUIR² enablement is needed for maintenance and for adding zHyperLink features by way of a miscellaneous equipment specification (MES).

5. Select the **zHyperLink** tab and then, select the **I/O Read Enabled** or **I/O Read Write** options, as shown in Figure 3-3.

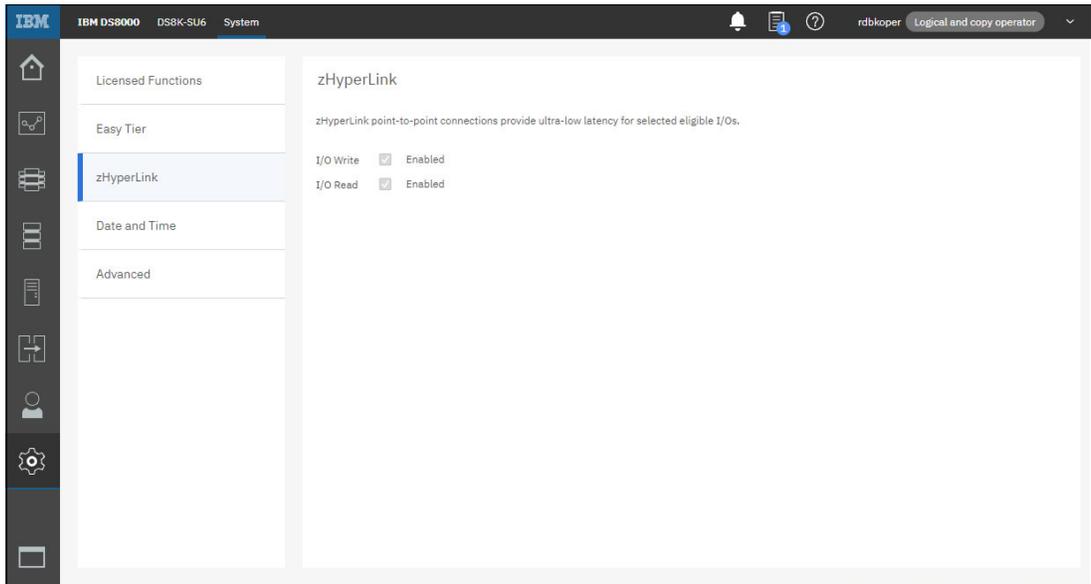


Figure 3-3 DS8000 GUI zHyperLink

² DS8000 Control-unit initiated reconfiguration (CUIR)

What happens when you enable zHyperlink on your DS8000 system?: When enabling zHyperlink write on the DS8000 for the first time, it will quiesce and resume both LPARs. Successive zHyperlink enables/disables of either write or read have no affect on DS8000 LPAR operation.

For more information about the DS8000 Storage Management GUI, see *IBM DS8900F Architecture and Implementation Release 9.2*, [SG24-8456](#).

3.1.6 Updating I/O configuration

In this section, we describe how to define and implement the I/O Configuration updates for zHyperLink PCIe features on the IBM z14 or IBM z15 CPC.

Because zHyperLinks are not defined on the control unit, you can continue to define up to eight CHPIDs per logical control unit (LCU). zHyperLink does not reduce that number. Association between the IBM Z CPC firmware and the DS8000 storage subsystem is discovered during link initialization. z/OS associates zHyperLinks with devices at IPL or vary online time.

Updating I/O definition file

Review the CFReport file that is supplied by your IBM support representative after ordering the features that are installed on your IBM Z CPC. The file contains information about the Physical Channel IDs (PCHIDs) for the installed zHyperLinks features.

In this example, we define two PCIe zHyperLink (HYL) features, which are shared by two partitions on the CPC. On each of the two features, we define four virtual functions on port 1 for both partitions, for a total of 16 virtual functions (because they cannot be shared). The process that is used to define virtual functions for port 2 is similar, but not included in the example.

We use Hardware Configuration Definition (HCD) for the definitions, but Hardware Configuration Manager (HCM) also can be used.

The zHyperLink part of the resulting IOCDs from this example is listed in Appendix C, “Sample IOCDs” on page 121.

Note: Unlike non-PCIe channels, such as OSA and FICON, PCIe functions are defined at the processor level, not at the Logical Channel Subsystem (LCSS) level.

To start the process, complete the following steps:

1. Enter HCD, specify the work IODF and select **option 1, Define, modify, or view configuration data**.
2. Select **option 3, Processors on the Define, Modify, or View Configuration Data panel**. The Processor List panel opens (Figure 3-4).

```

Processor List          Row 1 of 3 More:  >
Command ==> _____ Scroll ==> PAGE

Select one or more processors, then press Enter. To add, use F11.

/ Proc. ID Type +   Model +   Mode+ Serial-# + Description
_ ARIES   8561   T01   LPAR  0123458561 Aries
_ HYDRA   8562   LT2   LPAR  03F7A88562 Hydra
_ LEPUS   3907   ZR1   LPAR  0BB4B73907 Lepus
***** Bottom of data *****

```

Figure 3-4 Processor list

3. Select **Work with PCIe functions** by using the F line-command next to the CPC.
4. To define the zHyperLink PCIe function, press **F11** on the PCIe Function List panel.

Figure 3-5 shows the Add PCIe Function panel.

```

+----- Add PCIe Function -----+
|
| Specify or revise the following values.
|
| Processor ID . . . . . : ARIES      Aries
|
| Function ID . . . . . : 0100
| Type . . . . . : ZHYPERLINK +
|
| Channel ID . . . . . : 13C +
| Port . . . . . : 1 +
| Virtual Function ID . . . . . : 1 +
| Number of virtual functions . . 4
| UID . . . . . : _____
|
| Description . . . . . : DS8950F - SU6 - ZOS23BAA_____
|
+-----+

```

Figure 3-5 HCD Add PCIe Function panel

The following values are displayed on the panel:

- Function ID (FID): Similar to the control unit ID that you define when devices were added to an OSA or FICON channel. It is the identifier of the channel and virtual function ID.
- Type: Identifies the PCIe function type; in this case, ZHYPERLINK.
- Channel ID: The zHyperLink function channel ID that is documented in the CFReport; similar to a PCHID for an OSA or FICON channel.
- Port: Can be 1 or 2 for a zHyperLink PCIe feature. In this example, we define port 1 only.

- Virtual Function ID: A number in the range of 1 to 127 because the maximum number of virtual functions for a zHyperLink feature is 254. Two ports with a maximum of 127 virtual functions each.

Note: The virtual function number must be unique at the feature port level, and cannot be shared by multiple partitions.

- Number of virtual functions: The number of simultaneous I/O operations that can be active on a zHyperLink. We recommend defining four virtual functions for a zHyperLink port per LPAR.
- UID: The User Defined Identifier that can be used for PCIe functions for partitions with the attribute (flag), which enables or disables UID-checking for a logical partition. Not applicable to zHyperLink.
- Description: The optional description of the definition. Consider specifying information on the DS8000 to which the zHyperLink is connected and the partition.

5. Press **Enter**. The Define Access List panel opens, as shown in Figure 3-6. Use this panel to select the partition that can access the virtual functions.

```

+----- Define Access List -----+
|                                     Row 22 of 55 |
| Command ==> _____ Scroll ==> PAGE |
|                                     |
| Select one partition for the access list. |
|                                     |
| Function ID . . . . : 0100 |
|                                     |
| / CSS ID Partition Name  Number Usage Description |
| _ 1    ARIES11           1     OS   ZSKVMPF2 |
| _ 1    ARIES12           2     OS   Z71PERF1 |
| _ 1    ARIES13           3     OS   Z71PERF2 |
| _ 1    ARIES14           4     OS   Z71PERF3 |
| _ 1    ARIES15           5     OS   Z71PERF4 |
| / 1    ARIES16           6     OS   ZOS23BAA |
| _ 1    ARIES17           7     OS   Z71PCC01 |
| _ 1    ARIES18           8     OS   Z71PCC03 |
| _ 1    ARIES19           9     OS   Z71CRUH1 |
+-----+

```

Figure 3-6 HCD Define Access List panel

A virtual function cannot be shared between partitions; therefore, you can select a single partition only.

6. Press **Enter**. The Define Candidate List opens. Optionally select one or more partitions.

- Press **Enter**. Four virtual functions for the LPZOSAPR partition are defined, as shown in Figure 3-7.

```

-----
                                PCIe Function List
Command ==> _____ Scroll ==> PAGE

Select one or more PCIe functions, then press Enter. To add, use F11.

Processor ID . . . . : ARIES      Aries

/ FID  CHID+ P+ VF+ Type+      UID  Description
_ 0100 13C  1  1  ZHYPERLINK  _____ DS8950F - SU6 - ZOS23BAA
_ 0101 13C  1  2  ZHYPERLINK  _____ DS8950F - SU6 - ZOS23BAA
_ 0102 13C  1  3  ZHYPERLINK  _____ DS8950F - SU6 - ZOS23BAA
_ 0103 13C  1  4  ZHYPERLINK  _____ DS8950F - SU6 - ZOS23BAA

```

Figure 3-7 HCD PCIe Function List panel

You can update the Access List and Candidate Access List by scrolling right, the same way as you can for channels like OSA and FICON.

Next, we define another four virtual functions for the same Channel ID and port, but assigned to another partition.

- Use the A line-command next to one of the Function IDs that are shown in Figure 3-7. The Add PCIe Function panel opens and the values that you specified for the first partition are displayed.

```

+----- Add PCIe Function -----+
|
| Specify or revise the following values.
|
| Processor ID . . . . : ARIES      Aries
|
| Function ID . . . . . 0104
| Type . . . . . ZHYPERLINK  +
|
| Channel ID . . . . . 13C  +
| Port . . . . . 1  +
| Virtual Function ID . . . . . 5  +
| Number of virtual functions . . 4
| UID . . . . . _____
|
| Description . . . . . DS8950F - SU6 - ZOS24BPR
|
+-----+

```

Figure 3-8 HCD Add PCIe Function panel

11. Press **Enter**. Four virtual functions for the ZOS24BPR partition are now defined, as shown in Figure 3-10.

```

                                PCIe Function List
Command ==> _____ Scroll ==> PAGE

Select one or more PCIe functions, then press Enter. To add, use F11.

Processor ID . . . . : ARIES          Aries

/ FID  CHID+  P+  VF+  Type+      UID  Description
_ 0100  13C    1   1   ZHYPERLINK  _____ DS8950F - SU6 - ZOS23BAA
_ 0101  13C    1   2   ZHYPERLINK  _____ DS8950F - SU6 - ZOS23BAA
_ 0102  13C    1   3   ZHYPERLINK  _____ DS8950F - SU6 - ZOS23BAA
_ 0103  13C    1   4   ZHYPERLINK  _____ DS8950F - SU6 - ZOS23BAA
_ 0104  13C    1   5   ZHYPERLINK  _____ DS8950F - SU6 - ZOS24BPR
_ 0105  13C    1   6   ZHYPERLINK  _____ DS8950F - SU6 - ZOS24BPR
_ 0106  13C    1   7   ZHYPERLINK  _____ DS8950F - SU6 - ZOS24BPR
_ 0107  13C    1   8   ZHYPERLINK  _____ DS8950F - SU6 - ZOS24BPR

```

Figure 3-10 HCD PCIe Function List panel

12. Repeat steps 8 - 11 when you are defining Functions IDs for the other Channel ID.

The following values are used when defining Function IDs:

- Function ID (FID): For the next channel ID, the example uses the starting value of 1008. The Function ID must be unique within the I/O Configuration Definition File (IODF).
- Type: Identifies the PCIe function type; in this case, ZHYPERLINK.
- Channel ID: The zHyperLink function channel ID that is documented in the CFReport, which is similar to a PCHID for an OSA or FICON channel. In the example configuration, the other zHyperLink Channel ID is 1A8.
- Port: The example includes port 1 only.
- Virtual Function ID: Because we are now defining the second Channel ID, we can use the same Virtual Functions IDs as we did for the first Channel ID; that is, 1 - 4 and 5 - 8.
- Number of virtual functions: Define four virtual functions for a zHyperLink port as for the first channel ID.
- UID: Still not used.
- Description: Similar to the text that we used for the first Channel ID.

After defining Function IDs for the second Channel ID, the list of defined Functions is as shown in Figure 3-11.

```

Goto  Filter  Backup  Query  Help
-----
                                PCIe Function List  Row 169 of 351 More:  >
Command ==> _____ Scroll ==> PAGE

Select one or more PCIe functions, then press Enter. To add, use F11.

Processor ID . . . . : ARIES          Aries

/ FID   CHID+  P+  VF+  Type+      UID  Description
- 0100   13C   1   1   ZHYPERLINK  ____ DS8950F - SU6 - ZOS23BAA
- 0101   13C   1   2   ZHYPERLINK  ____ DS8950F - SU6 - ZOS23BAA
- 0102   13C   1   3   ZHYPERLINK  ____ DS8950F - SU6 - ZOS23BAA
- 0103   13C   1   4   ZHYPERLINK  ____ DS8950F - SU6 - ZOS23BAA
- 0104   13C   1   5   ZHYPERLINK  ____ DS8950F - SU6 - ZOS24BPR
- 0105   13C   1   6   ZHYPERLINK  ____ DS8950F - SU6 - ZOS24BPR
- 0106   13C   1   7   ZHYPERLINK  ____ DS8950F - SU6 - ZOS24BPR
- 0107   13C   1   8   ZHYPERLINK  ____ DS8950F - SU6 - ZOS24BPR
- 0108   21C   1   1   ZHYPERLINK  ____ DS8900F - SU6 - ZOS23BAA
- 0109   21C   1   2   ZHYPERLINK  ____ DS8900F - SU6 - ZOS23BAA
- 010A   21C   1   3   ZHYPERLINK  ____ DS8900F - SU6 - ZOS23BAA
- 010B   21C   1   4   ZHYPERLINK  ____ DS8900F - SU6 - ZOS23BAA
- 010C   21C   1   5   ZHYPERLINK  ____ DS8900F - SU6 - ZOS24BPR
- 010D   21C   1   6   ZHYPERLINK  ____ DS8900F - SU6 - ZOS24BPR
- 010E   21C   1   7   ZHYPERLINK  ____ DS8900F - SU6 - ZOS24BPR
- 010F   21C   1   8   ZHYPERLINK  ____ DS8900F - SU6 - ZOS24BPR
- 0200   13C   2   1   ZHYPERLINK  ____ ZOS23BAA

```

Figure 3-11 HCD PCIe Function List panel

Note: When you define zHyperLink for port 2, the Function ID still must be unique at the processor (CPC) level within the IODF; however, the Virtual Function IDs are identical to the IDs that were defined for port 1. Function IDs for port 2 can be assigned to the same or different partitions as the IDs that were defined for port 1.

In our example, we defined zHyperLinks between a single CPC and a single DS8000 by using a total of two CPC features and four ports. Continue defining similar connections for other CPCs, LPARs and DS8000 subsystems that are to use zHyperLink.

You have the choice of using the same or different Function IDs when defining zHyperLink connections for multiple CPCs. The Function ID must be unique at processor level within the IODF. Remember to check the CFReport for each CPC to have zHyperLinks defined, as the Physical Channel IDs (PCHIDs) might be different.

PFIDs also must be defined for connectivity to the secondary DS8000s if you use Metro Mirror and want to use zHyperLink writes.

Implementing I/O definition file

After updating the I/O definition file (IODF), follow your standard procedure for building and implementing the new production IODF.

The zHyperLink PCIe functions do not include updates to the Operating System Configuration part of the IODF.

3.1.7 Verifying zHyperLink cables

Verify zHyperLink cable connections after implementing the hardware, as described in 3.1, “Hardware deployment” on page 48. Start the verification process by issuing system commands on the z/OS system, and then, continue by verifying on the HMC.

Verifying zHyperLink cables on z/OS

Complete the following steps to verify the zHyperLink cables on the z/OS system:

1. Display PCIe functions by using the **D PCIe** system command. Consider the following points:
 - The PFID column represents the Function ID and the PN column represents the port.
 - The device Type name is 8 GB zHyperLink.
 - At this stage, the Status column should display STNBY because the functions are not yet online.
2. Configure the Function online by using the **CF PFID(xxxx),ON** system command.
3. Display PCIe functions again by using the **D PCIe** system command. The Status is changed to CFNG. After enabling zHyperLink at the z/OS level later, zHyperLink is allocated or in use, and the Status is then ALLC.

Verifying on HMC

Complete the following steps to verify on the HMC:

1. Log on to the HMC by using your browser and pointing to the HMC URL, https://<HMC_IP>.
2. Select the processor, the **Recovery task group** and then, the **Single Object Operations** task, which brings you to the Support Element.
3. At the Support Element, select **Systems Management**, the processor and **Channels**. Locate the first Channel ID for zHyperLink.
4. Select the **Channel**, the **Channel Operations group** and then the **Advanced Facilities** task. Select **Card Specific Advanced Facilities** and then, **View Port Parameters**.
5. Select the port number.

A sample HMC Advanced facilities window for a zHyperLink channel 021C port 1 is shown in Figure 3-12.

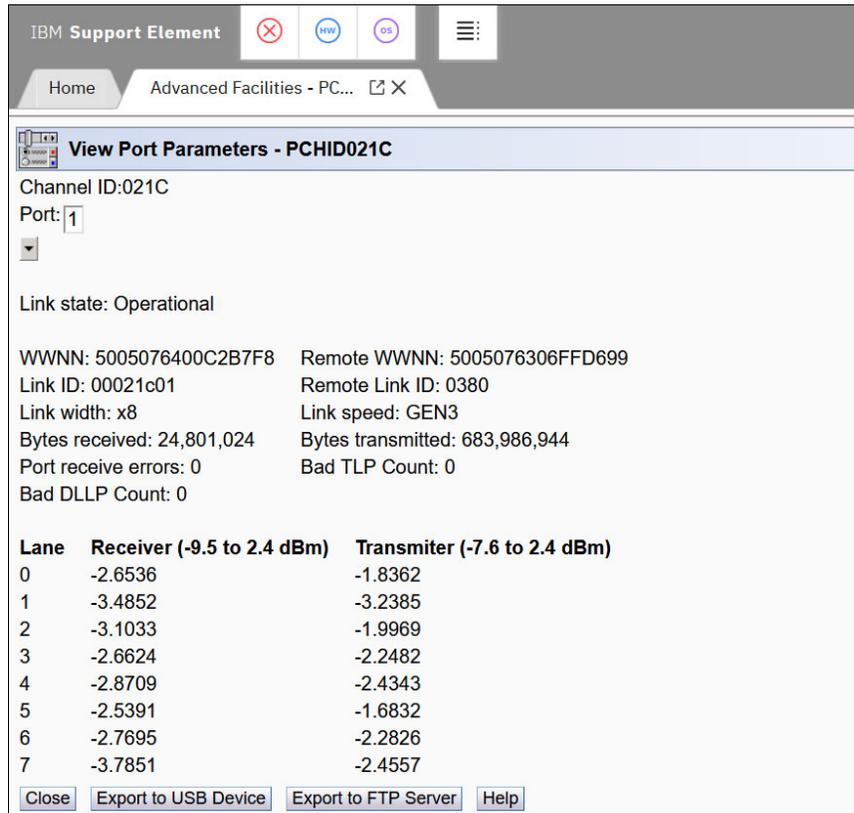


Figure 3-12 HMC Advanced Facilities window

6. Verify that a Remote worldwide node name (WWNN) is displayed. The WWNN is displayed if the WWNN was read when a path first came online to the LCU.
The Remote Link ID should display the physical zHyperLink port ID on the DS8000.
The remove Link Speed should be GEN3. If the Remote Link speed is GEN1 or GEN2, the link speed was degraded.
7. Verify the dBm values for the eight receiver lanes. A valid range for the Optics Receiver is -9.5 dBm - +2.4 dBm.
8. Verify the dBm values for the eight transmitter lanes. A valid range for the Optics Transmitter is -7.6 dBm - +2.4 dBm.

Repeat these steps for all installed zHyperLink Channel IDs and ports.

3.2 Software upgrade and maintenance

In this section, we describe deployment steps for zHyperLink concerning the software requirements as described in Chapter 2, “Planning” on page 15. We also describe the process of requesting the products and fixes that are required for zHyperLink.

3.2.1 Security profiles

The infrastructure for PCIe I/O services is provided by the PCIe address space, which is automatically started during IPL. Make sure a security profile for the PCIe task exists. A S0D6 abend occurs if no valid security profile is found for the PCIe task during IPL. A default user ID is sufficient for zHyperLink PCIe.

3.2.2 Product upgrades

When products for z/OS platform are to be installed or upgraded, the method for ordering the product and maintenance is through internet orders by using IBM ShopzSeries (Shopz). When internet orders are used, the package is immediately available for download after the order is processed. If you are not a Shopz user, an account must be created. For more information, see the [ShopzSeries website](#).

Internet and DVD delivery options are available. Ensure that you have the suitable network connectivity and firewall set-up if you are planning to use the Internet Delivery option. To verify your system setup for internet delivery of orders, see the [Connectivity Test for SW Download Readiness website](#).

The required space for your order in the download file system is provided on the Shopz download page in the Required information for ServerPac Dialog file. The download page can be found on the My Orders tab of the [ShopzSeries website](#) (a My Orders tab is generated only if a user logs in and creates an order).

The ServerPac provides you with the following libraries:

- ▶ Distribution libraries (DLIBs)
- ▶ Target libraries (TLIBs)
- ▶ System Management Program Extended (SMP/E) libraries

For more information about Shopz, see the [Shopz User's Guide](#).

ServerPac is available in a compacted (**TERSED**) file format. After downloading the ServerPac, an **UNTERSE** job must be run to unpack it.

Two programs are available to **UNTERSE** files: AMATERSE and TRSMAIN. A sample JCL to **UNTERSE** the IBMUSER.SERVRPAC.TERSED file by using the TRSMAIN program is shown in Example 3-1.

Example 3-1 Sample UNTERSE job

```
//UNTERSEQ JOB ( , ),REGION=OM,NOTIFY=&SYSUID,CLASS=A
//*****
//UNTERSEQ EXEC PGM=TRSMAIN,PARM=UNPACK
//SYSPRINT DD SYSOUT=*
//INFILE DD DSN=IBMUSER.SERVRPAC.TERSED,
// DISP=SHR
//OUTFILE DD DSN=IBMUSER.SERVRPAC.UNTERSED,
// DISP=(NEW,CATLG),
```

```
//          SPACE=(CYL,(1,1))
```

For more information about references to **AMATERSE**, see [IBM Documentation](#).

With **UNTERSE** being complete, you can start performing **RECEIVE**, **CHECK**, and **APPLY** processes to your SMPE libraries.

The **SMPE RECEIVE** command takes a SYSMOD that is outside of SMP/E and stages it into the SMP/E library domain. For more information about examples of **SMPE RECEIVE**, see [IBM Documentation](#).

When **RECEIVE** is complete, **APPLY CHECK** is advised before performing the **APPLY**. The **CHECK** command verifies whether any error occurs when SYSMOD is applied in your **APPLY** command. This step simulates an **APPLY**; however, the target system libraries are not updated.

After completion of the **APPLY CHECK** process, the **APPLY** can be run. It specifies which received SYSMODs are to be installed in the target libraries. SMP/E ensures that all SYSMODs that are set as prerequisites were installed or are being installed concurrently in the correct order. For more information about examples of **APPLY JCL**, see [IBM Documentation](#).

For more information about syntax for SMPE commands, see [IBM Documentation](#).

When **APPLY** is completed, all of the downloaded SYSMODs are installed in your target libraries and ready for implementation.

3.2.3 Product maintenance

The use of the **MISSINGFIX** report is required to determine whether any APARs exist that are required to use a function, but are not yet installed. The **REPORT MISSINGFIX** command checks the zones that are specified on the **ZONES** operand and determines whether any missing fixes exist based on the fix categories specified.

Run the SMP/E **REPORT MISSINGFIX ZONES (<your zone names>) FIXCAT (IBM.Function.zHyperLink)** command and install all the maintenance that is listed. Example 3-2 shows a sample **MISSINGFIX** JCL.

Example 3-2 Sample SMP/e control cards with FIXCAT option

```
SET BOUNDARY (GLOBAL) .  
REPORT  
MISSINGFIX  
ZONES (<your zone names>)  
FIXCAT(IBM.Function.zHyperLink) .
```

Note: Run SMP/E **REPORT MISSINGFIX** for the following function fix categories: **IBM.Function.zHighPerformanceFICON** and **IBM.Function.zHyperLink**.

3.3 zHyperLink enablement

This section describes what must be checked before enabling zHyperLink, the updates that are required on system PARMLIB members, and commands that are used to activate zHyperLink in your z/OS environment.

3.3.1 Verifications before activation

Some checks must be made before zHyperLink is enabled.

Check whether the devices that are attached to the DS8000 with zHyperLinks are supporting zHyperLink at the z/OS side through z/OS display matrix (**D M=DEV**) command (see Example 3-3).

Example 3-3 Sample D M=DEV command

```
D M=DEV(9000)
IEE174I 15.24.35 DISPLAY M 528
DEVICE 09000 STATUS=ONLINE
CHP          40  42  41  43
ENTRY LINK ADDRESS 012C 012D 022C 022D
DEST LINK ADDRESS 010C 010B 020C 020B
PATH ONLINE      Y   Y   Y   Y
CHP PHYSICALLY ONLINE Y   Y   Y   Y
PATH OPERATIONAL Y   Y   Y   Y
MANAGED          N   N   N   N
CU NUMBER       0690 0690 0690 0690
INTERFACE ID    0030 0230 0100 0300
MAXIMUM MANAGED CHPID(S) ALLOWED: 0
DESTINATION CU LOGICAL ADDRESS = 90
SCP CU ND       = 002107.996.IBM.75.0000000LBN71.0030
SCP TOKEN NED   = 002107.900.IBM.75.0000000LBN71.9000
SCP DEVICE NED  = 002107.900.IBM.75.0000000LBN71.9000
WVNN           = 500507630AFFD049
HYPERPAV ALIASES CONFIGURED = 16
ZHYPERLINKS AVAILABLE = 2
FUNCTIONS ENABLED = MIDAW, ZHPF, XPAV, ZHYPERLINK
```

Confirm that both functions are enabled, zHPF and zHyperLink. For more information about each field, see 4.1, “System commands” on page 80.

Check the devices that are attached to your storage subsystem, and use the z/OS display command for more information about the I/O operation that is supported by them. The command that is shown in Example 3-4 provides more information about what was enabled for a specific device.

Example 3-4 Sample D M=DEV(dev),ZHYPERLINK

```
D M=DEV(9000),ZHEPERLINK
IEE587I 15.30.29 DISPLAY M 541
DEVICE 09000 STATUS=ONLINE
DEVICE IS ENABLED FOR ZHYPERLINK
READ AND WRITE OPERATIONS ARE ENABLED
ZHYPERLINKS AVAILABLE = 2
```

When an I/O operation is not supported by zHyperLink, the system displays the reasons through the IEE587I message. For more information about possible reasons, see 4.1, “System commands” on page 80.

As a result of the use of the display device command, you have Control Unit (CU) number information, which is CU NUMBER 0233 that is shown in Example 3-3 on page 63.

Use the **D M=CU(<number>)** command to display zHyperLinks stats on Control Unit, as shown in Example 3-5.

Example 3-5 Sample D M=CU(0233)

```

D M=CU(0690)
IEE174I 13.40.52 DISPLAY M 382
CONTROL UNIT 0690
CHP                40   42   41   43
ENTRY LINK ADDRESS 012C 012D 022C 022D
DEST LINK ADDRESS  010C 010B 020C 020B
CHP PHYSICALLY ONLINE Y   Y   Y   Y
PATH VALIDATED     Y   Y   Y   Y
MANAGED            N   N   N   N
ZHPF - CHPID      Y   Y   Y   Y
ZHPF - CU INTERFACE Y   Y   Y   Y
INTERFACE ID       0030 0230 0100 0300
MAXIMUM MANAGED CHPID(S) ALLOWED = 0
DESTINATION CU LOGICAL ADDRESS = 90
CU ND              = 002107.996.IBM.75.0000000LBN71.0030
CU NED             = 002107.996.IBM.75.0000000LBN71.9000
TOKEN NED          = 002107.900.IBM.75.0000000LBN71.9000
WNNN              = 500507630AFFD049
FUNCTIONS ENABLED = ZHPF, ZHYPERLINK, XPAV
XPAV CU PEERS     = 0690
DEFINED DEVICES
  09000-090EF
DEFINED PAV ALIASES
  090F0-090FF
USABLE HYPERPAV ALIASES = 16
ZHYPERLINKS
  PFID      PCHID  Port  LinkId  S/W St  Port St
  00000202 013C   02   0181   Alloc  Oper
  00000402 021C   02   0081   Alloc  Oper

```

The command that is shown in Example 3-5 provides information about the Control Unit. More information (highlighted in bold in Example 3-5) was added to the IEE174I message in support of zHyperLink.

The **ZHYPERLINKS** section on the command output includes processor-side information (PCHID and Port data), and control-unit side information (LinkId).

Only the online PFIDs are displayed, regardless of whether the link is usable. Offline (standby) PFIDs are not included in the command output.

For more information about the message output and a description of the displayed fields, see 4.3, “System messages” on page 93.

3.3.2 Activating zHyperLink

After completing the verification steps as described in 3.3.1, “Verifications before activation” on page 63, you are ready to start the deployment process.

The steps that are described in this section are required to activate zHyperLink on z/OS. In this section, we describe the process of enabling the zHyperLink facility, and the messages that are expected as response to the activation.

Static activation

This section describes system parameter updates and parameter checks that are required to support zHyperLink.

LOADxx

As described in 2.4.4, “Software activation plan for Db2” on page 40, **LOADxx** member at z/OS **PARMLIB** must be checked before you start using zHyperLink. The transactional-execution facility must be enabled for zHyperLink to work.

MACHMIG is an optional parameter that identifies facilities that you do not want z/OS to use after migration to another processor or z/OS version. How to code the **MACHMIG** statement is listed in Table 3-1.

Table 3-1 *MACHMIG* fields

Column	Content
1-7	MACHMIG
10-72	A list of facilities not to use. When more than one facility is listed, separate each from the previous by one or more blanks or commas.

The following facilities can be specified in upper, lower, or mixed case:

- ▶ EDAT2 (the hardware-based enhanced-DAT facility 2)
- ▶ TX (the hardware-based transactional-execution facility)
- ▶ RI (a hardware-based facility that is reserved for IBM use only)
- ▶ GSF (the hardware-based Guarded Storage Facility)

To allow zHyperLink to work, **MACHMIG** must not contain the TX statement value in the **LOADxx** member. The following example shows an **MACHMIG** definition where transactional-execution facility and enhanced DAT facility 2 are set *not* to be used:

```
----+----1----+----2  
MACHMIG TX, EDAT2
```

The TX statement is preventing zHyperLink enablement, and must be removed from the **MACHMIG** definition to implement zHyperLink and support synchronous I/O.

If the **MACHMIG** statement's **TX** option is not specified, the system does not prevent the use of zHyperlink because the default option is None.

IECIOSxx

The **PARMLIB** member (**IECIOSxx**) must be customized to support zHyperLink. The **IECOSxx** member contains I/O-related parameters that are specified by the installation.

An important requirement for zHyperLink is High Performance FICON for IBM z Systems (zHPF) to be enabled. It is defined at the IECIOSxx PARMLIB member with a **ZHPF** parameter.

The **ZHPF** parameter defines the condition of the **ZHPF** function. If no **ZHPF** statement is specified, the **ZHPF** facility is disabled on the system by default and zHyperLink is not supported.

The following options are available for the **ZHPF** parameter:

- ▶ ZHPF=YES
- ▶ ZHPF=NO

When the YES statement is defined, **ZHPF** is activated in the system after an IPL. This activation of **ZHPF** supports the use of zHyperLink. If NO is defined, **ZHPF** is disabled, which prevents support of zHyperLink.

Another important requirement for zHyperLink write is IBM zHyperWrite data replication (zHYPERWRITE) to be enabled. It also is defined at the IECIOSxx PARMLIB member with a **HYPERWRITE** parameter.

The **HYPERWRITE** parameter defines the condition of the **HYPERWRITE** function. If no **HYPERWRITE** statement is specified, the **HYPERWRITE** facility is enabled on the system by default.

The following options are available for the **HYPERWRITE** parameter:

- ▶ HYPERWRITE=YES
- ▶ HYPERWRITE=NO

When the YES statement is defined, **HYPERWRITE** is activated on the system after an IPL. This activation of **HYPERWRITE** is a prerequisite for the use of zHyperLink Write. If NO is defined, **HYPERWRITE** is disabled, which prevents support of zHyperLink Write.

The **ZHYPERLINK** statement in the IECIOSxx PARMLIB member is required to enable the zHyperLink facility. If no **ZHYPERLINK** statement is specified, the zHyperLink facility is disabled on the system by default, as shown in the following example:

```
ZHYPERLINK OPER=ALL|READ|WRITE|NONE
```

The defined value on the zHyperLink statement specifies whether the zHyperLink facility is enabled on the following operations:

- ▶ ALL I/O operations
- ▶ Read I/O operations
- ▶ Write I/O operations
- ▶ Disabled for all I/O operations

When **OPER=ALL** is defined, it activates zHyperLink for READ and WRITE I/Os. When **OPER=READ** is defined, you activate zHyperLink for READ I/O requests only. **OPER=WRITE** defines that WRITE I/Os are enabled only on the system. You disable zHyperLink by specifying the default of **OPER=NONE**.

Example 3-6 shows a sample IECIOSxx member that enables zHPF, HYPERWRITE and zHyperLink for both READ and WRITE I/Os.

Example 3-6

```
BROWSE    SYS1.PARMLIB(IECIOS00) - 01.18           Line 0000000000 Col 001 080
Command ==>                                         Scroll ==> PAGE
***** Top of Data *****
HYPERPAV=YES
```

```
HYPERWRITE=YES
ZHPF=YES
ZHYPERLINK,OPER=ALL
***** Bottom of Data *****
```

For more information about syntax rules for IECIOSxx, see [IBM Documentation](#).

Remember: The IOS system parameter updates that are done through **PARMLIB** require an IPL to take effect.

Dynamic activation

The use of z/OS system commands allows you to dynamically activate zHyperLink on z/OS systems. In this section, we describe the available commands for zHyperLink activation.

By using **SETIOS** system commands, you can dynamically set the values for ZHPF, HYPERWRITE and ZHYPERLINK without the need of an IPL.

SETIOS ZHPF

The zHPF option must be activated.

The following syntax for the **SETIOS** command changes the zHPF operation mode:

```
SETIOS ZHPF={YES|NO}
```

Next, the following command enables zHPF:

```
SETIOS ZHPF=YES
```

In response to the **SETIOS** command, the system displays the following message:

```
IOS090I SETIOS. ZHPF UPDATE(S) COMPLETE
```

You can verify the zHPF status by issuing the display IOS command:

```
D IOS,ZHPF
```

In response to the display zHPF command, the system replies with an IOS630I message:

```
IOS630I hh.mm.ss ZHPF FACILITY 485
HIGH PERFORMANCE FICON FACILITY IS ENABLED
```

SETIOS HYPERWRITE

The zHYPERWRITE data replication option must be activated.

The following syntax for the **SETIOS** command changes the HYPERWRITE operation mode:

```
SETIOS HYPERWRITE={YES|NO}
```

Next, the following command enables HYPERWRITE:

```
SETIOS HYPERWRITE=YES
```

In response to the **SETIOS** command, the system displays the following message:

```
IOS090I SETIOS. HYPERWRITE UPDATE(S) COMPLETE
```

You can verify the HYPERWRITE status by issuing the display IOS command:

```
D IOS,HYPERWRITE
```

In response to the display HYPERWRITE command, the system replies with an IOS634I message:

```
IOS634I hh.mm.ss IOS SYSTEM OPTION 389  
HYPERWRITE IS ENABLED
```

SETIOS ZHYPERLINK

As described in this section, the **SETIOS** command specifies the operation mode of zHyperLink in the system.

As in the IECIOSxx specification, four different values are available for the OPER parameter in the **SETIOS ZHYPERLINK** command, as shown in the following example:

```
SETIOS ZHYPERLINK,OPER={ALL|READ|WRITE|NONE}
```

The value that is defined on the zHyperLink statement specifies whether the zHyperLink facility is enabled on the following operations:

- ▶ ALL I/O operations
- ▶ Read I/O operations
- ▶ Write I/O operations
- ▶ Disabled for all I/O operations.

Whether zHyperLink is used for an I/O request depends on if this request meets the eligibility requirements. **SETIOS ZHYPERLINK,OPER=ALL** activates zHyperLink for READ and WRITE I/Os

SETIOS ZHYPERLINK,OPER=READ activates zHyperLink for READ I/Os only.

SETIOS ZHYPERLINK,OPER=WRITE activates zHyperLink for WRITE I/Os only.

SETIOS ZHYPERLINK,OPER=NONE disables zHyperLink for all I/Os.

The following example shows how to activate read I/Os for zHyperLink and what is expected as the command result:

```
SETIOS ZHYPERLINK,OPER=READ
```

Message IOS090I is displayed in response to the **SETIOS ZHYPERLINK** command:

```
IOS090I SETIOS. ZHYPERLINK UPDATE(S) COMPLETE
```

For more information about the **SETIOS ZHYPERLINK** command, see 4.1, “System commands” on page 80.

Remember: All system parameters that are changed dynamically are valid until the system is started. To keep your updates permanent, you must update your system parameters in the PARMLIB. For more information, see “Static activation” on page 65.

3.4 Monitoring

Monitoring your systems is vital to ensure system availability and meet required service level agreement (SLA)s. The following sections describe some of the available options to monitor zHyperLink, and identify possible errors and interventions that might be required.

In addition to the options available here, we encourage you to check for other monitoring options that might be useful in your site and are not described here. Some options can include creating tailored report jobs, user programs, and other reporting tools.

3.4.1 System automation

The use of system automation tools for managing started tasks and for monitoring and reporting on relevant messages on z/OS systems is widely adopted by customers around the world. These tools relieve operators to focus on activities that cannot be automated and require manual intervention. It permits immediate reaction to any serious alert. The operations team can even use system automation tools to assist them in performing specific tasks, or creating alerts that are based on defined system messages.

The infrastructure for PCIe I/O services is provided by the PCIe address space that is running under IOSAS. The PCIe address space is automatically started during IPL, even if no PCIe features are available on the CPC. No System Automation activities are required for the PCIe address space.

zHyperLink introduced and updated a set of messages that can be used by your system automation tool to monitor zHyperLink states. An example of an error message IOS637E that states that PCIe zHyperLink task terminated is shown in Figure 3-13.

```
IOS637E THE ZHYPERLINK MANAGER TASK HAS TERMINATED.  
ZHYPERLINK AVAILABILITY MAY BE AFFECTED.
```

Figure 3-13 Error message IOS637E example

Figure 3-14 shows that the IOS639E message is displayed if the zHyperLink PCIe function failed recovery.

```
IOS639E INTERVENTION REQUIRED FOR PFID pfid, reasontext
```

Figure 3-14 Error message IOS639E example

For more information about IOS system messages, see 4.3, “System messages” on page 93.

3.5 Activating zHyperLink for VSAM data sets

zHyperLink can be activated at the data set level, which provides options for granularity. In this section, we describe the two available options to activate zHyperLink for VSAM data sets.

3.5.1 zHyperLink for SMS data sets

Storage Class Granularity turns zHyperLink off and on for specific non-SMS data sets, and to override the zHyperLink specification for SMS-managed data sets.

With Storage Class Granularity, the system administrator can enable and disable synchronous I/O to the data sets that are members of a storage class, and enable or disable synchronous I/O for specific data sets by operator command. The ability to dynamically enable or disable zHyperLink also facilitates the testing of this facility.

Through Interactive Storage Management Facility (ISMF), you can define or alter a storage class to enable Storage Class Granularity in support of zHyperLink. An example of storage class definition is shown in Figure 3-15.

```
SCDS Name . . . . . : SMS.SCDS
Storage Class Name : SCZHPLNK

To DEFINE Storage Class, Specify:

Guaranteed Space . . . . . N          (Y or N)
Guaranteed Synchronous Write . . . N  (Y or N)
Multi-Tiered SG . . . . .             (Y, N, or blank)
Parallel Access Volume Capability N    (R, P, S, or N)
CF Cache Set Name . . . . .           (up to 8 chars or blank)
CF Direct Weight . . . . .            (1 to 11 or blank)
CF Sequential Weight . . . . .        (1 to 11 or blank)
CF Lock Set Name . . . . .            (up to 8 chars or blank)
Disconnect Sphere at CLOSE . . . . N  (Y or N)
zHyperLink Eligible for read. . . . Y  (Y or N)
zHyperLink Eligible for write . . . Y  (Y or N)
```

Figure 3-15 Storage Class panel with zHyperLink options and enabling write

Two other options are added for zHyperLink. The specified values take effect when the SMS configuration is activated.

A storage class display on the ISMF panel is shown in Figure 3-16.

```
CDS Name . . . . . : SYS1.STPPLEX.SCD5
Storage Class Name : SCZHPLNK

Guaranteed Space . . . . . : NO
Guaranteed Synchronous Write . . : NO
Multi-Tiered SGs . . . . . :
Parallel Access Volume Capability : NOPREF
Cache Set Name . . . . . :
CF Direct Weight . . . . . :
CF Sequential Weight . . . . . :
Lock Set Name . . . . . :
Disconnect Sphere at CLOSE . . . : NO
zHyperLink Eligible for read. . . . YES
zHyperLink Eligible for write . . . NO
```

Figure 3-16 ISMF panel with zHyperLink option

3.5.2 zHyperLink for non-SMS data sets

The **VARY SMS** command is the only option that is available to enable zHyperLink for a non-SMS data set. This command provides an option to switch zHyperLink I/O status on an individual data set basis.

The **VARY** command overrides the storage class definition. After the command is issued, all current and subsequent opens use the specified value. It is the only option for non-SMS data sets to use zHyperLink:

- ▶ **VARY SMS,DSNAME(dsname),ZHLREAD=YES**
Enables synchronous I/O reads for the data set. When the vary command overwrites storage class definition, the reads are successful even if the storage class is ineligible for reads.
- ▶ **VARY SMS,DSNAME(dsname),ZHLREAD=NO**
Overrides storage class eligibility for reads. Issues reads to the data set by using Synchronous I/O. Reads should use zHPF because the data set is not enabled for Synchronous I/O reads.

After varying SMS and enabling zHyperLink to the required datasets, check the status of zHyperLink in your datasets by using the following display SMS command:

```
D SMS,DSNAME(datasetname|ALL),ZHL
```

For more information about this command in 4.1, “System commands” on page 80.

The **D SMS** command was updated to include the option to display zHyperLink attributes to a specific data set. You can use this command to display if SMS or non-SMS-managed data sets are zHyperLink eligible.

3.6 Activating zHyperLink for Db2

Before your Db2 applications can benefit from synchronous I/O operations, you must enable zHyperLink to your Db2 subsystem. You can activate zHyperLink from the installation and migration panel that is available on Db2. For more information about how to start the installation and migration panels, see *Db2 12 for z/OS Installation and migration*, [GC27-8851](#).

From the DSNTIPA3 panel, you can see a new option to control zHyperLink utilization for Db2. A sample DSNTIPA3 panel is shown in Figure 3-17.

```

DSNTIPA3          INSTALL DB2 - DATA PARAMETERS PANEL
====> _

Check parameters and reenter to change:
 1 PERMANENT UNIT NAME  ==> 3390      Device type for MVS catalog
                                   and partitioned data sets
 2 TEMPORARY UNIT NAME  ==> SYSDA     Device type for
                                   temporary data sets
 3 DB2 zHyperLinks SCOPE ==> ENABLE   Scope of zHyperLinks I/O connections:
                                   (DISABLE, ENABLE, DATABASE)

                                   ----- SMS -----
                                   VOL/SER  DATA CLASS  MGMT CLASS  STOR CLASS
                                   -----
 4 CLIST ALLOCATION      ==>          ==>          ==>          ==>
 5 NON-VSAM DATA       ==>          ==>          ==>          ==>
 6 VSAM CATALOG, DEFAULT, ==>      ==>          ==>          ==>
  AND WORK FILE DATABASE
 7 LOG COPY 1, BSDS 2   ==>          ==>          ==>          ==>
 8 LOG COPY 2, BSDS 1   ==>          ==>          ==>          ==>

```

Figure 3-17 Sample Db2 installation panel

Db2 zHyperLinks SCOPE includes the following options:

- ▶ **DISABLE**
Db2 does not use zHyperLink for any I/O requests.
- ▶ **ENABLE**
All eligible I/O requests that are performed by Db2 use the zHyperLink protocol.
- ▶ **DATABASE**
Db2 request only the zHyperLink protocol for database synchronous read I/Os.

The entries on the DSNTIPA3 panel define storage management preferences for the storage groups and data sets that are created during installation and migration.

After you apply and save your changes, the changes are dynamically loaded on storage and no Db2 recycle is required.

Note: In addition to Db2 configuration, you must enable the VSAM data sets to use zHyperLink before your Db2 systems can benefit from synchronous I/O. For more information about how to set up zHyperLink for VSAM data sets, see 3.5, “Activating zHyperLink for VSAM data sets” on page 70.

3.7 Measuring after deployment

After you implement zHyperLink in your system, it is highly recommended that you create the means to measure the benefits from the synchronous I/O, and plan for future growth.

Input for Resource Monitoring Facility (RMF) and zBNA is based on SMF records. Therefore, you can run reports that are based on records from before and after zHyperLink implementation. This feature enables you to compare results and assess the benefits of zHyperLink. You also can disable and enable zHyperLinks at data set, storage class, and system levels, which provides you with options for granularity.

In general, performance comparisons should always be performed on similar workloads.

3.7.1 Measuring by using Resource Monitoring Facility

After you install and enable zHyperLink in your systems, you can use RMF to measure zHyperLink utilization and benefits.

RMF panels

RMF was updated to include zHyperLink and synchronous I/O information. The information that is provided in RMF panels enables you to better view zHyperLink usage and efficiency. You can use the RMF panels or Postprocessor to gather zHyperLink and DASD-related information, including response time and cache hit ratio.

The RMF Monitor III PCIe Activity panel was updated to display information about zHyperLink PCIe cards, including information about the number of reads, usage, and success rate.

From the RMF main menu, you can select option 3, **monitor III**, and then, option 3, **resource**, and finally, option 14, **PCIe**, to display PCIe-related information, including zHyperLink functions.

The updated PCIe Activity panel is shown in Figure 3-18.

Command ==>		RMF V2R5		PCIe Activity		Line 1 of 4						
						Scroll ==> CSR						
Samples: 100		System: SC75		Date: 10/18/21		Time: 13.35.00		Range: 100		Se		
-----		Function		-----		Alloc - PCI Operations		Rate - -Xfer Rate				
ID	CHID	Type	Jobname	ASID	Status	Time%	Load	Store	Block	Refr	Read	Writ
0103	013C	zHypL	IOSAS	0019	Alloc	100					0	0.00
0203	013C	zHypL	IOSAS	0019	Alloc	100					0	
0303	021C	zHypL	IOSAS	0019	Alloc	100					0	0.00
0403	021C	zHypL	IOSAS	0019	Alloc	100					0	

Figure 3-18 RMF PCIe Activity panel

- Link (CPC)

The percentage of time spent on synchronous I/O processing on the synchronous I/O link (physical connection). If the time busy percentage is high, plan to add zHyperLink adapters.
- ▶ Request rate
 - This function

The number of total synchronous I/O requests per second for this function. This number includes requests that were rejected because of timeout conditions, cache failures, and link busy conditions.
 - Link (CPC)

The number of total synchronous I/O requests per second for the synchronous I/O link on which this function is defined. This number includes requests that were rejected because of timeout conditions, cache failures, and link busy conditions.
- ▶ Success percentage (%)
 - This Function

The percentage of synchronous I/O requests that completed successfully for this function.
 - Link (CPC)

The percentage of requests that completed successfully for the synchronous I/O link on which this function is defined.
- ▶ Read Rate:
 - This Function

The number of megabytes per second that were read from the storage controller by this synchronous I/O function.
 - Link (CPC)

The number of megabytes per second that were read from the storage controller on the synchronous I/O link on which this function is defined.
- ▶ Read Ratio:
 - This Function

The number of megabytes read per request that was processed by this synchronous I/O function.
 - Link (CPC)

The number of megabytes read per request that was processed on the synchronous I/O link on which this function is defined.
- ▶ Write Rate:
 - This Function

The number of megabytes per second that were written to the storage controller by this synchronous I/O function.
 - Link (CPC)

The number of megabytes per second that were written to the storage controller on the synchronous I/O link on which this function is defined.

► Write Ratio:

– This Function

The number of megabytes written per request processed by this synchronous I/O function.

– Link (CPC)

The number of megabytes written per request processed on the synchronous I/O link on which this function is defined.

RMF batch reports

You also can create the same report by running a batch job. A sample RMF job to extract PCIe information is shown in Example 3-7.

Example 3-7 Sample report PCIe JCL

```
//RMFPP EXEC PGM=ERBRMFPP,REGION=OM
//MFPINPUT DD DISP=(OLD,DELETE),DSN=*.RMFSORT.SORTOUT
//MFPMSGDS DD SYSOUT=*
//XPRPTS DD PATH='/u/mysuer/rmfrpt/report1.xml',
// PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
// PATHMODE=(SIRUSR,SIWUSR,SIRGRP),FILEDATA=TEXT
//XPXSRPTS DD PATH='/u/myuser/rmfrpt/report2.xml',
// PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
// PATHMODE=(SIRUSR,SIWUSR,SIRGRP),FILEDATA=TEXT
//*****
//*
//* RMF POSTPROCESSING OPTIONS GENERATED FROM
//* POSTPROCESSOR OPTIONS PANEL INPUT ONLY.
//*
//*****
//SYSIN DD *
SYSOUT(0)
REPORTS(PCIE)
```

The batch report generates XML files with PCIe-related information. The XML reports include the same information that is available on the PCIe Activity panel, in addition to the Synchronous I/O Response Time Distribution field.

The Synchronous I/O Response Time Distribution field shows the percentage of synchronous I/O requests that was completed within a response time by PFID. The values range from less than 20 microseconds to less than 100 microseconds. You can use this information to compare PFID and adapter performance, and identify possible link degradation states.

Another option to measure your performance after zHyperLink enablement is based on the report DASD device that is available on RMF Postprocessor. You can create a job to report your DASD environment and analyze synchronous and asynchronous response times.

A sample RMF job to report on DASD devices is shown in Example 3-8.

Example 3-8 Sample report DASD JCL

```
//RMFPP EXEC PGM=ERBRMFPP,REGION=OM
//MFPINPUT DD DISP=SHR,DSN=*.RMFSORT.SORTOUT
//MFPMSGDS DD SYSOUT=*
//*****
//*
//* RMF POSTPROCESSING OPTIONS GENERATED FROM
//* POSTPROCESSOR OPTIONS PANEL INPUT ONLY.
//*
//*****
//SYSIN DD *
        SYSOUT(0)
        REPORTS(DEVICE(DASD))
```

The output from the report device job is shown in Figure 3-20.

SYNCHRONOUS I/O DEVICE ACTIVITY															PAGE 1	
z/OS V2R2		SYSTEM ID MVS6				START 02/12/2018-17.50.00		INTERVAL 000.01.00								
		RPT VERSION V2R2 RMF				END 02/12/2018-17.51.00		CYCLE 0.200 SECONDS								
TOTAL SAMPLES = 300		IODF = 71		CR-DATE: 02/02/2018		CR-TIME: 14.10.07		ACT: ACTIVATE								
		- DEVICE ACTIVITY RATE -				-- AVG RESP TIME --		AVG SYNCH I/O		% % % %		--REJECTS--				
STORAGE GROUP	DEV NUM	DEVICE TYPE	VOLUME SERIAL	LCU	SYNCH I/O	ASYNCH I/O	SYNCH I/O	ASYNCH I/O	TRANSFER RATE	REQ	LINK	CACHE	READ	WRITE		
F000	33909	SRF000	0071 16083.7	0.000	4062.07	0.026	0.000	0.095	65.88	0.000	79.84	19.95	0.00	0.22	0.00	
F001	33909	SRF001	0071 15675.8	0.000	4140.55	0.026	0.000	0.095	64.21	0.000	79.11	20.68	0.00	0.21	0.00	
		LCU	0071 31759.5	0.000	8202.61	0.026	0.000	0.095	130.1	0.000	79.47	20.31	0.00	0.21	0.00	
F100	33909	SRF100	0072 13768.5	0.000	3322.40	0.026	0.000	0.095	56.40	0.000	80.56	19.25	0.00	0.19	0.00	
F101	33909	SRF101	0072 16067.9	0.000	3963.07	0.026	0.000	0.095	65.81	0.000	80.22	19.60	0.00	0.18	0.00	
		LCU	0072 29836.4	0.000	7285.46	0.026	0.000	0.095	122.2	0.000	80.38	19.44	0.00	0.19	0.00	
F200	33909	SRF200	0073 13884.8	0.000	4516.37	0.026	0.000	0.095	56.87	0.000	75.46	24.34	0.00	0.20	0.00	
F201	33909	SRF201	0073 15163.8	0.000	4171.43	0.026	0.000	0.095	62.11	0.000	78.43	21.37	0.00	0.20	0.00	
		LCU	0073 29048.6	0.000	8687.81	0.026	0.000	0.095	119.0	0.000	76.98	22.82	0.00	0.20	0.00	
F300	33909	SRF300	0074 13729.1	0.000	3320.81	0.026	0.000	0.095	56.23	0.000	80.52	19.29	0.00	0.19	0.00	
F301	33909	SRF301	0074 15724.3	0.000	4058.68	0.026	0.000	0.095	64.41	0.000	79.49	20.33	0.00	0.18	0.00	
		LCU	0074 29453.3	0.000	7379.49	0.026	0.000	0.095	120.6	0.000	79.97	19.85	0.00	0.19	0.00	

Figure 3-20 Synchronous I/O Device Activity output

The Synchronous I/O Device Activity report includes more information about devices for synchronous and asynchronous I/Os. Use the available information to check whether the number of synchronous requests, cache miss, and synchronous response time meet application expectations.

For more information about the fields in the Synchronous I/O Device Activity report, see *IBM z/OS RMF Report Analysis*, [SC34-2665](#).

3.7.2 Measuring using zBNA

The zBNA zHyperLink I/O Rate report documents I/Os per second (IOPS) for workloads that use zHyperlink and eligible zHyperLink workloads. The report also documents the highest potential I/O rate that is estimated to be achievable if all zHyperLink workloads were enabled.

You can use the report as input for evaluating whether zHyperLink was enabled for an optimal amount of data.

Note: The zBNA tool is designed to estimate the potential effect moving to new technologies can have for batch elapse time. It is not designed for measuring overall system performance. When a zBNA report is created multiple times, it cannot be guaranteed that all variables are the same for each report.

For more information about the zBNA tool, see 2.1, “Identifying zHyperLink eligible workloads” on page 16.



Operations

This chapter provides relevant information to daily operations after zHyperLink is implemented. This information can be used to create automation procedures, monitor activities, and troubleshoot errors.

This chapter includes the following topics:

- ▶ 4.1, “System commands” on page 80
- ▶ 4.2, “System automation” on page 91
- ▶ 4.3, “System messages” on page 93
- ▶ 4.4, “Hardware Management Console” on page 99
- ▶ 4.5, “Troubleshooting” on page 104

4.1 System commands

This section documents system commands that are used for displaying and managing resources that are related to zHyperLink.

4.1.1 Display commands

In this section, we introduce some commands that can be used to display zHyperLink Peripheral Component Interconnect Express (PCIe) related information.

D PCIE

The use of the **D PCIE** command shows the information that is registered for the device type with PCIe. The example that is shown in Figure 4-1 shows two zHyperLink PCIe I/O cards with two ports each.

```
D PCIE
IQP022I 06.58.02 DISPLAY PCIE 204
PCIE      0012 ACTIVE
PFID      DEVICE TYPE NAME          STATUS ASID  JOBNAME  CHID VFN  PN
00000102  8GB zHyperLink                   ALLC   0019  IOSAS    013C 0003 1
00000202  8GB zHyperLink                   ALCC   0019  IOSAS    013C 0003 2
00000302  8GB zHyperLink                   ALLC   0019  IOSAS    021C 0003 1
00000402  8GB zHyperLink                   ALCC   0019  IOSAS    021C 0003 2
```

Figure 4-1 Sample D PCIE command output

The second line displays the PCIe address space name followed by the task ID and status. In this example, the PCIe address space is PCIE with the ID of 0012.

The scope of the D PCIE command is all PCIe functions. For zHyperLink, the JOBNAME and corresponding ASID always is IOSAS. The CHID is the physical channel identifier and PN is the port number.

If the zHyperLink STATUS is different from ALLC (allocated), issue the **D PCIE,PFID=pfid** command for more information.

D PCIE,PFID=pfid

The use of the **D PCIE,PFID=pfid** command shows information about software and port states, as shown in Figure 4-2.

```
D PCIE,PFID=0102
IQP024I 07.06.44 DISPLAY PCIE 219
PCIE      0012 ACTIVE
PFID      DEVICE TYPE NAME          STATUS ASID  JOBNAME  CHID VFN  PN
00000102  8GB zHyperLink                   ALLC   0019  IOSAS    013C 0003 1
CLIENT ASIDS: NONE
CU WNN: 5005076306FFD699  CU Link Id: 0080
S/W State: Allocated
Port State: Operational
CU Node Descriptor: 002107.981.IBM.75.000000FCC41
```

Figure 4-2 Sample D PCIE,PFID=pfid command output

The output includes the following information:

- ▶ Worldwide node name of the storage system (CU) (if provided)
Displayed as `Not Available` if not provided (for example, synchronous I/O link was never operational). Information is residual for links that are not operational.
- ▶ CU link ID
The physical port on the disk subsystem to where the zHyperLink is connected. Displayed as `Not Available` if not provided. Information is residual for links that are not operational.
- ▶ Software State
State from the point of view of the sync I/O link manager (`Allocated`, `Error`, `Allocate Error`, `Deallocate Error`, `Recovery pending`, or `Intervention Required`).
- ▶ Node descriptor
If the WWNN is not provided, it is displayed as `Not available`. For example, this state can occur if the synchronous I/O link was never operational. Displaying the information regardless of whether the WWNN is provided enables the client to see the state of the link.
- ▶ Port state
This state from the point of view of the firmware (`None`, `Operational`, `Not Oper`, or `Undefined`). Port states are listed in Table 4-1.

Table 4-1 Port states

Summary state	Detailed State	Explanation
None	None	No state information is available about the port.
Operational	None	The port is operational and running at full capacity. A port state qualifier of 01 means fully operational.
Operational	Link speed degraded	This port is operational, but is running in a degraded state because the link started at a speed less than the maximum capable speed (for example, Gen-2 instead of Gen-3).
Operational	Link width degraded	The port is operational, but is running in a degraded state because only a subset of the fibers (lanes) on the link is functional (for example, four out of eight lanes).
Not Operational	None	The port is not operational. No other information is provided. For example, the port might be in the process of starting.
Not Operational	Link down	The port is not operational because the link is down.
Not Operational	Control unit not responding	The link is up, but the control unit is not responding to the start sequence.
Not Operational	Fenced by processor	The port was fenced by the processor because the number of detected errors exceeded the internal threshold.
Not Operational	Fenced by control unit	The port was fenced by the control unit because the number of detected errors exceeded the threshold.

Summary state	Detailed State	Explanation
Not Operational	Processor side in service mode	The port is not operational because the local (processor) side of the link is in service mode (for example, CE testing or replacing the optics).
Not Operational	Control unit side in service mode	The port is not operational because the remote (control unit) side of the link is in service mode.
Not Operational	Insufficient control unit resources	The port is not operational because of insufficient resources to support the link. For example, there are not enough CPUs on the control unit to support the link.
Undefined	None	The state of the port is undefined. Contact the IBM Support Center if this state is displayed.

D IOS,ZHYPERLINK | ZHL

The display IOS command (**D IOS**) with the **ZHYPERLINK** parameter shows if zHyperLink is ENABLED or DISABLED. When zHyperLink is ENABLED, it describes what I/O operation is available to use zHyperLink (READ|WRITE). In specific cases in which hardware requirements were not met, it might display NOT SUPPORTED BY THE PROCESSOR instead.

D IOS,ZHYPERLINK | ZHL,LINK=

The display IOS command (**D IOS,ZHYPERLINK**) with the **LINK=ALL** parameter is used to display summary information for all links.

The PCHID, port, subsystem port, Port state, manufacturer, and serial number of subsystem plus the number of available or unavailable online ports are displayed. Offline PFIDs are not included in the display.

The display IOS command (**D IOS,ZHYPERLINK**) with the **LINK=pchid.port** parameter is used to display summary information for a specific link. The display is identical to the display that is provided by the **LINK=ALL** parameter, but limited to a single PCHID and port.

Add the **DETAIL** parameter to include WWNN, PFID, and S/W state plus the number of successful and link busy read and write operations in the display, as shown in Figure 4-3.

```
D IOS,ZHL,LINK=013C.1,DETAIL
IOS644I 07.08.19 LINK DETAILS 223
      CU Port    ---- CU Info --- -- PFIDs -
PCHID PN Link State  Mfg.Ser or WWNN Avail Unav
013C 1 0080 Oper    IBM.0000000FCC41    1    0
  WWNN: 5005076306FFD699
  PFID: 00000102  S/W State: Allocated
  Success:  Read - 507 Write - 11,381
  Diagnostic Info: 00100000
```

Figure 4-3 Sample D IOS,ZHL,LINK=ALL,DETAIL command output

Note: The number of successful Read and Write operations is not displayed if both numbers are zero. Likewise, for the number of link busy Read and Write operations.

D IOS,ZHYPERLINKIZHL,PFID=

The display IOS (**D IOS,ZHYPERLINK**) command with the **PFID=ALL** parameter is used to display summary information for all PFIDs.

The PFID, PCHID, port, subsystem port, S/W and port state, manufacturer, and serial number of the subsystem are displayed. Offline PFIDs are not included in the display.

The display IOS command (**D IOS,ZHYPERLINK**) with the **PFID=pfid** parameter is used to display summary information for a specific link. The display is identical to the display that is provided by the **PFID=ALL** parameter, but limited to a single PFID.

Add the **DETAIL** parameter to include WWNN, PFID, and S/W state plus the number of successful and link busy read and write operations in the display.

D IOS,HYPERWRITE

The display IOS command (**D IOS,HYPERWRITE**) displays the status of the **HYPERSHARE** option that is defined in parmlib member **IECIO5xx** or set by way of the **SETIOS HYPERSHARE** system command.

Note: HyperWrite must be enabled for zHyperLink writes.

D M=DEV(devno)

The Display Matrix command (**D M**) with **DEV** parameter is commonly used to display the number of online channel paths to devices (including special devices) or a single channel path to a single device. It now provides the following information about zHyperLink:

► Interface IDs

This ID is useful for determining whether CHPIDs share a host adapter (the first three digits represent the adapter).

The interface ID line is displayed if the device was online at some point or self-description processing was performed for some other reason (for example, offline device discovery or IOSODS); that is, an SSCB exists for the device.

If a **D M=DEV(devno)** command is issued and the particular path was never online to any device in the LCU but other paths were online, the interface ID is displayed as "...".

If a **D M=DEV(devno,(chpid))** command was issued in this scenario, the interface ID line is not displayed.

► WWNN

The WWNN is displayed if the WWNN was read when a path first came online to the LCU.

The current device must be online at some point or self-description processing must be performed (that is, an SSCB exists for the device).

► zHyperLinks available

This line is not displayed if no zHyperLinks are available, regardless of whether the device is enabled to use zHyperLink.

A link is considered available (usable) if the PFID is online, the port is operational, and z/OS allocated the PFID. Links where the PFID is offline, the port is not operational, or z/OS failed to allocate the link are not included.

► Functions enabled

If the device is enabled for zHyperLink reads and writes, **ZHYPERLINK** is displayed. If the device is enabled for zHyperLink reads only, **ZHYPERLINK(R)** is displayed. If the device is enabled for zHyperLink writes only, **ZHYPERLINK(W)** is displayed.

An example of how the **D M=DEV** command displays zHyperLink in an enabled state and the new information is shown in Figure 4-4.

```

D M=DEV(9000)
IEE174I 18.09.26 DISPLAY M 830
DEVICE 09000 STATUS=ONLINE
CHP          40  42  41  43
ENTRY LINK ADDRESS 012C 012D 022C 022D
DEST LINK ADDRESS 010C 010B 020C 020B
PATH ONLINE      Y  Y  Y  Y
CHP PHYSICALLY ONLINE Y  Y  Y  Y
PATH OPERATIONAL Y  Y  Y  Y
MANAGED          N  N  N  N
CU NUMBER      0690 0690 0690 0690
INTERFACE ID   0030 0230 0100 0300
MAXIMUM MANAGED CHPID(S) ALLOWED: 0
DESTINATION CU LOGICAL ADDRESS = 90
SCP CU ND        = 002107.996.IBM.75.0000000LBN71.0030
SCP TOKEN NED    = 002107.900.IBM.75.0000000LBN71.9000
SCP DEVICE NED   = 002107.900.IBM.75.0000000LBN71.9000
WNNN          = 500507630AFFD049
HYPERPAV ALIASES CONFIGURED = 16
ZHYPERLINKS AVAILABLE = 2
FUNCTIONS ENABLED = MIDAW, ZHPF, XPAV, ZHYPERLINK

```

Figure 4-4 Output from **D M=DEV** command

D M=DEV(devno),ZHYPERLINK

The **D M=DEV(devno),ZHYPERLINK** display command supports the option to display zHyperLink capabilities. It also shows the reason why a device is not enabled for zHyperLink. A new message was introduced to the output of the command: IEE587I.

If the device that is displayed is enabled for zHyperLink, it provides information about the type of zHyperLink (READ|WRITE) and the number of zHyperLinks that are available for accessing the device.

Whether zHyperLink is entirely disabled for READ and WRITE, or partially disabled for READ or WRITE, it displays why it is disabled with the following statements:

- ▶ Processor reasons
 - PROCESSOR DOES NOT SUPPORT ZHYPERLINK
- ▶ z/OS reasons (IECIOSxx, SETIOS):
 - HYPERLINK IS DISABLED FOR THE SYSTEM
 - HYPERLINK {READS | WRITES} ARE DISABLED FOR THE SYSTEM
 - HYPERWRITE IS DISABLED FOR THE SYSTEM (Writes only)
- ▶ Control unit reasons:
 - CONTROL UNIT DOES NOT SUPPORT ZHYPERLINK
 - CONTROL UNIT DOES NOT SUPPORT ZHYPERLINK {READS | WRITES}
 - CONTROL UNIT WNNN COULD NOT BE RETRIEVED
 - CONTROL UNIT DOES NOT SUPPORT HYPERWRITE (Writes only)
- ▶ zHyperLinks reasons
 - THERE ARE NO ZHYPERLINKS AVAILABLE

- ▶ Secondary device reasons (when the primary is displayed, writes only):
 - THERE ARE NO ZHYPERLINKS AVAILABLE FOR SECONDARY devno
 - CONTROL UNIT WWNN COULD NOT BE RETRIEVED FOR SECONDARY devno
 - CONTROL UNIT DOES NOT SUPPORT ZHYPERLINK WRITES FOR SECONDARY devno
 - CONTROL UNIT DOES NOT SUPPORT HYPERWRITE FOR SECONDARY devno

Note: IEE587I PROCESSOR DOES NOT SUPPORT ZHYPERLINK is received because the processor does not support the zHyperLink function or **MACHMIG TX** was specified in LOADxx to disable the transactional execution facility. zHyperLink is supported on IBM z14 and later processors. The transactional execution facility must be enabled to use zHyperLink.

D M=CU(cuno)

Issue the **D M=CU** command to see more information about zHyperLink from Control Unit.

As with the **D M=DEV** command, the Control Unit display includes fields that were recently added to its command response. For more information about Interface IDs, WWNN, and Functions enabled fields, see “D M=DEV(devno)” on page 83.

An example of a Control Unit display with the fields added to command response is shown in Figure 4-5.

```

D M=CU(0690)
RESPONSE=SC75
IEE174I 18.12.52 DISPLAY M 838
CONTROL UNIT 0690
CHP                40  42  41  43
ENTRY LINK ADDRESS 012C 012D 022C 022D
DEST LINK ADDRESS  010C 010B 020C 020B
CHP PHYSICALLY ONLINE Y  Y  Y  Y
PATH VALIDATED     Y  Y  Y  Y
MANAGED            N  N  N  N
ZHPF - CHPID      Y  Y  Y  Y
ZHPF - CU INTERFACE Y  Y  Y  Y
INTERFACE ID      0030 0230 0100 0300
MAXIMUM MANAGED CHPID(S) ALLOWED = 0
DESTINATION CU LOGICAL ADDRESS = 90
CU ND              = 002107.996.IBM.75.0000000LBN71.0030
CU NED             = 002107.996.IBM.75.0000000LBN71.9000
TOKEN NED          = 002107.900.IBM.75.0000000LBN71.9000
WWNN              = 500507630AFFD049
FUNCTIONS ENABLED = ZHPF, ZHYPERLINK, XPAV
XPAV CU PEERS     = 0690
DEFINED DEVICES
  09000-090EF
DEFINED PAV ALIASES
  090F0-090FF
USABLE HYPERPAV ALIASES = 16
ZHYPERLINKS
  PFID    PCHID Port LinkId S/W St  Port St
00000203 013C  02   0181  Allloc  Oper
00000403 021C  02   0081  Allloc  Oper

```

Figure 4-5 Sample D M=CU(cuno) output

Only online PFIDs are shown, regardless of whether the link is usable. Offline (standby) PFIDs are not included. Processor-side information includes PFID, PCHID, and port data. Control unit side information includes CU link ID (which is similar to interface ID).

The following software and port states are available:

- ▶ Software states:
 - Alloc: The PFID is allocated and usable.
 - Error: The PFID is in permanent error or port is not operational.
 - RcvPend: Recovery is pending for the PFID.
 - AllocEr: An error occurred while attempting to allocate the PFID.
 - DeallocEr: An error occurred while attempting to deallocate the PFID.
 - IntvReq: Intervention is required to make the PFID usable.
 - NotAvail: The PFID is configured online to the logical partition, but was not allocated.
- ▶ Port states:
 - Only high-level states are shown. Issue **D PCIE,PFID=pfid** for more information about non-operational conditions.
 - None: The state of the port is unknown.
 - Oper: The port is operational.
 - NotOper: The port is not operational.
 - Degraded: The port is operational, but running in degraded state (port state qualifier is 02 or 03).

DS QD,devno

The **DS QD,devno** command is used to identify the disk subsystem SCU-serial for the specified device number. The SCU-serial is needed for the **DS QD,MACH=scu-serial,ZHL** command.

DS QD,MACH=scu-serial,ZHL

The **DS QD,MACH=scu-serial,ZHL** command displays zHyperLink write tokens that are allocated to data sets on a disk subsystem. The **MACH=** parameter is the 10-byte SCU serial number of the disk subsystem that was obtained by using the **DS QD,devno** command.

One token is representing a zHyperLink write session with the DS8000. The maximum number of write sessions at disk subsystem level is 64.

Example 4-1 shows a sample output from the **DS QD,scu-serial,ZHL** command.

Example 4-1 Sample DS QD,scu-serial,ZHL command

```

DS QD,MACH=0175-FCC41,ZHL
IEE459I 04.10.26 DEVSERV QDASD 419
  UNIT VOLSER SCUTYPE DEVTYPE      CYL  SSID SCU-SERIAL DEV-SERIAL EFC
09200 CONCD1 2107981 2107900      1113 0230 0175-FCC41 0175-FCC41 *OK
  ZHYPERLINK WRITE TOKENS ASSIGNED
TOKEN DATE    TIME  UNIT  DATA SET NAME
00AD 07/07/21 15:10 09C16 DB2BL.D2B1.LOGCOPY2.DS02.DATA
01AE 07/07/21 15:10 09B96 DB2BL.D2B1.LOGCOPY1.DS02.DATA
02AF 07/07/21 15:10 09B96 DB2BL.D2B2.LOGCOPY1.DS02.DATA
03B0 07/07/21 15:10 09639 DB2BL.D2B2.LOGCOPY2.DS02.DATA
****          4 WRITE TOKEN(S) MET THE SELECTION CRITERIA
****          1 DEVICE(S) MET THE SELECTION CRITERIA
****          0 DEVICE(S) FAILED EXTENDED FUNCTION CHECKING

```

Note: The **DS QD,MACH=scu-serial,ZHL** command display information at the disk subsystem level. The display is not limited to the system on which it is issued.

DB2 Display log

The DB2 Display LOG (**-Display log**) command can be used to determine whether the last DB2 log write used zHyperLink. The command does not provide any information about how often zHyperLink is used. For more information about zHyperLink statistics, see “D SMS,DSNAME(dsname),STATS(ZHLWRITE)[,RESET]” on page 88.

A sample DB2 Display Log command is provided in Example 4-2. Because the command example is issued on a z/OS console, the *command prefix* for the DB2 subsystem is specified.

Example 4-2 Sample DB2 Display log command

```
-D2B1 DISPLAY LOG
DSNJ370I  -D2B1 DSNJC00A LOG DISPLAY 485
CURRENT COPY1 LOG = DB2BL.D2B1.LOGCOPY1.DS02 IS 33% FULL
CURRENT COPY2 LOG = DB2BL.D2B1.LOGCOPY2.DS02 IS 33% FULL
          H/W RBA = 0000000000036789C1E0
          H/O RBA = 00000000000363EB7FFF
          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  -D2B1 DB2 RESTARTED 17:21:56 JUL 11, 2021 486
          RESTART RBA 000000000003673C5000
          CHECKPOINT FREQUENCY 5 MINUTES
          LAST SYSTEM CHECKPOINT TAKEN 08:42:01 JUL 12, 2021
DSN9022I  -D2B1 DSNJC001 '-DISPLAY LOG' NORMAL COMPLETION
```

D SMS,DSNAME(datasetname|ALL),ZHL

The Display SMS (**D SMS**) command was updated to include the option to display zHyperLink attributes to a specific data set. You can use this command to display whether SMS or non-SMS-managed data sets are zHyperLink eligible. Example 4-3 shows a sample output from the **D SMS** command.

Example 4-3 Sample D SMS,DSNAME command

```
D SMS,DSNAME(APPL01.APPDATA.VSAM.DATA),ZHL
IGW285I 475
D SMS,DSNAME Start of Report
zHyperLink----- Data Set Name-----
OVERRIDE STORCLAS
Rd Wr  Rd Wr
Yes Def - -  APPL01.APPDATA.VSAM.DATA
D SMS,DSNAME End of Report
```

The command output includes the following fields:

- ▶ **Override Rd:** The current setting for zHyperLink Read. Yes|No|Default
- ▶ **Override Wr:** The current setting for zHyperLink Write. Yes|No|Default
- ▶ **Storclas Rd:** The current storage class specification for zHyperLink Read. Yes|No|-

- ▶ Storclas Wr: The current storage class specification for zHyperLink Write. Yes |No|-
- ▶ Name of the data set

For **sr** and **wr**, “-” indicates that no storclas is associated with the data set; blanks indicate that the storclas was not determined.

D SMS,DSNAME(dsname),STATS(ZHLWRITE)[,RESET]

The Display SMS (**D SMS**) command with the **STATS(ZHLWRITE)** parameter is used for displaying zHyperLink write statistics for the data set that is specified in the **DSNAME** parameter.

Adding the **RESET** parameter clears statistics that follow the display. The timestamp of the last **RESET** is displayed at the top of the output. If a **RESET** for the data set was never done, the time stamp from when the data set was opened is used.

The display is divided in two sections: the first section includes statistics for the data set; the second section includes statistics for the device.

Example 4-4 shows a sample output from the **D SMS,DSNAME(dsname),STATS(ZHLWRITE)** command.

Example 4-4 Sample D SMS,DSNAME(dsname),STATS(ZHLWRITE) command output

```

D SMS,DSNAME(DB2BL.D2B1.LOGCOPY1.DS02.DATA),STATS(ZHLWRITE)
IGW289I 523
D SMS,DSNAME,STATS(ZHLWRITE) Start of Report
DATA SET DB2BL.D2B1.LOGCOPY1.DS02.DATA
STATISTICS Since 07/07/2021 15:10:24.086875
SUMMARY
      TOTAL      %SYNC -----%ASYNC-----
WRITE REQUESTS WRITES  SKIP LNKBSY  ^EST  MISC DISABL
      15729  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  ^EST  MISC  MISS  DELAY
0243 09B96 15794 99.99   0.00   0.00  <0.01   0.00  0.00  0.00
D SMS,DSNAME,STATS(ZHLWRITE) End of Report

```

4.1.2 Modify commands

Modify commands allow you to alter system parameters to enable or disable specific functions or alter system values that are defined at IPL time. This section introduces some modify commands to control zHyperLink behavior in your systems.

VARY SMS,DSNAME(datasetname)

The **VARY SMS** command now includes the **DSNAME** option to enable zHyperLink eligibility changes to specific data sets. SMS and non-SMS data sets can be used with vary SMS command.

The following command options are available:

▶ **DSNAME**(datasetname)

The data set name. For VSAM, this name must be a component name and not a sphere name. Only VSAM data sets are supported for zHyperLink. A message IGW287I is issued if the data set is not cataloged or if the data set is not a VSAM-type component.

▶ **ZHLREAD=**YES|NO|DEFAULT

Specifies whether the data set can use zHyperLink for read requests:

– YES

This data set can use zHyperLink for read requests. It overrides the Storage Class setting for zHyperLink read eligibility.

– NO

This data set cannot use zHyperLink for read requests. It overrides the Storage Class setting for zHyperLink read eligibility.

– DEFAULT

This data set uses the Storage Class setting for zHyperLink read eligibility. zHyperLink read eligibility is governed by the Storage Class. If no storage class is associated with the data set, it cannot use zHyperLink for reads. If the data set is non-SMS, zHyperLink is not used to read this data set.

▶ **ZHLWRITE=**YES|NO|DEFAULT

Specifies whether the data set can use zHyperLink for write requests:

– YES

This data set can use zHyperLink for write requests. It overrides the Storage Class setting for zHyperLink write eligibility.

– NO

This data set cannot use zHyperLink for write requests. It overrides the Storage Class setting for zHyperLink write eligibility.

– DEFAULT

This data set uses the Storage Class setting for zHyperLink write eligibility. zHyperLink write eligibility is governed by the Storage Class. If no storage class is associated with the data set, it cannot use zHyperLink for writes. If the data set is non-SMS, zHyperLink is not used to write this data set.

▶ **DELETE**

Specifies that the attributes should be set to default. This parameter is mutually exclusive with ZHLREAD and ZHLWRITE. For SMS-managed data sets, zHyperLink eligibility is based on the Storage Class attributes. For non-SMS data sets, zHyperLink is not used to read or write to this data set.

Example 4-5 shows the **VARY SMS** command and output to enable a data set to use zHyperLink for READ operations.

Example 4-5 Sample VARY SMS command

```
VARY SMS,DSNAME(APPL01.APPDATA.VSAM.DATA),ZHLREAD=YES
IGW286I zHyperLink for APPL01.APPDATA.VSAM.DATA 485
  Read=Yes Write=Def
  No Storage Class
```

SETIOS HYPERWRITE

The **SETIOS HYPERWRITE** command is used for turning the HYPERWRITE option on or off:

```
SETIOS HYPERWRITE=YES|NO
```

By default the option, is on. The HYPERWRITE option must be on for zHyperLink write.

Note: The status set that uses the **SETIOS HYPERWRITE** command is preserved during the IPL only. Update the parmlib member IECIOSxx to preserve the setting.

SETIOS ZHYPERLINKIZHL

The **SETIOS ZHYPERLINK** command that is described in this section specifies the operation mode of zHyperLink in the system.

As in the **IECIOSxx** specification, four different values are available for the **OPER** parameter in the **SETIOS ZHYPERLINK** command:

```
SETIOS ZHYPERLINK,OPER={ALL|READ|WRITE|NONE}
```

The defined value in the **ZHYPERLINK** statement specifies whether the zHyperLink facility is enabled on operations:

- ▶ ALL I/O operations
- ▶ READ I/O operations
- ▶ WRITE I/O operations
- ▶ DISABLED for all I/O operations

The **SETIOS ZHYPERLINK,OPER=ALL** commands activates zHyperLink for READ and WRITE I/Os. Whether zHyperLink is used for an I/O request depends on whether this request meets the eligibility requirements. The following examples show how to activate read I/Os for zHyperLink, and what is expected as a result of the use of the command:

```
SETIOS ZHYPERLINK,OPER=READ
```

Message IOS090I is displayed in response to the use of the **SETIOS ZHYPERLINK** command:

```
IOS090I SETIOS. ZHYPERLINK UPDATE(S) COMPLETE
```

Note: The mode that is activated that uses the **SETIOS ZHYPERLINK** command is preserved during the IPL only. Update parmlib member IECIOSxx to preserve the setting.

4.1.3 Configuration commands

Configuration commands allow you to change status of the resources that are defined to the system.

CF PFID(pfid),ONIOFF

The config PFID (**CF PFID**) command is used for configuring defined PCIe Function IDs online or offline. Optionally, you also can specify the **FORCE** parameter for the **OFFLINE** option.

If the PFID is allocated as shown in Figure 4-1 on page 80, z/OS requires the use of the **FORCE** option for configuring the PFID offline, even if zHyperLink is disabled.

Attention: **FORCE** is a powerful option. Never specify **FORCE** unless you understand all of its consequences for your system. The **FORCE** option configures the PFID offline unconditionally.

An example of the **CF PFID(pfid),ON** command response is shown in Figure 4-6.

```
CF PFID(0303),ON
IQP034I PCIE FUNCTION 00000303 ONLINE. 894
  PCIE DEVICE TYPE NAME = (8GB zHyperLink      ).
IEE504I PFID(303),ONLINE
IEE712I CONFIG  PROCESSING COMPLETE
```

Figure 4-6 Sample CF PFID(pfid),ON output

4.2 System automation

The use of system automation tools to monitor and report any relevant messages on z/OS systems was widely adopted by customers around the world. These tools allow operators to focus on activities that cannot be automated and require manual intervention. The operations team can even use system automation tools to assist them in performing specific tasks, or creating alerts that are based on defined system messages.

zHyperLink introduced or updated a set of messages that can be tracked to monitor zHyperLink enabled and disabled states, and link degraded states that can affect your system performance.

4.2.1 Error messages

Tracking error messages automatically is vital to ensure a timely response to any errors that can affect your availability or your application response time.

This section identifies error messages that are related to zHyperLink so that you can take corrective actions.

Example 4-6 shows an IOS637E error message stating that the zHyperLink task terminated.

Example 4-6 Sample IOS637E error message

```
IOS637E  THE ZHYPERLINK MANAGER TASK HAS TERMINATED.
ZHYPERLINK AVAILABILITY MAY BE AFFECTED.
```

This error message is displayed if the zHyperLink management task that is running under the IOSAS address space is ended. This task is responsible for starting zHyperLink and reacting to changes in their states.

If this error occurs, the available zHyperLinks still can handle requests if the system has available PFIDs. If the PCIe Function Identifier (PFID) status is changed or the link recovers from an error, the system does not react to those events.

Another error message that was added to report zHyperLink errors is message ID IOS639E. This message is displayed when the zHyperLink PCIe function fails recovery, and a manual intervention is required. Example 4-7 shows a sample output message.

Example 4-7 Sample IOS639E error message

```
IOS639E INTERVENTION REQUIRED FOR PFID pfid, reasontext
```

You can attempt to configure the PFID offline and online again to reset and recover the PFID. If the error persists, contact the IBM support center.

4.2.2 Informational messages

In addition to error messages, you might find it useful to track some informational messages that are displayed when zHyperLink or PCIe connectivity status change.

Example 4-8 shows a sample system message that states a PCIe function change. This message is issued whenever a zHyperLink status changes. The following statuses are possible:

- ▶ ONLINE
The z/OS PCIE FUNCTION is configured online and is ready to be used.
- ▶ OFFLINE
The z/OS PCIE FUNCTION was unconfigured and entered the standby state. It is still in the system configuration and can be reconfigured online for use.
- ▶ AVAILABLE FOR CONFIGURATION
The z/OS PCIE FUNCTION is in the system configuration. It is in the standby state and might need to be configured online before it can be used.
- ▶ NOT AVAILABLE FOR USE
The z/OS PCIE FUNCTION is removed from the system configuration and is no longer available for use.

Example 4-8 Sample IQP034I message

```
IQP034I PCIE FUNCTION 00000102 ONLINE.  
PCIE DEVICE TYPE NAME = (8GB zHyperLink)
```

You might want to track your PCIe function changes, especially those changes that might result in disabling zHyperLink connections, such as OFFLINE and NOT AVAILABLE FOR USE.

Another system message that also can be monitored is IOS640I, which states a change in zHyperLink port state. Check for unexpected changes in port status, and define procedures for extra checks and actions to be performed. Example 4-9 on page 93 shows a sample IOS640I message stating that the port state changed to an operational state. The following values are possible:

- ▶ Operational
The port is operational.
- ▶ Not Operational
The port is not operational.
- ▶ None
The state of the port is not known.
- ▶ Undefined
The state of the port is undefined. Contact the IBM Support Center if this state is displayed.

For more information about the IOS640I message, see *z/OS MVS System Messages, Vol 9 (IGF-IWM)*, [SA38-0676](#).

Example 4-9 Sample IOS640I message

```
IOS640I PORT STATE HAS CHANGES FOR 00000102, PCHID=0140, PORT 1  
PORT STATE: Operational
```

Also, the IOS090I message was updated to inform when zHyperLink IOS updates were completed. Example 4-10 shows the result from a **SETIOS** command to update zHyperLink operation.

Example 4-10 Sample IOS090I message

```
IOS090I SETIOS. ZHYPERLINK UPDATE(S) COMPLETE
```

You also can set your system automation to issue regular commands to check the zHyperLink status, and take the necessary actions from there. For more information about the system commands that display zHyperLink information, see 4.1, “System commands” on page 80.

4.3 System messages

System messages are often used to determine the system health. In many cases, further effects can be avoided by an error prevention technique or by a system operator when automation setups are performed for a specific message.

System messages access the status of the operating system, optional software products, and applications. IBM System Automation can be an important component to help monitor zHyperLink operation.

Tracking error messages with IBM System Automation is vital to ensure a timely response to any errors that might affect your availability or application response time. You can set it to issue regular commands and check zHyperLink status with command outputs.

Note: For more information about system commands that are available to display and manage zHyperLink, see 4.1, “System commands” on page 80.

The message prefix for PCIe system messages is IQP. This prefix includes all PCIe functions, and not only zHyperLink. Other message prefixes that are relevant to zHyperLink are IEE, which include output from display matrix commands and IOS that are Input/Output supervisor messages for the PCIE started task and zHyperLink functions.

The scope and content of system messages can be updated as part of upgrades or maintenance implementation. For more information about latest edition of messages, see this [IBM Documentation web page](#).

4.3.1 Error messages

This section identifies error messages that are related to zHyperLink so that you can take corrective actions. We suggest monitoring these messages to ensure that suitable actions are taken if a message occurs.

IOS637E

The IOS637E error message states that zHyperLink task terminated, as shown in Example 4-11.

Example 4-11 Sample IOS637E message

```
IOS637E THE ZHYPERLINK MANAGER TASK HAS TERMINATED.  
ZHYPERLINK AVAILABILITY MAY BE AFFECTED.
```

This error message is displayed whenever the zHyperLink management task that is running under the IOSAS address space is ended. This task is responsible for starting zHyperLink and reacting to changes in their states.

If this error occurs, the available zHyperLinks still can handle requests if the system has available PFIDs. If the PCIe Function Identifier (PFID) status is changed or the link recovers from an error, the system does not react to those events.

Contact IBM Support if an error occurs. The system must be restarted to recover the function.

IOS639E

This message is displayed if zHyperLink PCIe function fails recovery. In this case, a manual intervention is required. Example 4-12 shows a sample output message.

Example 4-12 Sample IOS639E message

```
IOS639E INTERVENTION REQUIRED FOR PFID pfid, reasontext
```

In this case, you can attempt to configure the PFID offline and online again to reset and recover the PFID. However, if the error persists, contact IBM Support.

4.3.2 Informational messages

In addition to error messages, informational messages are displayed when zHyperLink or PCIe connectivity status changes, or as a command output. These messages are described next.

IOS090I

The IOS090I message was updated in support of zHyperLink as a response to **SETIOS ZHYPERLINK, OPER={ALL|READ|WRITE|NONE}**. It provides information about when zHyperLink IOS change was completed. Example 4-13 shows the result from a **SETIOS** command to update zHyperLink operation.

Example 4-13 Sample IOS090I message

```
IOS090I SETIOS. ZHYPERLINK UPDATE(S) COMPLETE
```

IOS634I

The IOS634I messages display the status of the HYPERWRITE and ZHYPERLINK options that are specified in the IECIOSxx parmlib member or set by using the **SETIOS** system command. Example 4-14 shows the result from a **SETIOS,ZHYPERLINK,OPER=ALL** command to enable zHyperLink reads and writes.

Example 4-14

```
SETIOS ZHYPERLINK,OPER=ALL
IOS090I SETIOS. ZHYPERLINK UPDATE(S) COMPLETE
```

IOS640I

The IOS640I message is issued when the state of a zHyperLink port changed. It identifies the physical port and its state, and the PFID that is associated with that port for this LPAR. This message is displayed only if the PFID is online to the current LPAR. Example 4-15 includes a sample message display.

Example 4-15 Sample IOS640I message

```
IOS640I PORT STATE HAS CHANGES FOR 00000102, PCHID=0140, PORT 1
PORT STATE: Operational
```

The following statuses are possible for PORT STATE on the IOS640I message:

► Not Operational

The port is not operational. The following information might be displayed that indicates why the port is not operational:

– Link down

The port is not operational because the link is down.

– Control unit not responding

The link is up, but the control unit is not responding to the start sequence.

– Fenced by processor

The port was fenced by the processor because the number of errors that was detected exceeded the internal threshold.

– Fenced by control unit

The port was fenced by the control unit because the number of errors that was detected exceeded the internal threshold.

– Processor side in service mode

The port is not operational because the processor side of the link is in service mode (for example, a support person is testing or replacing the optics).

– Control unit side in service mode

The port is not operational because the control unit side of the link is in service mode (for example, a support person is testing or replacing the optics).

– Insufficient control unit resources

► The port is not operational because not enough resources are available to support the link. For example, not enough CPUs are available on the control unit to support the link.None

The state of the port is not known.

- ▶ Undefined
The state of the port is undefined. Contact IBM Support if this state is displayed.
- ▶ Operational
If the port state is operational, the system allocates the PFID for zHyperLink requests.

Note: If the state is not operational, none, or undefined, the system deallocates the PFID (if it was previously allocated), and it becomes unavailable for zHyperLink requests.

IOS644I

The IEE644I message is the output from the **D IOS,ZHL, LINK=** command. The display shows PCHID, Port, Link, Port state and manufacturer, and serial for the subsystem. For each PCHID and Port, the number of available and unavailable PFIDs are displayed.

PCHID and Port represent the physical port on the CPC. Link represents the physical port on the disk subsystem.

Offline PFIDs are not included in the display.

Information about WWNN, PFID, and S/W state is displayed if the **DETAIL** parameter was included in the command. Also, the number of successful and link busy read and write operations are displayed (unless those numbers are zero).

IOS645I

The IEE644I message is the output from the **D IOS,ZHL, PFID=** command. The display shows PFID, PCHID, Port, Link, S/W, and Port State, plus manufacturer and serial for the subsystem.

PCHID and Port represent the physical port on the CPC while Link represent the physical port on the disk subsystem.

Offline PFIDs are not included in the display.

IOS646I

The IEE644I message is the output from the **D IOS,ZHL, PFID=pfid,DETAIL** command. The display shows PFID, PCHID, Port, Link, S/W, and Port State plus CU node descriptor, including serial for the subsystem.

Information about WWNN and the number of successful and link busy read and write operations are displayed (unless those numbers are zero).

IEE148I

The IEE148I is the result of a **CF PFID(pfid),OFF** command of an allocated PFID. For zHyperLink, an allocated PFID can be configured OFFLINE only when the **FORCE** option is used, even if zHyperLink is disabled (see Example 4-16).

Example 4-16 Sample IEE148I message

```
IEE148I PFID(303) NOT RECONFIGURED - PCI FUNCTION CURRENTLY IN USE
```

IEE587I

The IEE587I message was introduced in support of zHyperLink as a response to the **D M=DEV(devno), ZHYPERLINK** command. In Example 4-17, we display the zHyperLink capability on the device. The display shows whether the device is enabled for zHyperLink READ operations, WRITE operations, or both READ and WRITE operations. If zHyperLink capability is disabled for the device or disabled for only READ or WRITE operations, the reasons why are displayed.

Example 4-17 Sample IEE587I message

```
IEE587I 16.05.05 DISPLAY M 652
DEVICE 09000 STATUS=ONLINE
DEVICE IS ENABLED FOR ZHYPERLINK
READ AND WRITE OPERATIONS ARE ENABLED
ZHYPERLINKS AVAILABLE = 2
```

IQP024I

This message is issued in response to the **DISPLAY PCIE, PFID=<pfid>** system command. This command shows the status of one specific z/OS PCIE FUNCTION or device when it is in the configuration, and might be issued to monitor PCIe for zHyperLink. If the PCIE FUNCTION requested is not in the configuration file, system replies that indicate that the PFID that is displayed is unavailable.

The IQP024I command output includes the following structure:

```
IQP024I hh.mm.ss DISPLAY PCIE
PcieName PcieAsid status
text
```

Example 4-18 shows a status of PCIE for PFID 0202.

Example 4-18 Example of command DISPLAY PCIE,PFID=0202

```
D PCIE=PFID=0202
IQP024I 16.08.17 DISPLAY PCIE 669
PCIE      0012 ACTIVE
PFID      DEVICE TYPE NAME          STATUS  ASID  JOBNAME  CHID VFN  PN
00000202  8GB zHyperLink                ALLC   0019  IOSAS    013C 0003  2
CLIENT ASIDS: NONE
CU WWNN: 500507630AFFD049  CU Link Id: 0181
S/W State: Allocated
Port State: Operational
CU Node Descriptor: 002107.996.IBM.75.0000000LBN71
```

The highlighted **ACTIVE** statement is the STATUS of the PCIe address space. It can feature the following values:

- ▶ NOT ACTIVE
The z/OS PCIe address space was not started.
- ▶ ACTIVE
The z/OS PCIe address space is active.
- ▶ ENDED
The z/OS PCIe address space ended.

The following fields are in the text section and include the values that are displayed when the previous command is issued:

▶ PFID

PCIE Functional-ID

▶ DEVICE NAME

The printable name of the device type for the listed PFID. If the device was not named, the name is displayed as UNNAMED (*aaaa bbbb*), where *aaaa* is the vendor ID and *bbbb* is the device ID.

▶ STATUS

The STATUS is for the device. It can have one of the following values:

– ALLC

The device is allocated or in use.

– ENBL

The device is enabled but not fully started for use.

– CNFG

The device is configured online.

– STNBY

The device is in standby mode and ready to be configured online.

– DP

The device is deallocate-pending and is waiting for a deallocate command from its owner to clean up its resources.

– PERR

The device is in permanent error. It must be unconfigured to recover from this status.

▶ ASID

The address space ID that owns the z/OS PCIe device that was identified by the PFID. The ASID is not displayed if the device is not owned.

▶ JOBNAME

The job name of the owning address space.

▶ CHID

The physical or virtual channel identifier.

▶ VFN

The virtual PCIe function number.

▶ PN

The physical port number for the PCIe device. If blank, the port is not applicable or more than one port is associated with the PCIe device.

IQP034I

The IQP034I message displays the new status of a PCIE FUNCTION that was changed. The IQP034I message features the following structure:

```
IQP034I  PCIE FUNCTION pfid dev_status.  
PCIE DEVICE TYPE NAME = (DevTypeName)
```

Example 4-19 shows a sample IQP034I message.

Example 4-19 Sample IQP034I message

```
IQP034I PCIE FUNCTION 00000102 ONLINE.  
PCIE DEVICE TYPE NAME = (8GB zHyperLink).
```

The z/OS PCIE FUNCTION can have the following statuses displayed by IQP034I:

- ▶ ONLINE
The z/OS PCIE FUNCTION is configured online and is ready to be used.
- ▶ OFFLINE
The z/OS PCIE FUNCTION was unconfigured and entered the standby state. It is still in the system configuration and can be reconfigured online for use.
- ▶ AVAILABLE FOR CONFIGURATION
The z/OS PCIE FUNCTION is in the system configuration. It is in the standby state and might need to be configured online before it can be used.
- ▶ NOT AVAILABLE FOR USE
The z/OS PCIE FUNCTION is removed from the system configuration and is no longer available for use.

As critical resources for zHyperLink, you might want to track your PCIe functions, which can result in disabling zHyperLink connections.

Although we displayed some of the available messages for zHyperLink monitoring, you might want to check for other messages to build your own automation setup. For more information about PCIe-related messages, see *z/OS MVS System Messages, Vol 9 (IGF-IWM)*, SA38-0676.

4.4 Hardware Management Console

This section documents Hardware Management Console (HMC) functions that are available to monitor and operate zHyperLink. The following procedure gets zHyperLink functions on the HMC:

1. To log on to the HMC, use your browser and point to the HMC URL: `https://<HMC_IP>`.

The HMC supports Multi-factor Authentication, and has five different roles for user IDs. The Advanced Facilities tasks that are described in this section are available for the Sysprog and Service roles. The Configure ON/OFF tasks are available for the Operator, Advanced, Sysprog, and Service roles.

For more information about log-on options and user ID roles, see [this IBM Support web page](#).

2. After logging on to the HMC, select **Systems Management** and then, the CPC.

3. Select the **Recovery** task group, and the **Single Object Operation** task for logging on to the Support Element (SE).
4. When you are logged on to the SE, select **Systems Management**.
5. Select the CPC and then, **Channels**, and the Channel ID of the zHyperLink channel to be managed. After selecting the Channel ID, options that are described next are available for managing zHyperLink. All tasks are available in the Channel Operations group. Some of the tasks also are available in the CHPID Operations task group.

Note: The PCIe zHyperLink is defined at the CPC level, not at the Logical Channel Subsystem (LCSS) level or the LPAR level. zHyperLink Channels are not Channel Path IDs (CPHIDs); therefore, you cannot toggle it online/offline by using the Configure Channel Path ON/OFF task. It is also not available on the System Input/Output Configuration Analyzer task.

Viewing adapter details

To view adapter details, select **Adapter details**.

The window displays the physical location of the adapter within the CPC, the operating status and the LPARs capable of accessing the adapter. Function IDs for each LPAR also are displayed.

Figure 4-7 on page 101 shows an example of the Adapter Details window.

General

Name: 021C
System: ARIES
Location: Z01B-D110J.01-D210J.01
Type: zHyperLink Express

Status

Status: ✔ Operating
State: Online

Acceptable status

<input checked="" type="checkbox"/> Operating	<input type="checkbox"/> Loss of signal	<input type="checkbox"/> Test mode
<input type="checkbox"/> Suspended	<input type="checkbox"/> Loss of synchronization	<input checked="" type="checkbox"/> Bit error threshold exceeded
<input type="checkbox"/> No power	<input type="checkbox"/> Not operational link	<input checked="" type="checkbox"/> IFCC threshold exceeded
<input type="checkbox"/> Service	<input type="checkbox"/> Sequence time-out	<input type="checkbox"/> Stopped
<input type="checkbox"/> Not defined	<input type="checkbox"/> Sequence not permitted	<input type="checkbox"/> I/O suppressed
<input checked="" type="checkbox"/> Definition error	<input type="checkbox"/> Terminal condition	<input type="checkbox"/> Fabric login sequence failure
<input type="checkbox"/> Wrap block	<input type="checkbox"/> Offline signal received	<input type="checkbox"/> Port login sequence failure
<input type="checkbox"/> Check stopped	<input type="checkbox"/> Initializing	<input type="checkbox"/> State change registration failure
<input type="checkbox"/> Permanent error	<input type="checkbox"/> Degraded	<input type="checkbox"/> Invalid attachment failure

Apply acceptable status settings to all adapters

Partitions

Partition Function IDs
Function IDs (FIDs) defined to this adapter for all partitions

Partition name	Partition status	FID
ARIES01	Operating	0300
ARIES01	Operating	0400
ARIES21	Operating	0301
ARIES21	Operating	0401
ARIES22	Operating	0302
ARIES22	Operating	0402
ARIES23	Operating	0303
ARIES23	Operating	0403
ARIES24	Operating	0304
ARIES24	Operating	0404
ARIES25	Operating	0305
ARIES25	Operating	0405

Figure 4-7 Adapter details

Viewing Port Parameters window

To view the port parameters, complete the following steps:

1. Under Channel Operations, select **Advanced Facilities** → **Card Specific Advanced Facilities**.
2. Select **View Port Parameters** and then, select the port (1 or 2).

The window displays the WWNN and Link IDs for both ends of the connection. It also includes other information, such as the Link Speed and Transmitter/Receiver power rates for all eight lanes.

Figure 4-8 shows an example of the View Port Parameters window.

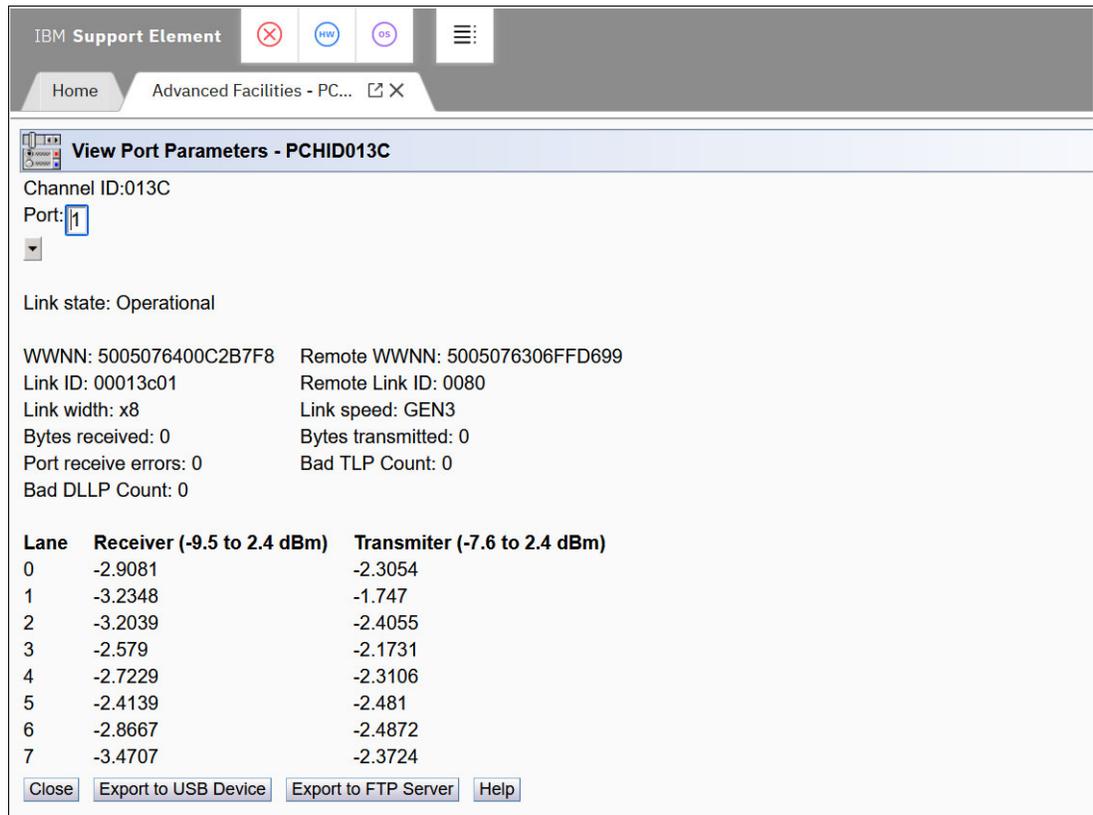


Figure 4-8 Port parameters

Running Port Diagnostics window

To run port diagnostics, complete the following steps:

1. Under Channel Operations, select **Advanced Facilities** → **Card Specific Advanced Facilities**.
2. Select **Run port Diagnostics** then, select the port (1 or 2).

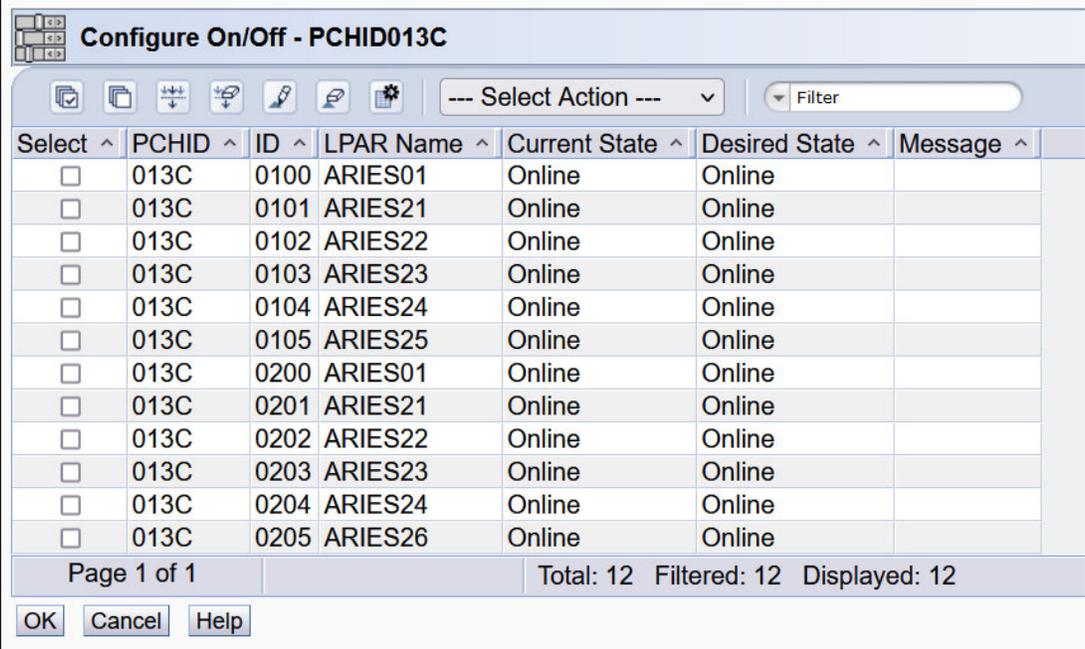
Use this window to test the hardware ports on the selected zHyperLink channels.

Attention: A diagnosis test cannot be stopped after it is started.

Configuring On/Off window

Select **CHPID Operations** or **Channel Operations** and then, **Configure On/Off** to configure one or more function IDs online or offline to one or more partitions.

Figure 4-9 shows an example of the Configure On/Off window.



The screenshot shows a window titled "Configure On/Off - PCHID013C". It features a toolbar with icons for selection, copy, paste, undo, redo, and help. Below the toolbar is a table with columns: Select, PCHID, ID, LPAR Name, Current State, Desired State, and Message. The table lists 12 rows of LPARs, all with a Current State of "Online" and a Desired State of "Online". At the bottom of the window, there are buttons for "OK", "Cancel", and "Help", and a status bar showing "Page 1 of 1", "Total: 12", "Filtered: 12", and "Displayed: 12".

Select ^	PCHID ^	ID ^	LPAR Name ^	Current State ^	Desired State ^	Message ^
<input type="checkbox"/>	013C	0100	ARIES01	Online	Online	
<input type="checkbox"/>	013C	0101	ARIES21	Online	Online	
<input type="checkbox"/>	013C	0102	ARIES22	Online	Online	
<input type="checkbox"/>	013C	0103	ARIES23	Online	Online	
<input type="checkbox"/>	013C	0104	ARIES24	Online	Online	
<input type="checkbox"/>	013C	0105	ARIES25	Online	Online	
<input type="checkbox"/>	013C	0200	ARIES01	Online	Online	
<input type="checkbox"/>	013C	0201	ARIES21	Online	Online	
<input type="checkbox"/>	013C	0202	ARIES22	Online	Online	
<input type="checkbox"/>	013C	0203	ARIES23	Online	Online	
<input type="checkbox"/>	013C	0204	ARIES24	Online	Online	
<input type="checkbox"/>	013C	0205	ARIES26	Online	Online	

Page 1 of 1 Total: 12 Filtered: 12 Displayed: 12

OK Cancel Help

Figure 4-9 Configure On/Off Channel ID window

Note: z/OS system commands must be used for configuring Channel IDs online and offline on active systems, if possible. The operating system is not notified when the HMC interface is used.

4.5 Troubleshooting

This section documents several examples of how to identify and solve zHyperLink-related issues, such as zHyperLink not being allocated or being disabled for devices. For more information about performance issues, see 3.7, “Measuring after deployment” on page 73.

4.5.1 zHyperLink port in CNFG status

To display the port status and troubleshoot, complete the following steps:

1. Issue the **D PCIE** system command to display the port status. In this example, a zHyperLink adapter port is in status CNFG, as shown in Figure 4-10.

```
D PCIE,PFID=0202
IQP024I 07.19.48 DISPLAY PCIE 243
PCIE      0012 ACTIVE
PFID      DEVICE TYPE NAME          STATUS ASID  JOBNAME  CHID VFN  PN
00000202  8GB zHyperLink                CNFG                    013C 0003 2
CLIENT ASIDS: NONE
CU WNN: Not available CU Link Id: Not available
S/W State: Error
Port State: Not Operational
CU Node Descriptor: Not available
```

Figure 4-10 zHyperLink adapter port in CNFG status

2. Issue the **D M=CU(cuno)** system command, which in this example displays a Software Status of Error and Port Status of NotOper, as shown in Figure 4-11.

```
D M=CU(0233)
IEE174I 07.22.53 DISPLAY M 249
CONTROL UNIT 0233

ZHYPERLINKS
PFID      PCHID  Port  LinkId  S/W St  Port St
00000102  013C   01   0080   Error   NotOper
00000302  021C   01   0380   Alloc   Oper
```

Figure 4-11 zHyperLink Error NotOper status

In this case, the cable between zHyperLink adapters in the IBM z14 CPC and zHyperLink adapters on the DS8880 was not connected.

4.5.2 zHyperLink port in STNB status

To display the port status and troubleshoot, complete the following steps:

1. Issue the **D PCIE** system command to display status. In this example, a zHyperLink adapter port is in status STNB, as shown in Figure 4-12.

```
D PCIE
IQP022I 07.25.07 DISPLAY PCIE 251
PCIE      0012 ACTIVE
PFID      DEVICE TYPE NAME          STATUS ASID  JOBNAME  CHID VFN  PN
00000102  8GB zHyperLink                 STNBY
00000202  8GB zHyperLink                 ALLC   0019  IOSAS   013C 0003 2
```

Figure 4-12 zHyperLink adapter port in STNB status

2. Issue the **D PCIE,PFID=0102** system command and review message IQP024I. In this example, S/W State is Not Available, while the Port State is Operational, as shown in Figure 4-13.

```
D PCIE,PFID=0102
IQP024I 07.28.13 DISPLAY PCIE 253
PCIE      0012 ACTIVE
PFID      DEVICE TYPE NAME          STATUS ASID  JOBNAME  CHID VFN  PN
00000102  8GB zHyperLink                 STNBY
CLIENT ASIDS: NONE
CU WWNN: 5005076306FFD699 CU Link Id: 0080
S/W State: Not Available
Port State: Operational
CU Node Descriptor: 002107.981.IBM.75.0000000FCC41
```

Figure 4-13 zHyperLink S/W State Not Available

3. Issue the **CF PFID(0102),ON** system command to configure the Function ID online.
4. Issue the **D PCIE** system command again to verify that the zHyperLink status is now ALLC.

4.5.3 zHyperLink is disabled

To display the device status and troubleshoot, complete the following steps:

1. Issue the **D M=DEV(devno)** system command to display device status. In this example, zHyperLink is not enabled for the device, as shown in Figure 4-14.

```
D M=DEV(9000)
IEE174I 07.45.13 DISPLAY M 266
DEVICE 09000 STATUS=ONLINE
CHP          40  42  41  43
ENTRY LINK ADDRESS  012C 012D 022C 022D
DEST LINK ADDRESS  010C 010B 020C 020B
PATH ONLINE        Y   Y   Y   Y
CHP PHYSICALLY ONLINE Y   Y   Y   Y
PATH OPERATIONAL   Y   Y   Y   Y
MANAGED            N   N   N   N
CU NUMBER          0690 0690 0690 0690
INTERFACE ID       0030 0230 0100 0300
MAXIMUM MANAGED CHPID(S) ALLOWED:  0
DESTINATION CU LOGICAL ADDRESS = 90
SCP CU ND          = 002107.996.IBM.75.0000000LBN71.0030
SCP TOKEN NED      = 002107.900.IBM.75.0000000LBN71.9000
SCP DEVICE NED     = 002107.900.IBM.75.0000000LBN71.9000
WWNN               = 500507630AFFD049
HYPERPAV ALIASES CONFIGURED = 16
FUNCTIONS ENABLED = MIDAW, ZHPF, XPAV
```

Figure 4-14 zHyperLink not enabled for device

The **FUNCTIONS ENABLED** must include ZHPF and ZHYPERLINK(R); however, ZHYPERLINK(R) is missing in the example.

2. Issue the **D M=DEV(devno),ZHYPERLINK** system command to obtain more details. The result is shown in Figure 4-15.

```
D M=DEV(9000),ZHYPERLINK
IEF196I IEF237I 93D5 ALLOCATED TO SYS00156
IEE587I 07.46.58 DISPLAY M 269
DEVICE 09000 STATUS=ONLINE
DEVICE IS DISABLED FOR ZHYPERLINK FOR THE FOLLOWING REASON(S):
  THERE ARE NO ZHYPERLINKS AVAILABLE
WRITE OPERATIONS ARE DISABLED FOR THE FOLLOWING REASON(S):
  CONTROL UNIT DOES NOT SUPPORT ZHYPERLINK WRITES
```

Figure 4-15 zHyperLink disabled for system

zHyperLink is disabled for the system.

3. Enable zHyperLink by issuing the **SETIOS ZHYPERLINK,OPER=READ** system command.
4. Verify the device again by using the **D M=DEV(devno)** system command. zHyperLink is now enabled.

4.5.4 zHyperLink is degraded

To display the control unit status and troubleshoot, complete the following steps:

1. Issue the **D M=CU(*cuno*)** system command to display control unit status. In this example, zHyperLink is degraded, as shown in Figure 4-16.

```
D M=CU(0232)
IEE174I 08.16.39 DISPLAY M 383
CONTROL UNIT 0232
CHP                40   42   41   43
ENTRY LINK ADDRESS 012C 012D 022C 022D
DEST LINK ADDRESS  0123 0102 0221 0206
CHP PHYSICALLY ONLINE Y   Y   Y   Y
PATH VALIDATED     Y   Y   Y   Y
MANAGED            N   N   N   N
ZHPF - CHPID      Y   Y   Y   Y
ZHPF - CU INTERFACE Y   Y   Y   Y
INTERFACE ID      0231 0233 0100 0101
MAXIMUM MANAGED CHPID(S) ALLOWED = 0
DESTINATION CU LOGICAL ADDRESS = 32
CU ND              = 002107.981.IBM.75.0000000FCC41.0231
CU NED             = 002107.981.IBM.75.0000000FCC41.3200
TOKEN NED          = 002107.900.IBM.75.0000000FCC41.3200
WWNN               = 5005076306FFD699
FUNCTIONS ENABLED = ZHPF, ZHYPERLINK, XPAV
XPAV CU PEERS     = 0232, 0236, 023A, 023E, 0242, 0246
DEFINED DEVICES
  09300-0936F
DEFINED PAV ALIASES
  09370-0937F
USABLE HYPERPAV ALIASES = 16
ZHYPERLINKS
  PFID      PCHID  Port  LinkId  S/W St  Port St
  00000102  013C   01   0080   Alloc  Degraded
  00000302  021C   01   0380   Alloc  Oper
```

Figure 4-16 zHyperLink degraded

Degraded connections can be caused by cable issues, such as a cable connector that is not clean.

2. Configure zHyperLink offline and request the IBM service representative to clean the connectors.
3. Configure zHyperLink back online and verify again.

How to verify cabling is described next.

4.5.5 zHyperLink cabling issue

If any of the following conditions are met, the cable might not be connected, needs its connectors cleaned, or needs to be replaced:

- ▶ If the Receiver or Transmitter dBm value is far beyond the valid range for all lanes (for example at -40), the cable is most likely disconnected.
- ▶ If the Receiver or Transmitter dBm value is -INF, the lane is broken and the cable must be replaced.
- ▶ If the Remote Link speed is GEN1 or GEN2, the link speed is degraded. The Remote Link Speed should be GEN3.
- ▶ One or more Receiver lane dBm values are outside the range of -9.5 dBm - +2.4 dBm.
- ▶ One or more Transmitter lane dBm values are outside the range of -7.6 dBm - +2.4 dBm.

To correct these conditions, complete the following steps:

1. Configure zHyperLink offline and request the CE to clean the connectors or replace the cable.
2. Configure zHyperLink back online and verify again.

For more information about the HMC Advanced Facilities window, see 3.1.7, “Verifying zHyperLink cables” on page 59.

4.5.6 Other reasons that zHyperLink is not used

zHyperLink is not used for a request for the following reasons:

- ▶ The I/O request does not meet the requirements.
- ▶ HPF is not enabled for the device.
IOS does not require zHPF to be enabled to use synch I/O; therefore, the channel driver does work, but media manager requires zHPF to be enabled.
- ▶ zHyperLink is not enabled in the user’s parameters; for example, Db2 **ZPARM**.
- ▶ The device is not in a usable state or is undergoing recovery:
 - Device offline
 - No paths available
 - Hot I/O
 - Device not ready
 - Validating paths after an MIH condition
- ▶ The device is reserved to another system or pending from this system.
- ▶ All zHyperLinks are busy with other requests.
- ▶ Not enough usable zHyperLinks are available to process the request; for example, more read requests than the number of available virtual functions.



zHyperLink checklist

This appendix provides a checklist for zHyperLink implementation on z/OS.

zHyperLink implementation checklist

In Table B-1, we provide a checklist that you can use as you install and enable zHyperLink in your systems. You can add any other steps your that might be relevant to your scenario.

Table A-1 Sample implementation checklist

Task	Reference
<input type="checkbox"/> Verify hardware level requirements	1.3.1, "Hardware prerequisites" on page 5
<input type="checkbox"/> Verify software level requirements	1.3.3, "Software requirements" on page 9
<input type="checkbox"/> Identify eligible workloads	2.1, "Identifying zHyperLink eligible workloads" on page 16
<input type="checkbox"/> Ensure host and DASD hardware are within zHyperLink range (150 m)	2.1.1, "zHyperLink in site configurations overview" on page 17
<input type="checkbox"/> Determine number of required zHyperLink adapters	2.2.1, "Number of connections" on page 27
<input type="checkbox"/> Measure environment before enabling zHyperLink	2.2.2, "Using Resource Measurement Facility data for sizing" on page 28
<input type="checkbox"/> Order Feature Codes for z CPC and DS8000	3.1.1, "Ordering features" on page 48
<input type="checkbox"/> Update the Microcode Levels on the ZCPC and IBM DS8000	3.1.2, "Updating microcode" on page 49
<input type="checkbox"/> Verify DS8000 IBM Z Synergy License	3.1.3, "Enabling the DS8000 zHyperLink feature" on page 49
<input type="checkbox"/> Install the Feature Codes for IBM Z CPC_ and DS8000	3.1.4, "Installing features" on page 50
<input type="checkbox"/> Connect the cables between ZCPC and the DS8000 subsystem	3.1.4, "Installing features" on page 50
<input type="checkbox"/> Verify whether IBM Z C.U.I.R. support is enabled	3.1.5, "Activating zHyperLink on the DS8000" on page 50
<input type="checkbox"/> Disable DS8000 I/O Priority Manager	3.1.5, "Activating zHyperLink on the DS8000" on page 50
<input type="checkbox"/> Activate DS8000 zHyperLink for Read I/O	3.1.5, "Activating zHyperLink on the DS8000" on page 50
<input type="checkbox"/> Update I/O Configuration	3.1.6, "Updating I/O configuration" on page 52
<input type="checkbox"/> Verify zHyperLink cables	3.1.7, "Verifying zHyperLink cables" on page 59
<input type="checkbox"/> Upgrade products to required version or release	3.2.2, "Product upgrades" on page 61
<input type="checkbox"/> Run the SMP/E REPORT MISSINGFIX for FIXCAT, and install APARs or PTFs if required	3.2.3, "Product maintenance" on page 62
<input type="checkbox"/> Verify that MACHMIG TX parameter is not specified inLOADxx member	3.3, "zHyperLink enablement" on page 63
<input type="checkbox"/> Verify zHyperLink status on PCIe links	3.3, "zHyperLink enablement" on page 63
<input type="checkbox"/> Verify enablement of zHPF	3.3, "zHyperLink enablement" on page 63

Task	Reference
<input type="checkbox"/> Enable zHyperLink on IOS	3.3, "zHyperLink enablement" on page 63
<input type="checkbox"/> Verify device status for zHyperLink	3.3, "zHyperLink enablement" on page 63
<input type="checkbox"/> Configure System Automation	3.4.1, "System automation" on page 69
<input type="checkbox"/> Activate VSAM for zHyperLink	3.5, "Activating zHyperLink for VSAM data sets" on page 70
<input type="checkbox"/> Activate Db2 v12 for zHyperLink	3.6, "Activating zHyperLink for Db2" on page 72
<input type="checkbox"/> Measure environment after enabling zHyperLink	3.7, "Measuring after deployment" on page 73



B

Problem determination

This appendix provides an overview of the questions to ask, problem determination tools, and techniques in a short format. The information that is presented is intended to help you with identifying, assessing, and solving issues that you might encounter with your synchronous I/O deployment.

This appendix includes the following topics:

- ▶ “Where do I look to see if zHyperLink is being used?” on page 114
- ▶ “Why is zHyperLink not used?” on page 114
- ▶ “Why are zHyperLink I/Os failing?” on page 115
- ▶ “zHyperLink cable problems” on page 116
- ▶ “Commands to use for problem determination” on page 116

Where do I look to see if zHyperLink is being used?

You can confirm whether zHyperLink is used by checking the following levels:

- ▶ Device
 - RMF Device activity report:
 - “S” appears next to device if I/Os are being done synchronously
 - Synchronous I/O activity report shows the details
 - ▶ zHyperLink connection:
 - RMF PCIE activity report
 - Shows activity on the zHyperLink connections from the IBM Z point of view
 - RMF ESS Link statistics report
 - Shows activity on the zHyperLink connections from the DS8000 point of view
 - **D IOS,ZHL,LINK=link,DETAIL** or **D IOS,ZHL,PFID=pfid,DETAIL**
 - Shows current activity on a zHyperLink connection (link) or for a PFID
- ▶ Data Set
 - SMF 42-6 records
 - Contain more statistics for synchronous I/O
 - zBNA tool
 - Shows zHyperLink usage
 - **D SMS,DSNAME(dsname),STATS(ZHLWRITE)**
 - DSNAME** is the data set that is used for Db2 log writes
- ▶ Application
 - Omegamon XE for Db2 Performance Expert on z/OS or equivalent product
- ▶ Storage System:
 - RMF Cache activity report
 - Shows zHyperLink activity from a storage system point of view
 - RMF ESS Link Statistics
 - Shows activity on the zHyperLink connections from the DS8000 perspective

Why is zHyperLink not used?

You can determine why zHyperLink is not used by asking the following questions:

- ▶ Is the device enabled for zHyperLink?
 - `D M=DEV(devno),ZHL`
 - This command covers many checks, including whether it is enabled at the system level, if usable links exist, whether Metro Mirror requirements are met, and so on.
- ▶ If not enabled at the device level because no usable links exist, check the state of the links and PFIDs by using the following commands:
 - `D IOS,ZHL,LINK`
 - `D IOS,ZHL,PFID`
 - `PCIE,`

D M=CU

- ▶ Is zHPF enabled for the device?
- ▶ Is the storage class enabled for zHyperLink?
- ▶ Is Db2 enabled for zHyperLink (ZPARM)?
- ▶ Is VSAM enabled for zHyperLink (IGDSMSxx parmlib)?
- ▶ Is the data set eligible for zHyperLink
 - For reads, CI size <= 4K, not VSAM NSR sequential, 16 byte buffer alignment for VSAM
- ▶ For writes, have you exceeded the maximum number of write sessions for a DS8000?

Why are zHyperLink I/Os failing?

You can determine why zHyperLink I/Os are failing by asking the following questions:

- ▶ Are there enough links to handle the I/O requests?
 - Link busy% in RMF device activity report
 - Local reject% in RMF PCIE activity report
 - Are some links down? Run the following commands:

```
D PCIE
D IOS,ZHL,LINK
PD IOS,ZHL,PFID
D M=CU
```
- ▶ Is the cache hit ratio low?
 - RMF device activity and cache reports for device and storage system level
 - zBNA analysis for data set level
- ▶ Do unsupported copy relationships exist?
 - z/OS disables zHyperLink for devices when Metro Mirror or global mirror requirements are not met
 - z/OS does not detect and disable because the device:
 - Is an IBM FlashCopy® target device (reads and writes)
 - In an XRC primary (writes)
- ▶ Do other errors exist?
 - GTF trace shows zHyperLink errors. Requesting I/O tracing only (no SSCH, no CCW) often is good enough to see the types of errors that are occurring.
 - **D IOS,ZHL,CU=cunum,DIAG:**
 - Internal command can be used to display the zHyperLink response code history, typically at the direction of L2.
 - zHyperLink response codes and qualifiers are internal only.

zHyperLink cable problems

Consider the following points when zHyperLink cable problems are investigated:

- ▶ When installing zHyperLink transceivers and cabling, the MPO/MTP-24 connector faces *must* be cleaned by using the IBC 'One-Click' MPO/MTP cleaning tool or equivalent.
- ▶ This process of cleaning connector faces *must* be done at the DS8K and the IBM Z CPC, even when the transceivers and cables are new and had dust covers on them.
- ▶ The MPO/MTP connections provide 12x the opportunity for failure as a regular HA connection

Commands to use for problem determination

Use the following commands for problem determination:

- ▶ `D M=DEV(devno),[ZHL|ZHYPERLINK]`
 - If I/O requests are not using zHyperLink, check whether zHyperLink is enabled for reads or writes for the device.
 - `D M=DEV(devno),[ZHL|ZHYPERLINK]` (see Figure B-1) shows whether it is enabled, and the reasons why if it is not enabled.
 - Check the following aspects in the command output:
 - Is zHyperLink enabled at the system level (IECIOSxx)?
 - Is zHyperLink enabled at the storage system (control unit) level?
 - Do usable zHyperLink connections exist?
 - Is zHyperWrite enabled (writes, Metro Mirror)?
 - Is the device HyperSwap managed and in duplex state (writes, Metro Mirror)?
 - It does not check enablement at the storage class or middleware level, CI sizes eligibility, and so on.

```
D M=DEV(980),ZHL

IEE587I 16.10.51 DISPLAY M 652
DEVICE 08400 STATUS=ONLINE
DEVICE IS ENABLED FOR ZHYPERLINK
READ OPERATIONS ARE ENABLED
WRITE OPERATIONS ARE DISABLED FOR THE FOLLOWING REASON(S):
  ZHYPERLINK WRITES ARE DISABLED FOR THE SYSTEM
  CONTROL UNIT DOES NOT SUPPORT ZHYPERLINK WRITES
  ZHYPERWRITE IS DISABLED FOR THE SYSTEM
THERE ARE NO ZHYPERLINKS AVAILABLE FOR SECONDARY 10980
CONTROL UNIT DOES NOT SUPPORT ZHYPERLINK WRITES FOR SECONDARY 10980
```

Figure B-1 the “`D M=DEV(devno),[ZHL|ZHYPERLINK]`” command output

- ▶ `D PCIE`
 - `D PCIE` command can be used to display PFID information. However, this command is generalized for all PCIE devices.
 - Displays the following information, which is not interesting or insufficient for zHyperLinks:
 - Type of PCIE function: 8 GB zHyperLink

- Virtual function
 - Jobname and asid where allocated: Always **IOSAS**
 - Status: **ALLC** is good, but anything else is bad and you must issue **D PCIE,PFID=pfid** to get more information
- New **D IOS,ZHYPERLINK** command can be used to display zHyperLink specific information

Display Link Summary

The following commands are available:

- ▶ Display summary information for all links (see Figure B-2):
D IOS,{ZHYPERLINK | ZHL},LINK[=ALL]
- ▶ Display summary information for a specific link by PCHID and port:
D IOS,{ZHYPERLINK | ZHL},LINK=pchid.port

```
D IOS,ZHL,LINK=ALL

IOS644I hh.mm.ss LINK SUMMARY
      CU Port      ---- CU Info --- -- PFIDS -
PCHID PN Link State  Mfg.Ser or WWNN Avail Unav
cccc p link pppppppp mmm.ssssssssssss nnnn nnnn
```

Figure B-2 D IOS,ZHL, LINK=ALL command

Where:

- ▶ The *pppppppp* represents the port state and show whether the link is operational, degraded, or not operational. If the link is not operational, it also shows whether it is fenced or in service mode.
- ▶ The *mmm.ssssssssssss* field represents the manufacturer and serial number from the node descriptor, if available. Otherwise, the WWNN is shown as Not available.
- ▶ The *nnnn nnnn* field represents the number of available (online and allocated) and unavailable (port not operational or allocation unsuccessful) PFIDs. Offline PFIDs are *not* included.

Display Control Unit Information

Control Unit information can be displayed by using the **D M=CU** command, as shown in Figure B-3.

```
D M=CU(700)
IEE174I 19.16.24 DISPLAY M 715
CONTROL UNIT 0700
CHP          43  44  45  20  65  75
ENTRY LINK ADDRESS  B355 B353 B157 B051 .. ..
DEST LINK ADDRESS   B37C B37D B17C B06B .. ..
CHP PHYSICALLY ONLINE Y   Y   Y   Y   Y   Y
...
ZHPF - CU INTERFACE Y   Y   Y   Y   Y   Y
INTERFACE ID       0501 0510 0401 0413 0320 0101
MAXIMUM MANAGED CHPID(S) ALLOWED = 0
DESTINATION CU LOGICAL ADDRESS = 07
CU ND             = 002107.000.IBM.PK.0000000000003.0001
CU NED            = 002107.F20.IBM.PK.0000000000003.0701
TOKEN NED         = 002107.900.IBM.PK.0000000000003.0700
WWNN              = world-wide-node-name
FUNCTIONS ENABLED = ZHPF, ZHYPERLINK
DEFINED DEVICES
  00700,00710-00717
DEFINED PAV ALIASES
  00718,00719,10718
ZHYPERLINKS
PFID    PCHID Port LinkId S/W St   Port St
pfid    pchid p   link  ssssss pppppppp
pfid    pchid p   link  ssssss pppppppp
```

Figure B-3 Display Control Unit information command

Display Write Session Information

This command displays the zHyperLink write tokens for a storage system. The token includes the device number and data set.

The **MACH=** keyword specifies the 10-character serial number from DS QD,devnum (see SCU-Serial in Figure B-4).

```
DS QD,0C00
IEE459I 07.04.45 DEVSERV QDASD 742
UNIT VOLSER SCUTYPE DEVTYPE          CYL  SSID SCU-SERIAL DEV-SERIAL EFC
00C00 H10C00 2107996 2107900          3339 1246 0175-LBL41 0175-LBL41 *OK
**** 1 DEVICE(S) MET THE SELECTION CRITERIA
**** 0 DEVICE(S) FAILED EXTENDED FUNCTION CHECKING

DS QD,MACH=0175-LBL41,ZHL
IEE459I 17.53.47 DEVSERV QDASD 786
UNIT VOLSER SCUTYPE DEVTYPE          CYL  SSID SCU-SERIAL DEV-SERIAL EFC
00C00 IN9000 2107996 2107900          3339 2400 0175-LBL41 0175-LBL41 *OK
ZHYPERLINK WRITE TOKENS TO STORAGE FACILTY
TOKEN DATE    TIME  UNIT  DATA SET NAME
06B7 10/09/18 17:53 0900D LBS1TST5.LDS.DATA
07B8 10/09/18 17:53 0900C LBS1TST5.LDS.DATA
****          2 WRITE TOKEN(S) MET THE SELECTION CRITERIA
****          1 DEVICE(S) MET THE SELECTION CRITERIA
```

Figure B-4 Display Write Session information

Display Data Set zHyperLink Write Statistics

When Db2 uses zHyperLink writes, the Db2 **DISPLAY LOG** command can be used to display whether the last log write used zHyperLink. This information does not provide any insight into how often this issue occurs or why zHyperLink cannot be used.

The following new command was created to display the zHyperLink write statistics for a data set:

```
D SMS,DSNAME(dsname),STATS(ZHLWRITE) [,RESET]
```

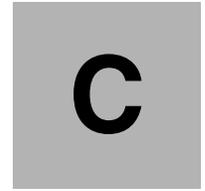
The command contains the following information (see Figure B-5):

- ▶ Overall statistics for the data set (summary)
- ▶ Statistics for individual devices
- ▶ Each device that is associated with a striped data set
- ▶ Metro Mirror secondary devices

```
D SMS,DSNAME,STATS(ZHLWRITE) Start of Report
DATA SET D2U.DSNUDB2.UD1G.LOGCOPY1.DS01.DATA
STATISTICS Since 03/07/2021 02:25:28.071289
SUMMARY
      TOTAL      %SYNC -----%ASYNC-----
WRITE REQUESTS WRITES  SKIP LNKBSY  -EST  MISC DISABL
      3197822    75.02    0.01   0.00 <0.01 <0.01  24.94
      -----%ASYNC-----
                MISS  DELAY  DUAL
                <0.01 <0.01  0.01
DEVICE STATISTICS
      TOTAL      %SYNC -----%ASYNC-----
SSID DEVNO WRITES WRITES  SKIP LNKBSY  -EST  MISC  MISS  DELAY
49C0 0105E 271e04  99.98   0.01   0.00 <0.01 <0.01 <0.01 <0.01
3813 05302 351e04  77.26   0.01   0.00 <0.01 22.72  0.00 <0.01
D SMS,DSNAME,STATS(ZHLWRITE) End of Report
```

Figure B-5 zHyperLink write statistics

As shown in Figure B-5, the SMF 42-6 record contains two new Synch I/O sections that further describe the statistics.



Sample IOCDS

The IOCDS zHyperLink statements for two CPC features that are assigned to three Logical Partitions (LPARs) are provided in this appendix.

Sample I/O Configuration Data Set (IOCDS) statements for the zHyperLink configuration are described in 3.1.6, “Updating I/O configuration” on page 52.

Sample IOCDs

The IOCDs statements for CPC1 at SITE1 with two zHyperLink adapters for two LPARs are shown in Example C-1. In this scenario, both ports on both adapters are configured, and two other LPARs are specified on the Candidate List.

Example C-1 Sample IOCDs for CPC1 at SITE1

```

ID      MSG1='zHyperLink sample IOCDs CPC1SIT1',          *
        MSG2='SYS1.IODF64 - 2021-07-12 13:44',          *
        SYSTEM=(8561,1),LSYSTEM=CPC1SIT1,              *
        .....
FUNCTION FID=1001,VF=1,PCHID=13C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1002,VF=2,PCHID=13C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1003,VF=3,PCHID=13C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1004,VF=4,PCHID=13C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1005,VF=5,PCHID=13C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1006,VF=6,PCHID=13C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1007,VF=7,PCHID=13C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1008,VF=8,PCHID=13C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1041,VF=1,PCHID=13C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=1042,VF=2,PCHID=13C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=1043,VF=3,PCHID=13C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=1044,VF=4,PCHID=13C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=1045,VF=5,PCHID=13C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=1046,VF=6,PCHID=13C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=1047,VF=7,PCHID=13C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=1048,VF=8,PCHID=13C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=1081,VF=1,PCHID=21C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1082,VF=2,PCHID=21C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1083,VF=3,PCHID=21C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1084,VF=4,PCHID=21C,                       *
        PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1085,VF=5,PCHID=21C,                       *
        PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1086,VF=6,PCHID=21C,                       *

```

```

PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1087,VF=7,PCHID=21C,*
PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=1088,VF=8,PCHID=21C,*
PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=1
FUNCTION FID=10C1,VF=1,PCHID=21C,*
PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C2,VF=2,PCHID=21C,*
PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C3,VF=3,PCHID=21C,*
PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C4,VF=4,PCHID=21C,*
PART=((LPZOSAPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C5,VF=5,PCHID=21C,*
PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C6,VF=6,PCHID=21C,*
PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C7,VF=7,PCHID=21C,*
PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C8,VF=8,PCHID=21C,*
PART=((LPZOSBPR),(LPZOS1DE,LPZOS2DE)),TYPE=HYL,PORT=2

```

The IOCDS statements for CPC2 at SITE2 with two zHyperLink adapters for two LPARs are shown in Example C-2. In this scenario, both ports on both adapters are configured, and another LPAR is specified on the Candidate List.

Example C-2 Sample IOCDS for CPC2 at SITE2

```

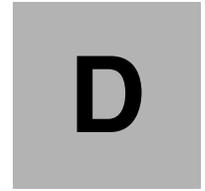
ID MSG1='zHyperLink sample IOCDS CPC2SIT2',*
MSG2='SYS1.IODF64 - 2021-07-12 14:12',*
SYSTEM=(8561,1),LSYSTEM=CPC2SIT2,*
.....*
FUNCTION FID=1001,VF=1,PCHID=140,*
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1002,VF=2,PCHID=140,*
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1003,VF=3,PCHID=140,*
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1004,VF=4,PCHID=140,*
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1005,VF=5,PCHID=140,*
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1006,VF=6,PCHID=140,*
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1007,VF=7,PCHID=140,*
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1008,VF=8,PCHID=140,*
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1041,VF=1,PCHID=140,*
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=1042,VF=2,PCHID=140,*
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=1043,VF=3,PCHID=140,*
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=1044,VF=4,PCHID=140,*

```

```

PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=1045,VF=5,PCHID=140,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=1046,VF=6,PCHID=140,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=1047,VF=7,PCHID=140,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=1048,VF=8,PCHID=140,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=1081,VF=1,PCHID=248,
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1082,VF=2,PCHID=248,
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1083,VF=3,PCHID=248,
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1084,VF=4,PCHID=248,
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1085,VF=5,PCHID=248,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1086,VF=6,PCHID=248,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1087,VF=7,PCHID=248,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=1088,VF=8,PCHID=248,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=1
FUNCTION FID=10C1,VF=1,PCHID=248,
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C2,VF=2,PCHID=248,
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C3,VF=3,PCHID=248,
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C4,VF=4,PCHID=248,
PART=((LPZOSDPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C5,VF=5,PCHID=248,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C6,VF=6,PCHID=248,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C7,VF=7,PCHID=248,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=2
FUNCTION FID=10C8,VF=8,PCHID=248,
PART=((LPZOSEPR),(LPZOS3DE)),TYPE=HYL,PORT=2

```



Mapping DCOLLECT data

DCOLLECT Record Type 'SC' is modified to support zHyperLink and display the new SC attributes zHyperLink Read and zHyperLink Write.

The mapping of DCOLLECT data is provided in this appendix.

DCOLLECT mapping

Two bits of CMM flags were assigned to display zHyperLink attributes:

- ▶ DSCHLERD being ON indicates that zHyperLink Read is eligible.
- ▶ DSCHLEWR being ON indicates that zHyperLink Write is eligible.

The mapping of DCOLLECT data is listed in Table D-1.

Table D-1 Sample of DCOLLECT

Offset	Type	Length	Name	Description
265(X'109')	UNSIGNED	1	DSCPAV	PAV requirements, 0 = None, 1 = Standard, 2 = Preferred 3 = Required
266(X'10A')	UNSIGNED	1	DSCSTOSL	OAM SUBLEVEL VALUE
267(X'10B')	BITSTRING	1	DSCVFLG2	VSAM FLAG 2
	1... ..		DSCDCLS	Disconnect Sphere at CLOSE 1 = YES 0 = NO
	.111 1111		*	RESERVED
268(X'108')	BITSTRING	1	DSCCMMFT	CMM Flags
	1... ..		DSCHLERD	Eligible for zHyperLink Reads 1 = YES 0 = NO
	.1... ..		DSCHLEWR	Eligible for zHyperLink Writes 1 = YES 0 = NO



Consistent Read from Metro Mirror Secondary

Consistent Read from Secondary (CRS) is a performance improvement in a z/OS Metro Mirror (MM) environment. It applies to FICON and zHyperLink configuration and is available with DS8000 Licensed Machine Code (LMC) 9.2.

This appendix includes the following topics:

- ▶ “Overview” on page 128
- ▶ “Consistent read management” on page 129
- ▶ “z/OS requirements and behavior” on page 129
- ▶ “Supported data sets” on page 131
- ▶ “Consistent read from secondary support for zHyperlink” on page 131
- ▶ “Extended Long Busy” on page 131
- ▶ “Enabling a device for Consistent Read from Secondary” on page 132

Overview

In a single site workload MM configuration, the host processor and primary storage are in the same site and the secondary storage is at some distance from the host processor (see Figure E-1). All read and write operations are performed locally to the primary storage (A1); writes are mirrored to the secondary storage (A2). If the primary storage is within 150 meters (492 feet) of the processor, zHyperLink can be used for eligible read and write operations.

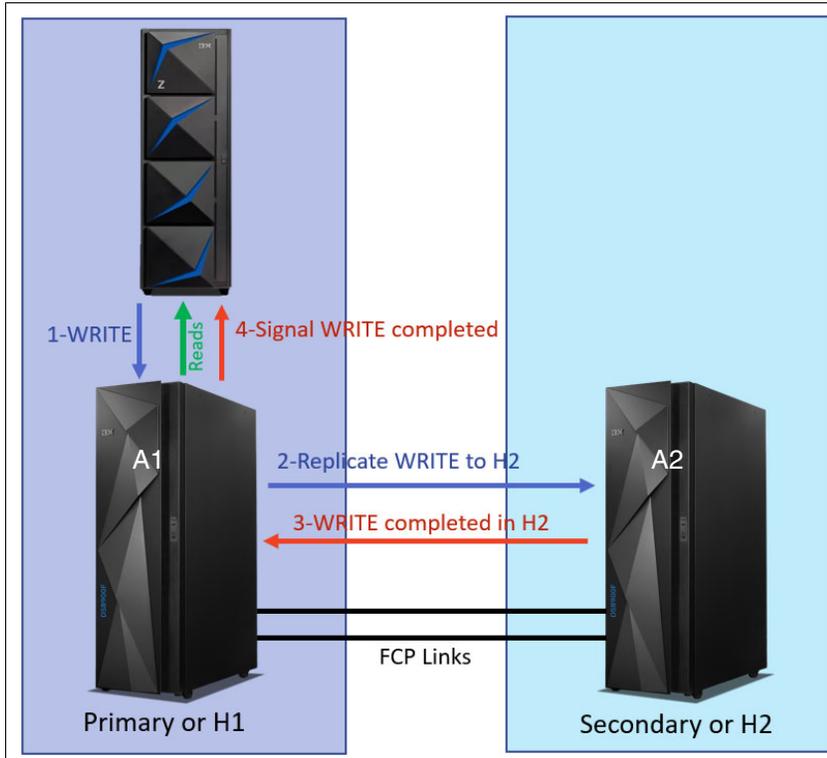


Figure E-1 HyperSwap configuration

After a HyperSwap is performed and the mirroring direction is reversed (A2 → A1), the secondary is now closer to the processor, and the primary is at distance, which means that reads and writes must travel over distance and zHyperLink cannot be used. I/O latency is improved if read operations can be issued to the secondary.

Another use case is in a multi-site workload configuration where host processors are in both sites that are accessing the storage systems that are in the MM relationship.

The processor at Site 1 can perform all of the read/write I/Os by using the local disk. However, the I/Os from Site 2 must travel over a distance to Site 1. I/O latency is improved if read operations can be issued to the secondary.

Consistent reads from Metro Mirror Secondary

Consistent read means that the data is read from the secondary and it is the same as though it were read from the primary storage. The following conditions must be met for this to be true:

- ▶ The application or middleware must not rely on behavior within the storage system to serialize concurrent read and write requests. That is, the application or middleware must have some type of serialization mechanism in place to ensure that read and write requests to the same set of tracks do not occur concurrently and must request the bypass extent checking attribute when issuing their I/Os. This process must be transparent to applications that use Db2, IMS, or VSAM directly or indirectly, because many of the I/Os that are issued by these components specify bypass extent checking.
- ▶ The MM pairs must be in *full-duplex* state; that is, the primary and secondary are identical and being kept in sync. They also must be part of a HyperSwap configuration.

Consistent read management

To enable consistent reads, z/OS must issue a request to start a consistent read management (CRM) session between the primary and secondary storage systems for each participating logical subsystem (LSS).

The CRM session maintains a heartbeat between the primary and secondary storage system.

If the secondary system does not receive a heartbeat from the primary within a predetermined period, reads are disallowed to the secondary system.

When the heartbeat is resumed, reads are allowed from the secondary system.

Consistent read management becomes active for an LSS when the first system issues a request to start consistent read management. Subsequent start requests by other systems are ignored.

CRM remains active, even if the system that started CRM fails or all sharing systems fail.

CRM is stopped if an MM (PPRC) suspension occurs or a specific request is issued to stop CRM.

z/OS requirements and behavior

z/OS enables a secondary device for consistent reads when the following conditions are true:

- ▶ Consistent read from secondary is enabled at the system level by specifying **READSEC=YES** on the **ZHPFOPTS** parameter in the **IECIOSxx** parmlib member or the **SETIOS** command.
- ▶ zHPF is enabled for the primary and secondary devices.
- ▶ The primary and secondary storage systems support the consistent read from secondary feature.
- ▶ The MM relationship is in duplex state and the primary and secondary devices are part of a HyperSwap configuration.
- ▶ The secondary storage system is significantly closer to the processor than the primary or the secondary has an online zHyperLink connection.

This is determined on a system-by-system basis because some systems might be closer to the primary while others are closer to the secondary.

If an online zHyperLink connection to the primary exists, the primary is in the local site; therefore, read I/Os are directed to the primary, regardless of whether a zHyperLink connection to the secondary exists.

If an online zHyperLink connection to the secondary exists but not the primary, eligible read I/Os are directed to the secondary to take advantage of zHyperLink, even if the primary is also in the local site.

If no online zHyperLink connections exist to the primary or secondary, z/OS issues a series of test I/Os to determine the relative distance. Then, it allows or disallows read I/Os to the secondary based on the results.

- ▶ A consistent read management session is active for the logical subsystem. z/OS starts CRM when all the preceding conditions (except for zHPF) are met. z/OS stops CRM if the HyperSwap configuration is purged or consistent read from secondary is disabled at the system level by specifying **READSEC=NO** on the **ZHPFOPTS** parameter in the **IECIOSxx** parmlib member or the **SETIOS** command.

If a zHyperLink read to the secondary is unsuccessful, the I/O request is retried asynchronously to the primary or secondary, depending on whether the I/O is likely to succeed on the secondary. For example, the following types of errors cause the I/O to be retried asynchronously on the secondary:

- ▶ A link was not assigned, all links were busy, or some type of link error occurred. This type of error affects only the ability to perform a zHyperLink I/O request.
- ▶ A read cache miss occurred. A zHyperLink read request cannot wait for data to be staged into cache from disk because it takes too long. However, this issue is not true for FICON I/O. On the secondary storage system, before reads to the secondary are performed, the cache contains only writes to the primary that were mirrored to the secondary. Initially, reads to the secondary can encounter cache misses. Eventually, as more reads to the secondary are performed, the cache hit ratio improves.
- ▶ A timeout occurred. For example, the storage system cannot access the track within a predefined period. These types of operational delays are not tolerated for zHyperLink I/O requests because of the latency requirements, but are tolerated for asynchronous I/O requests.

If the asynchronous retry to the secondary is unsuccessful, the I/O is retried on the primary. Regardless of whether the I/O is retried asynchronously on the primary or secondary, the next I/O request attempts to use zHyperLink to the secondary.

For I/O requests that are not eligible to use zHyperLink, if the asynchronous I/O request to the secondary is unsuccessful, the I/O request is retried on the primary. The next I/O request is issued to the secondary.

If reads are disallowed because the secondary did not receive the heartbeat within the predetermined period, read I/O requests to the secondary are rejected. Also, z/OS temporarily directs I/Os to the primary for 1 - 2 seconds to allow the primary and secondary time to resume the heartbeat, and to prevent unnecessary I/O errors to the secondary and retries to the primary.

Supported data sets

For zHyperLink requests, consistent read from secondary is supported for the same I/O requests that support zHyperLink read to the primary. Consider the following points:

- ▶ Db2 synchronous database reads where the control interval is 4 K.
- ▶ VSAM reads regardless of the record type (that is, KSDS, ESDS, RRDS, or VRRDS) or type of access (NSR, LSR, GSR, or RLS) except for NSR sequential. The control interval must be 4 K or less.

For non-zHyperLink requests, consistent read is supported for the following types of I/O requests:

- ▶ Db2 synchronous database reads, regardless of the control interval size.
- ▶ Db2 pre-fetch reads (sequential, dynamic, or list).
- ▶ VSAM reads, regardless of the record type or type of access (including NSR sequential) where the control interval (CI) size does not span tracks. For example, when a 32 K CI size is used, each 32 K CI consists of two 16 K physical records. Because three 16 K records fit on a track, some CIs span a track.
- ▶ IMS OSAM request: Linear data sets only.
- ▶ CVAF VTOC access services: This function provides service to search VTOC.
- ▶ z/OS File System (ZFS) reads.

Consistent read from secondary support for zHyperlink

The secondary storage might still “think” that it continues in full-duplex and it is not receiving any updates from the primary.

To address this scenario, a new Heart Beat (HB) function called *Consistent Read Management* was implemented in the code. With this function, the primary storage periodically sends an HB command to the secondary storage to indicate that the volumes are still in full-duplex. The HB signal starts a time interval window that is called *Read Allow Time*.

If the secondary storage continues to receive HB communication from the primary, the reads can be allowed. When the secondary gets an HB, it enables reads for a specific time window.

If the secondary stops receiving the HB, some type of communication problem might occur, and then it disallows reads after the ongoing Read Allow Time window expires.

Extended Long Busy

With MM, when a suspension or freeze occurs, the primary storage goes into an Extended Long Busy (ELB) state. The purpose of the ELB is to prevent any further dependent I/O on that set of volumes.

Normally, when a suspend occurs, the controlling software recognizes it and conducts a freeze. This freeze cuts all the communication paths from the primary to the secondary and causes all the volumes to go into a long busy state.

In this situation, when an attempt to write to the primary occurs, the host receives the long busy and does not complete.

Any I/O that comes after the long busy is not started because the previous I/O did not complete. After all the “freezes” are completed on the primary, storage systems know that the secondary is consistent. Next, a command that is called Consistency Group Created (also known as Run command), ends the Extended Long Busy (ELB) state, and I/O to the primary storage is allowed again. This behavior is considered normal MM behavior today. When the Run command is run, the long busy ends immediately.

The consistent reads introduced another check that adds time before releasing the ELB condition to ensure that the secondary storage stopped allowing reads before any more I/O is allowed to the primary.

Enabling a device for Consistent Read from Secondary

A secondary device is enabled for CRS when *all* of the following conditions are true:

- ▶ The Primary and Secondary storage systems support the function DS8900 with R9.2 level of microcode.
- ▶ The MM relationship is in a duplex state and a HyperSwap configuration is loaded.
- ▶ The secondary storage system is closer to the processor than the primary
Evaluated on a system-by-system base
- ▶ z/OS started Consistent Read Management for the LSS
CRM is the Heartbeat
- ▶ zHPF is enabled for the device

How it works

HyperSwap must be enabled because z/OS must know the mapping between the primary and the secondary volumes so that it knows from which secondary it reads. z/OS implemented code to detect which site is the closer site. This feature determines whether the read is done from the primary or the secondary.

Distance calculations are based on the Command Response Time (CMR) or the existence of zHyperLink connections. The lowest CMR indicates the apparent distance to the primary and secondary.

z/OS determines which storage systems are closer to the processor. Consider the following points:

- ▶ GDPS and SDM feature *site* awareness; that is, which storage systems are at which site
- ▶ GDPS features *processor* awareness; that is, the processor is in the same site as the storage system

The following algorithm is used to determine the use of primary or secondary:

- ▶ Use *Primary* if an online zHyperLink connection to the primary exists.
This use implies that the storage system is within 150 meters (XX feet) of the processor.
- ▶ Use *Secondary* if an online zHyperLink connection to the secondary exists.

Use the minimum initial command response (ICMR) to determine which storage is closer to the processor.

To enable CRS, IECIOSxx must be enabled by using **ZHPFOPTS READSEC=YES**:

- ▶ YES *enables* consistent read from secondary for the current system
- ▶ NO *disables* consistent read from secondary for the current system

ZHPF also must be enabled in IECIOSxx.

CRS stops when:

- ▶ HyperSwap configuration is purged
- ▶ READSEC parameter is set to NO

To read from secondary, a channel program uses a new bit in the Prefix Command (Aux. bit 3). When received, the DS8000 performs the following checks:

- ▶ CRS was enabled
- ▶ Full-duplex secondary
- ▶ Device is in the Eligible Device bitmap that was received from the primary
- ▶ The Read Allow Time did not yet expire

Related publications

The publications that are listed in this section are considered particularly suitable for a more detailed discussion of the topics that are covered in this paper.

IBM Redbooks

The following IBM Redbooks publications provide more information about the topic in this document. Note that some publications that are referenced in this list might be available in softcopy only:

- ▶ *System Management Interface Tool SMIT*, REDP-0105
- ▶ *IBM Z Connectivity Handbook*, SG24-5444
- ▶ *IBM z14 Technical Guide*, SG24-8451
- ▶ *IBM DS8880 and IBM Z Synergy*, REDP-5186

You can search for, view, download, or order these documents and other Redbooks, Redpapers, Web Docs, draft, and additional materials, at the following website:

ibm.com/redbooks

Other publications

The following publications also are relevant as further information sources:

- ▶ *IBM DS8880 Version 8 Release 3.3 Introduction and Planning Guide*, GC27-8525
- ▶ *Resource Measurement Facility Report Analysis Version 2 Release 3*, SC34-2665
- ▶ *MVS System Commands Version 2 Release 3*, SA38-0666
- ▶ *MVS System Messages Volume 9 (IGF-IWM) Version 2 Release 3*, SA38-0676

Online resources

The following websites also are relevant as further information sources:

- ▶ IBM Techdocs zBNA:
<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS5132>
- ▶ IBM Techdocs CP3KEXTR:
<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS4229>
- ▶ IBM Resource Link:
<https://www.ibm.com/server/resourcelink>
- ▶ IBM Data Storage Feature Activation:
<https://www-03.ibm.com/storage/dsfa/home.wss>

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services



REDP-5493-01

ISBN 0738460109

Printed in U.S.A.

Get connected

