

Coupling Facility Performance: A Real World Perspective

Comparison of different Coupling
Facility configurations

Considerations for
system-managed duplexing

Focus on DB2 and
WebSphere MQ



Frank Kyne
Mike Ebbers
George Handera
Doug Trimboli



International Technical Support Organization

Coupling Facility Performance: A Real World Perspective

March 2006

Archived

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

Archived

First Edition (March 2006)

© Copyright International Business Machines Corporation 2006. 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	v
Trademarks	vi
Preface	vii
The team that wrote this Redpaper	vii
Become a published author	vii
Comments welcome	viii
Chapter 1. Background	1
1.1 Setting the stage	2
1.2 Configuration prior to project	2
1.3 What was the objective of the test?	3
1.4 Configuration used for measurements	3
1.5 What tests were run?	5
1.5.1 Customer requirements	5
1.6 Terminology	6
1.7 Heuristics	6
Chapter 2. DB2 structure experiences	7
2.1 Introduction	8
2.2 DB2 lock structures	8
2.2.1 ICB versus ISC for lock structures	8
2.2.2 Lock structure duplexing	14
2.2.3 CF processor impact	18
2.3 DB2 GBP structures	22
2.3.1 ICB versus ISC for GBP structures	22
2.3.2 User-managed duplexing	26
Chapter 3. WebSphere MQ queue sharing structures	31
3.1 MQ Application structures	33
3.2 MQ Application and Admin structures in simplex	33
3.2.1 ISC links	33
3.2.2 ICB links	35
3.3 MQ Admin structures in duplex	37
3.3.1 ISC links	38
3.3.2 ICB links	39
Chapter 4. Results	41
4.1 Configuration selected	42
4.2 Lessons learned	42

Archived

Notices

This information was developed for products and services offered in the U.S.A.

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 give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law. 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 states 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

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

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.


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 the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates 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. 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 IBM's application programming interfaces.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

@server®
@server®
Redbooks (logo) ™
z/OS®
zSeries®

CICS®
DB2®
GDPS®
IBM®
MVS™

Parallel Sysplex®
Redbooks™
RMF™
WebSphere®

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

Preface

This Redpaper can help you understand the Coupling Facility technology and how the different configuration options can affect your workload. It is based on the results of a benchmark by a large IBM® client at the IBM Benchmark Center in Poughkeepsie, New York.

To avoid having misleading results, you must have a robust test environment if you are planning a similar evaluation in your environment.

The team that wrote this Redpaper

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

Frank Kyne is a Certified IT Specialist at the International Technical Support Organization, Poughkeepsie Center. He writes extensively and teaches IBM classes worldwide on all areas of Parallel Sysplex®. Frank is also responsible for the GDPS® product documentation. Before joining the ITSO seven years ago, Frank worked in IBM Global Services in Ireland as an MVS™ Systems Programmer.

George Handera is a systems engineer specializing in performance management and infrastructure architecture design. He has 28 years of experience in the performance field. He is currently working for a large insurance company.

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

Paola Bari
IBM Poughkeepsie

Mike Ebbers
IBM Poughkeepsie

Gary King
IBM Poughkeepsie

Doug Trimboli
IBM Poughkeepsie

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

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 Redpaper or other IBM Redbooks™ in one of the following ways:

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

ibm.com/redbooks

- ▶ Send your comments in an email to:

redbook@us.ibm.com

- ▶ Mail your comments to:

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



Background

This chapter describes the background of this project and the customer configuration and workload considerations that led to the decision to take part in this project. The objectives of the test and the applications and methodology that were used for the tests are discussed. This chapter also describes the configuration used for the tests, and defines the terminology and heuristics that were used for this IBM Redpaper.

1.1 Setting the stage

To accommodate normal business growth and the introduction of a new application, a large insurance company had a plan to move an entire production sysplex system from IBM 2064 to IBM 2084 processors. An integral part of the plan was to upgrade the Coupling Facilities (CFs) to deliver the required levels of performance and availability.

Because of concerns about a tendency for CF technology to be left behind the levels used for the operating systems, the company considered moving to an all ICF configuration (for more information about ICF, see 1.6, “Terminology” on page 6), where the CFs would run as logical partitions (LPARs) in the z/OS® processors. The company had used separate and external CFs in the past. However, before proceeding with such a move, they wanted to ensure that such a configuration could deliver the performance required, considering that the absence of external CFs would mandate the use of system-managed CF duplexing for some structures. They also wanted to understand the impact, if any, that CF CPU utilization and different types of CF links would have on CF performance.

To be able to obtain this information accurately the company decided to run benchmarks to verify how the different configurations fit with their workload and obtain the best performance and availability results. Their workload was primarily IBM CICS® and DB2®; however, WebSphere® MQ was a big part of that workload, along with a growing WebSphere for z/OS environment.

Specifically, the things that needed to be tested during the benchmarks were:

- ▶ ICF versus external CF
- ▶ Impact of different link types
- ▶ Impact of SM Duplexing

1.2 Configuration prior to project

The company configuration prior to the start of this project is shown in Figure 1-1.

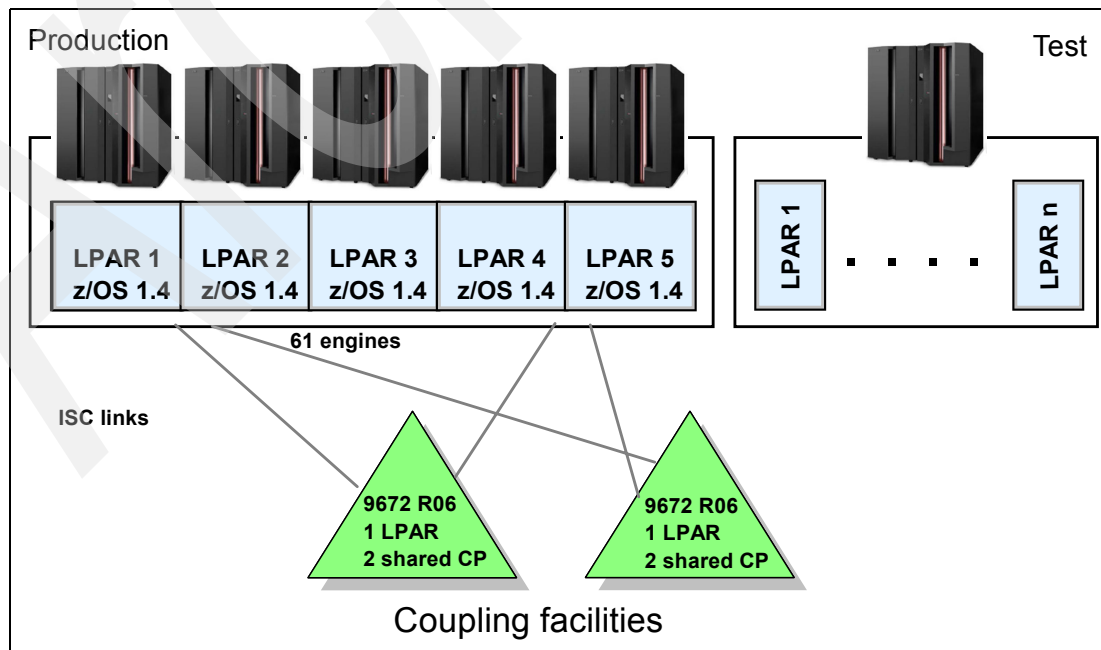


Figure 1-1 Configuration prior to the project

The configuration was set up as follows:

- ▶ Both production CFs were located in IBM 9672 R06s with two shared engines each. The CF LPAR was the only LPAR defined. The CFs were running CFlevel 11.
- ▶ The CFs were connected to the z/OS processors with ISC links.
- ▶ At the time of the test, IBM z/OS 1.4, DB2 V7, and WebSphere MQ were in use in the configuration.
- ▶ The production workload was distributed through five LPARs running on five IBM 2064s (multiple models) with a total of 61 engines.
- ▶ The test workload ran on multiple LPARs that were hosted in a single IBM 2064.
- ▶ The performance disparity between the z/OS processors (2064s) and the CFs (9672 G5) meant that a large number of CF requests were being converted to asynchronous requests, meaning long response times. Increasing the speed difference between the z/OS processors and the CFs by moving z/OS to 2084s would exacerbate this situation.

The most important workload for the installation, and one of the focuses of the test, was DB2, which was the largest user of CF resources. The DB2 systems had significant virtual storage problems caused by heavy partitioning and compression. This resulted in a constraint on the number of concurrently open data sets, which in turn resulted in additional CF activity when tablespace update interests were communicated.

The current DB2 data sharing configuration was performing with high CF structure service times.

1.3 What was the objective of the test?

The objectives of the benchmark were to develop an understanding of the performance attributes in a sysplex environment of the 2084 class of machines when used with a 2084 CF and to understand the effects of using various CF link types. The testing configuration facilitated the use of various combinations of Inter-System Channels (ISCs), Integrated Cluster Bus (ICB), and Internal Coupling (IC) links.

The company also wanted to understand the impact that CF CPU utilization has on service time with a focus on how the service times can be affected by the arrival rate.

The tests were also used to determine for the following components:

- ▶ The optimum service times for DB2 Group Buffer Pool and Lock structures
- ▶ The impact of duplex mode on DB2 Group Buffer Pool and Lock structures
- ▶ What factors trigger the use of the MQ Admin structure

Based on the results of the tests, the company would decide whether to switch to internal CFs or continue to use external CFs, what types of CF links should be used, whether they would use structure duplexing (and for which structures), and how or if they would exploit MQ shared queues.

1.4 Configuration used for measurements

The hardware configuration in the IBM Poughkeepsie Benchmark Center that was used for the tests is shown in Figure 1-2 on page 4.

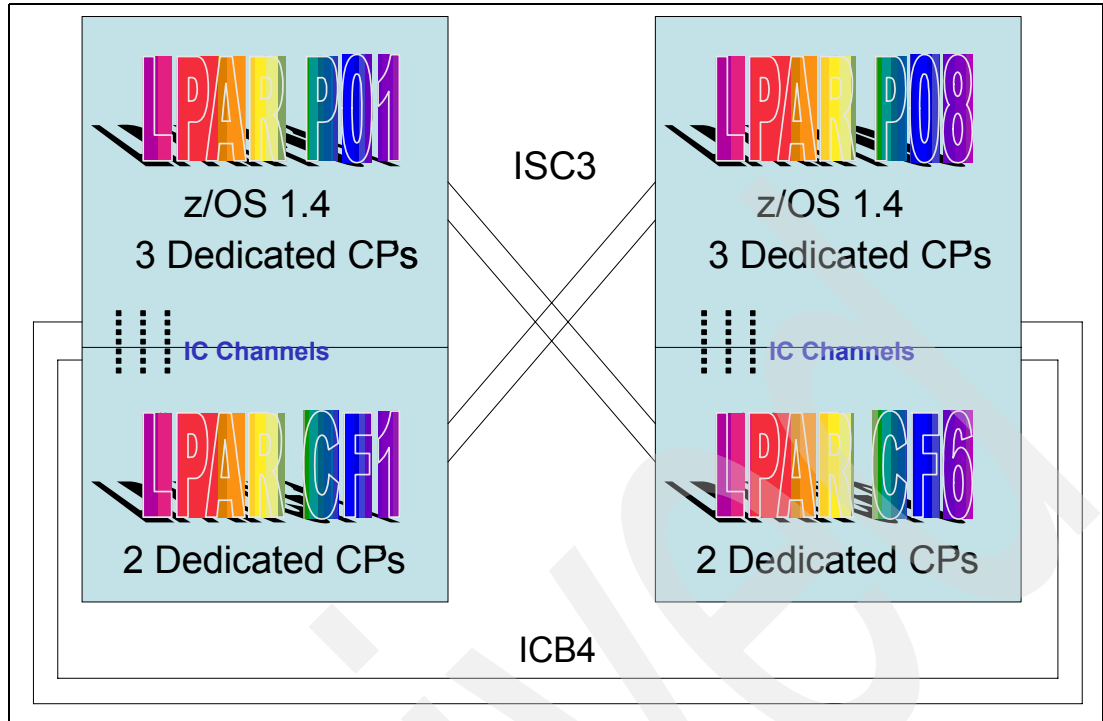


Figure 1-2 Configuration used during the test cases

Two 2084-B16 processors were used. Each processor had one z/OS LPAR that was defined with three dedicated CPUs, and one CF LPAR that was defined with two dedicated engines. From a CF link perspective:

- ▶ IC links (memory-to-memory) were used to communicate with the CF in the same physical processor. IC links enable very high-speed, efficient communication between the CF partition and one or more z/OS logical partitions running on the same IBM zSeries® server. An IC does not require any hardware or cabling.
- ▶ ISC links, in this case ISC-3, were available to be used to connect the z/OS image to the CF in the other processor. ISC links are point-to-point fiber connections that require a unique channel definition at each end of the link. ISC links support a maximum of 100 km between the CF and the connected z/OS.
- ▶ In addition, ICB links, in this case ICB-4, were available to connect the z/OS images to the CF in the other processor. These are an alternative to using the ISC links; they are faster than ISC links, attaching directly to a Self-Timed Interconnect (STI) bus of the server. They are copper rather than fiber links and support a maximum distance of about 7 m between the processors.
- ▶ LPAR P08 access to the CF in the other processor (CF1) was provided by either ICB or ISC links (both options were measured). Access to CF6 is through IC links.
- ▶ LPAR P01 access to the CF in the other processor (CF6) was provided by either ICB or ISC links (both options were measured). Access to CF1 is through IC links.

During the tests, it was noticed that the type of CF link that was used to access the structures had a significant effect on the structure service times.

1.5 What tests were run?

Several programs were developed specifically to provide the ability to stress the CF structures:

► **The DB2 programs:**

The DB2 structure activity was produced by running a program designed to push large volumes of update requests and lists against a DB2 table. Verbs read from a batch file created the function stream, allowing for repeatable runs that affected a consistent number of rows/pages each time the script was executed.

The test scenarios (JCL and data input) were saved by creating clone files whenever a script change was made, so that individual runs could be recreated at any time in the future.

The data test bed consisted of seven databases that were clones of each other. Each database had a program that would drive activity. The databases were associated with three Buffer Pools. DB1 through DB4 ran against BP20, DB5 and DB6 ran against BP2, and the remaining DB7 had BP1 all to itself.

► **The MQ programs:**

The MQ structure activity was produced by running pairs of programs designed to push large volumes of PUTs and GETs, controlled by external option files.

There were two types of GET programs:

- GRHMQGET: This standard GET program set applicable options provided in the options file.
- GRHMQGW1: This GET program was designed to emulate a WebSphere Application Server cluster. Multiple instances of the GET program were run to simulate a cluster instance of the server. The program had two phases: the first phase listened to the designated queue with a generic browse request (the CR function). When a message was received, the program extracted the CORRELID from the message header and initiated a second request to read the message using the CORRELID.

When a shared queue request is CORRELID specific, the queue must be defined with a CORRELID index (adding content that must be carried in the CF structure).

There were three types of PUT programs:

- GRHMQPUT: This is the base program, which PUTs messages to a single queue.
- GRHMQPT2: This program PUTs messages to two queues in separate structures, causing activity in the Admin structure as well as the application structures.
- GRHMQPW1 - This program PUT messages to a single queue but enforced the use of a CORRELID (which was optional in the other PUT programs). When a shared queue request is CORRELID specific, the queue must be defined with a CORRELID index (adding content that must be carried in the CF structure).

For further details about the WebSphere MQ environment, refer to *WebSphere MQ in a z/OS Parallel Sysplex Environment*, SG24-6864.

1.5.1 Customer requirements

One of the criteria for evaluating the results of the tests was availability, which represented one of the high priority goals for the installation. Performance was also a criterion that was used during the evaluation, while the cost of the solution was not an overriding factor in design of the final solution.

1.6 Terminology

In this section we clarify some of the terminology used throughout the remainder of this document.

ICFs are processor units (PUs) in a general purpose server that are configured to run only CFCC code.

At one time, the term ICF was also used for an Internal CF, meaning a CF that was in the same processor as the operating systems, in contrast to one in a special purpose CF processor such as a 9672-R06, or a 2064-100.

In this document, we use the term *local ICF* to identify a CF that is in the same processor as operating system LPARs that it is connected to. We use the term *external CF* to identify a CF that is only connected to operating systems that do not reside in the same processor.

1.7 Heuristics

One of the elements considered during the tests is the heuristic algorithm that is used by the CF. The objective of the algorithm is to use the most efficient type of CF requests.

In this case, a new algorithm helped determine whether or not it is more efficient to issue a CF request synchronously or asynchronously.

Synchronous requests will always deliver better response times than asynchronous requests. However, if a synchronous request is allowed to run for a long time, it consumes a considerable amount of CPU time. To limit the impact of poor synchronous CF requests, z/OS 1.2 introduced a heuristic algorithm. This algorithm decides the most efficient way to process a request (in terms of CPU utilization) based on the actual response times being achieved, and the speed of the CPC that z/OS is running on. At low response times, the request is generally more efficient if it is left as synchronous. But as response times increase, a threshold is reached where it is more efficient to issue the request asynchronously. For more information about this heuristic algorithm, refer to *WSC Flash 10159*



DB2 structure experiences

In this chapter we describe the results of the various DB2-based tests. Specifically, the following tests were run:

- ▶ Comparing the use of IC, ICB, and ISC links for the DB2 lock structure
- ▶ Determining the impact of system-managed duplexing the DB2 lock structure
- ▶ Establishing the impact of CF CPU utilization on CF request response times
- ▶ Measuring the relative performance of different link types on the DB2 Group Buffer Pool (GBP) structures
- ▶ Investigating the impact of DB2 user-managed duplexing for the GBP structures

2.1 Introduction

While you are reviewing the DB2 test results, it is important to keep several points in mind:

- ▶ The GBPs were defined to contain only changed pages.
- ▶ All the buffer pools used in the tests had a page size of 4 KB.
- ▶ A SPUFI query was used to maintain a data sharing interest during single system runs.
- ▶ One system is always designated as the cast out owner, which can cause activity in a system where no job is currently running.
- ▶ Most of the CPU time that is used for a synchronous request is charged back to the requesting address space. The CPU time that is used by asynchronous requests is distributed to the XCF address space and to the requesting address space. A small amount of the time that is used for asynchronous requests is also distributed to whatever address space happens to be running when the request completion is recognized. As a result, the CPU time for a job can appear to decrease when requests are converted to be asynchronous.
- ▶ RMF™ Monitor III was used to accurately select the intervals used for the utilization and activity reports. It is important to exclude time when the system is idle from the reports because this can affect the reported rates and actual utilizations.
- ▶ The reason for using the IBM Benchmark Center for these tests was to eliminate all workloads other than the one being measured. This would have been impossible in a production environment because all hardware (disks, paths, CPUs, and so on) is shared.

The primary focus of the tests was the response times of different CF configurations, and the impact these response times could have on the CPU utilization of the jobs that were using those CFs. We did not scrutinize how the various configurations affected z/OS CPU utilization, subchannel utilization, or job elapsed times.

The configuration that was used for most of the tests was two CFs, each with two dedicated engines, and two z/OS LPARs, with three dedicated engines each in some tests and two dedicated engines in others. The type of link between the z/OS LPARs and the external CFs was either ISC or ICB, depending on the test. In one of the tests, one engine in each CF was taken offline to observe the service time impacts of higher utilization.

2.2 DB2 lock structures

Using the DB2 workload described in 1.5, “What tests were run?” on page 5, the following tests were performed to understand the workload behavior when changing the configuration. In all of the tests, the lock structure was in CF1, which was in the same CPC as z/OS system P01, and the GBP structures were in CF6, which was in the same CPC as z/OS system P08.

2.2.1 ICB versus ISC for lock structures

The focus of this test was how the type of link used to connect z/OS to the CF affects the response time of the DB2 lock structure. We ran seven update jobs on P01 and seven update jobs on P08 against three GBPs. Data sharing was in place. We used the following structures:

- ▶ DB2G_LOCK1
- ▶ DB2G_SCA
- ▶ DB2G_GBP1
- ▶ DB2G_GBP2
- ▶ DB2G_GBP20

ICB run

The first run was made using ICB links to connect the z/OS images to the external CF. IC links were used to connect to the local ICF. The DB2 structures were spread over the two CFs, with the lock structure being in CF1 and the three GBP structures in CF6. Table 2-1 contains the execution details of the test jobs.

Table 2-1 Summary of job execution using ICB links

Step/Jobname	Link Type for CF1	Elapsed	CPU Time	SRB Time
DBU1P105	IC link	10:49	50.68	0.18
DBU2P105	IC link	10:16	50.49	0.17
DBU3P105	IC link	10:03	50.58	0.18
DBU4P105	IC link	10:05	50.56	0.17
DBU5P105	IC link	7:36	48.85	0.10
DBU6P105	IC link	8:18	49	0.10
DBU7P105	IC link	8:07	48.69	0.10
DBU1P805	ICB link	10:44	53.75	0.18
DBU2P805	ICB link	10:24	53.32	0.15
DBU3P805	ICB link	10:16	52.69	0.17
DBU4P805	ICB link	10:15	53.54	0.17
DBU5P805	ICB link	7:18	52.79	0.06
DBU6P805	ICB link	8:03	51.67	0.06
DBU7P805	ICB link	7:54	51.29	0.06

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
---- Coupling Facility ----- Processor ----- Request  -- Storage --
Name  Type  Model Level  Util% Defined Effect  Rate  Size  Avail
CF1   2084   B16  13  11.1  2  2.0  27592  8126M  6868M
CF6   2084   B16  13  31.9  2  2.0  44135  8126M  6917M

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
CF: CF1      Type ST System  --- Sync --- ----- Async -----
                                     Rate Avg   Rate Avg  Chng Del
Structure Name                               Serv   Serv  %    %
DB2G_LOCK1  LOCK  *ALL  27314  7  139.7  441  0.0  0.0
             P01  13631  6  70.5  439  0.0  0.0
             P08  13684  9  69.3  444  0.0  0.0

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
CF: CF6      Type ST System  --- Sync --- ----- Async -----
                                     Rate Avg   Rate Avg  Chng Del
Structure Name                               Serv   Serv  %    %
DB2G_GBP1   CACHE *ALL  6145  13  213.2  222  0.0  0.0
             P01  3010  14  109.3  224  0.0  0.0
             P08  3136  12  103.9  221  0.0  0.0
DB2G_GBP2   CACHE *ALL  12646  12  430.4  249  0.0  0.0
             P01  6141  14  219.1  252  0.0  0.0
             P08  6505  10  211.3  246  0.0  0.0
DB2G_GBP20  CACHE *ALL  23958  12  840.1  249  0.0  0.0
             P01  11873  13  439.4  250  0.0  0.0
             P08  12085  10  400.7  248  0.0  0.0

```

Figure 2-1 DB2 lock structure using ICB links

Figure 2-1 shows an excerpt from the RMF CF report for the period when the test jobs were running. As you can see, the percentage of asynchronous lock requests is insignificant. The focus, therefore, is the synchronous lock request service times. You can see that the service time for requests coming from the P01 image (where IC links were used) is 6 microseconds, compared to the response time of 9 microseconds for the P08 image (where ICB links were used).

Trying to associate changes in the type of link used to connect to the lock structure with changes in SRB and CPU time is difficult. Each set of jobs used both lock and GBP structures and both IC and ICB links. The P01 jobs used IC links to talk to the lock structure and ICB links to talk to the GBP structure. The P08 jobs used ICB links to talk to the lock structure and IC links to talk to the GBP structure.

In these tests, it was not possible to segregate the lock structure performance from the GBP performance when looking at CPU and SRB time that was used by each job. However, given that the bulk of the requests was synchronous, it is possible to approximate the amount of time that each set of jobs spent communicating with the structures. Using the report in Figure 2-2 on page 12, we can see that the jobs running in P08 spent more time running synchronous requests (multiply number of requests by the service time). The CPU time that is associated with processing synchronous requests is included in the CPU time for a job. Therefore, the P8 jobs see a slightly higher CPU time than the P01 jobs.

If we compare the jobs that ran on P01 (which was connected to the lock structure with IC links) to the jobs that ran on P08, we see no real difference in SRB time. The CPU time used

to process asynchronous requests is reported as SRB time, and because the percentage of requests from each system that are asynchronous is similar, you would not expect to see any significant difference in SRB times.

You might also notice that there is a response time difference between lock and GBP requests, the main reason being that GBP requests carry 4 KB of data, but lock requests contain no data.

ISC run

A second run was made, this time using ISC links to connect the z/OS images to the external CF. IC links were once again used to connect to the local ICF, and the DB2 structures were in the same CFs as the first test. Table 2-2 contains the execution details for the test jobs.

Changes were made to the configuration after the ICB run, meaning that job elapsed times could not reliably be compared from one run to another. These changes did not, however, impact the accuracy of the CPU time that was charged to the jobs.

Table 2-2 Summary of job execution using ISC links

Step/Jobname	Link Type for CF1	Elapsed	CPU Time	SRB Time
DBU1P106	IC link	8:58	54.77	0.78
DBU2P106	IC link	9:02	54.42	0.78
DBU3P106	IC link	9:21	54.45	0.77
DBU4P106	IC link	9:21	54.54	0.78
DBU5P106	IC link	8:39	53.27	0.72
DBU6P106	IC link	9:11	53.64	0.73
DBU7P106	IC link	9:09	53.31	0.72
DBU1P806	ISC link	8:41	1:12.18	0.11
DBU2P806	ISC link	8:46	1:12.96	0.11
DBU3P806	ISC link	9:05	1:12.00	0.10
DBU4P806	ISC link	9:06	1:12.82	0.10
DBU5P806	ISC link	8:24	1:12.34	0.05
DBU6P806	ISC link	8:56	1:12.32	0.06
DBU7P806	ISC link	8:54	1:12.38	0.06

```

Command ==>
RMF V1R2  CF Overview  - PELPLEXA
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.31.00  Range: 540  Sec
---- Coupling Facility ----- Processor ----- Request  -- Storage --
Name  Type  Model Level  Util% Defined Effect  Rate  Size  Avail
CF1   2084   B16  13   15.0   2    2.0   30565  8126M  6868M
CF6   2084   B16  13   43.6   2    2.0   44865  8126M  6917M

Command ==>
RMF V1R2  CF Activity  - PELPLEXA
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.31.00  Range: 540  Sec
CF: CF1      Type ST System  --- Sync ---  ----- Async -----
                                     Rate  Avg  Rate  Avg  Chng  Del
                                     Serv  Serv  %    %
Structure Name
DB2G_LOCK1  LOCK  *ALL  29600  15  156.6  421  0.0  0.0
           P01  14456  6   76.6  349  0.0  0.0
           P08  15144  23  80.0  490  0.0  0.0

Command ==>
RMF V1R2  CF Activity  - PELPLEXA
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.31.00  Range: 540  Sec
CF: CF6      Type ST System  --- Sync ---  ----- Async -----
                                     Rate  Avg  Rate  Avg  Chng  Del
                                     Serv  Serv  %    %
Structure Name
DB2G_GBPO   CACHE *ALL  0.1  29  <0.1  200  0.0  0.0
           P01  0.1  30  <0.1  200  0.0  0.0
           P08  <0.1  10  0.0  0  0.0  0.0
DB2G_GBP1   CACHE *ALL  3837  20  2678  100  0.0  0.0
           P01  496.1  43  2556  95  0.0  0.0
           P08  3341  17  121.7  219  0.0  0.0
DB2G_GBP2   CACHE *ALL  7897  17  5571  101  0.0  0.0
           P01  1076  43  5320  95  0.0  0.0
           P08  6821  13  250.9  242  0.0  0.0
DB2G_GBP20  CACHE *ALL  14804  18  10480  100  0.0  0.0
           P01  1951  43  10010  94  0.0  0.0
           P08  12853  15  470.4  232  0.0  0.0

```

Figure 2-2 DB2 lock structure using ISC links

As you can see in Figure 2-2, even in this run, the percentage of asynchronous requests to the lock structure are insignificant. You can see that the service time for the lock requests coming from the P01 image (where IC links were used) is still 6 microseconds as in the previous run; however, the response time for the P08 image (where ISC links were used) has increased to 23 microseconds.

The difference between the service times in the two different runs (23 – 9 = 14 microseconds) is a result of the additional time that it takes to convert the CF signals for the fiber ISC links.

You might also notice that the response time for synchronous requests from P01 to the GBP structures increased significantly, with the bulk of the GBP requests from P01 now being asynchronous. While we are concentrating on the lock structure in this section, it is important to remember that the changed GBP response time for P01 jobs has an impact on CPU time for those jobs.

Table 2-3 on page 13 shows a comparison of the CPU time for the jobs when ICB links are used with that for the jobs when ISC links are used. For example, the CPU time for job DBU1P1 was 8% higher when ISC links were used to communicate with the GBP structure. The CPU time for the equivalent job on P8, DBU1P8, increased by 34%. From Figure 2-2, you can infer the reason for this difference. In that illustration, the bulk of GBP requests from P1 are now asynchronous, while nearly all the CF requests from P8 are still synchronous. Previously, we said that the CPU time that is consumed by synchronous requests is reported as

CPU time for the requesting job, while some of the CPU that is consumed by asynchronous requests is reported as SRB time, with the remainder being charged to other address spaces. So, in Table 2-3 we are seeing the effect of this. The CPU time for the P8 jobs appears to have increased significantly, but the CPU time for the P1 jobs has not increased as much; most of those requests are now asynchronous. The CPU time that is associated with those requests is being charged to other address spaces, with a little of that CPU showing up as higher SRB time for the P1 jobs.

Table 2-3 Comparison of ISC and ICB runs

Jobname	CPU time change for ISC run
DBU1P106	+8%
DBU2P106	+8%
DBU3P106	+7%
DBU4P106	+8%
DBU5P106	+9%
DBU6P106	+10%
DBU7P106	+10%
DBU1P806	+34%
DBU2P806	+37%
DBU3P806	+37%
DBU4P806	+36%
DBU5P806	+37%
DBU6P806	+40%
DBU7P806	+41%

Summary

The most obvious result of these two tests is the impact of link type on request response times. There are two attributes of CF links that affect the response time they can deliver:

- ▶ The bandwidth provided by the link: Links with more bandwidth, such as IC or ICB, make a bigger difference in response time when the request involves data transfer, such as to a cache or list structure.
- ▶ The actual process of converting the request into an electrical or optical signal to be sent to the CF: No conversion, as in the case of IC links, is obviously the fastest. After that, conversion to an electrical signal, for any of the ICB links, is the next fastest. Conversion to optical signals are almost always the slowest.

Because we are focusing on the lock structure, and lock requests contain no data, the link bandwidth does not have much impact. However, you can see in this configuration that IC links provided about a 3 microsecond advantage over ICB, and ICB had a 14 microsecond advantage over ISC—this is caused mainly by the physical process of converting the electrical signals to light signals for the fibre optic ISC links.

While a difference of 3 or 14 microseconds are small enough to appear nearly irrelevant, remember that the requesting z/OS CP spins for the duration of a synchronous request. So, the shorter the response time, the lower is the cost in terms of z/OS CPU utilization.

Based on these two tests, the preferred link type is ICB over ISC wherever possible (ICB links support a maximum distance of 7 meters between CPCs).

Note: You might have noticed in Figure 2-2 on page 12 that the CPU utilization of both CFs, but especially CF6, has increased when the ICB links were changed to ISC. This effect is discussed in 2.3.1, “ICB versus ISC for GBP structures” on page 22

2.2.2 Lock structure duplexing

If the customer had proceeded with the original plan to replace external CFs with CFs that were located in z/OS CPCs, it would have been necessary to use system-managed duplexing with any structure that had a failure-isolation requirement. To evaluate the impact of using this function, we ran a set of tests that compared response times for the DB2 lock structure with and without system-managed duplexing. ICB connections were used for the z/OS-to-external CF and the CF-to-CF links in these tests. Duplexing for the DB2 GBP structures was tested in another run; see 2.3.2, “User-managed duplexing” on page 26 for details.

For the tests, we ran seven update jobs on P01 and seven update jobs on P08. The following DB2 structures were in use:

- ▶ DB2G_LOCK1 (duplexed using SM Duplexing)
- ▶ DB2G_SCA
- ▶ DB2G_GBP1
- ▶ DB2G_GBP2
- ▶ DB2G_GBP20

The same jobs that were used previously for the ISC versus ICB test were used for this test. Therefore, the base measurements are the results of the ICB run, as shown in Table 2-4 and Figure 2-3 on page 15.

Table 2-4 Summary of job execution with simplex lock structure

Step/Jobname	Link Type for CF1	Elapsed	CPU Time	SRB Time
DBU1P105	IC link	10:49	50.68	0.18
DBU2P105	IC link	10:16	50.49	0.17
DBU3P105	IC link	10:03	50.58	0.18
DBU4P105	IC link	10:05	50.56	0.17
DBU5P105	IC link	7:36	48.85	0.10
DBU6P105	IC link	8:18	49	0.10
DBU7P105	IC link	8:07	48.69	0.10
DBU1P805	ICB link	10:44	53.75	0.18
DBU2P805	ICB link	10:24	53.32	0.15
DBU3P805	ICB link	10:16	52.69	0.17
DBU4P805	ICB link	10:15	53.54	0.17
DBU5P805	ICB link	7:18	52.79	0.06
DBU6P805	ICB link	8:03	51.67	0.06
DBU7P805	ICB link	7:54	51.29	0.06


```

RMF V1R2 CF Overview - PELPLEXA
Command ==>
Samples: 540 Systems: 2 Date: 06/08/04 Time: 14.02.00 Range: 540 Sec
---- Coupling Facility ----- Processor ----- Request -- Storage --
Name Type Model Level Util% Defined Effect Rate Size Avail
CF1 2084 B16 13 11.1 2 2.0 27592 8126M 6868M
CF6 2084 B16 13 31.9 2 2.0 44135 8126M 6917M

RMF V1R2 CF Activity - PELPLEXA
Command ==>
Samples: 540 Systems: 2 Date: 06/08/04 Time: 14.02.00 Range: 540 Sec
CF: CF1
Type ST System --- Sync --- Async -----
Rate Avg Rate Avg Chng Del
Serv Serv % %
Structure Name
DB2G_LOCK1 LOCK *ALL 27314 7 139.7 441 0.0 0.0
P01 13631 6 70.5 439 0.0 0.0
P08 13684 9 69.3 444 0.0 0.0

RMF V1R2 CF Activity - PELPLEXA
Command ==>
Samples: 540 Systems: 2 Date: 06/08/04 Time: 14.02.00 Range: 540 Sec
CF: CF6
Type ST System --- Sync --- Async -----
Rate Avg Rate Avg Chng Del
Serv Serv % %
Structure Name
DB2G_GBP1 CACHE *ALL 6145 13 213.2 222 0.0 0.0
P01 3010 14 109.3 224 0.0 0.0
P08 3136 12 103.9 221 0.0 0.0
DB2G_GBP2 CACHE *ALL 12646 12 430.4 249 0.0 0.0
P01 6141 14 219.1 252 0.0 0.0
P08 6505 10 211.3 246 0.0 0.0
DB2G_GBP20 CACHE *ALL 23958 12 840.1 249 0.0 0.0
P01 11873 13 439.4 250 0.0 0.0
P08 12085 10 400.7 248 0.0 0.0

```

Figure 2-3 DB2 simplex lock structure performance

The results of the run that used system-managed duplexing with the lock structure are shown in Table 2-5 and Figure 2-4 on page 16. Figure 2-4 on page 16 summarizes the CF service times. Changes were made to the configuration after the ICB run, meaning that job elapsed times could not reliably be compared from one run to another. These changes did not, however, impact the accuracy of the CPU time charged to the jobs.

Table 2-5 Summary of job execution with duplexed lock structure

Step/Jobname	Link Type for CF1	Elapsed	CPU Time	SRB Time
DBU1P114	IC link	8:34	2:03.36	0.24
DBU1P814	ICB link	9:10	2:10.00	0.16
DBU2P114	IC link	9:59	2:04.74	0.25
DBU2P814	ICB link	9:35	2:05.76	0.09
DBU3P114	IC link	9:48	2:04.58	0.25
DBU3P814	ICB link	8:26	2:08.29	0.18
DBU4P114	IC link	9:48	2:04.76	0.25
DBU4P814	ICB link	9:31	2:07.30	0.20
DBU5P114	IC link	8:54	2:01.89	0.19

Step/Jobname	Link Type for CF1	Elapsed	CPU Time	SRB Time
DBU5P814	ICB link	8:32	2:07.24	0.09
DBU6P114	IC link	9:49	2:04.55	0.18
DBU6P814	ICB link	9:27	2:06.92	0.08
DBU7P114	IC link	8:53	2:04.04	0.18
DBU7P814	ICB link	8:32	2:05.06	0.07

You can see in Figure 2-4 that the response times for the lock structure have increased from the 6 and 9 microseconds that were achieved in the simplex run to 52 and 51 microseconds respectively; this is a result of the inherent additional CPU usage of system-managed duplexed requests and the much higher CF CPU utilization. The percentage of requests converted to asynchronous increased from .5% for the simplex structure to 4.5% for the duplex structure. This equates to a higher overall average lock response time. If a sufficiently large number of lock requests were to become asynchronous, this could result in an increase in lock contention.

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 600  Systems: 2  Date: 06/10/04  Time: 13.00.00  Range: 600  Sec
---- Coupling Facility ----
Name  Type  Model Level  Util% Defined Effect  Request  -- Storage --
CF1   2084   B16  13    52.3    2    2.0    28150   8126M  4123M
CF6   2084   B16  13    75.1    2    2.0    69470   8126M  6059M

RMF V1R2  CF Activity  - PELPLEXA Command ==>
Samples: 600  Systems: 2  Date: 06/10/04  Time: 13.00.00  Range: 600  Sec
CF: CF1      Type ST System  --- Sync ---  ----- Async -----
Structure Name  Rate  Avg  Rate  Avg  Chng  Del
                Serv  %
DB2G_LOCK1 LOCK *ALL 25989 51 1140 470 0.0 0.0
                P01 12219 52 900.7 327 0.0 0.0
                P08 13771 51 239.4 1010 0.0 0.0

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 600  Systems: 2  Date: 06/10/04  Time: 13.00.00  Range: 600  Sec
CF: CF6      Type ST System  --- Sync ---  ----- Async -----
Structure Name  Rate  Avg  Rate  Avg  Chng  Del
                Serv  %
DB2G_GBP1 CACHE *ALL 5816 22 340.1 178 0.0 0.0
                P01 2775 24 214.7 167 0.0 0.0
                P08 3041 21 125.4 196 0.0 0.0
DB2G_GBP2 CACHE *ALL 11929 21 704.1 194 0.0 0.0
                P01 5676 24 450.4 181 0.0 0.0
                P08 6253 18 253.7 218 0.0 0.0
DB2G_GBP20 CACHE *ALL 21695 22 1378 189 0.0 0.0
                P01 10725 25 905.1 179 0.0 0.0
                P08 10970 19 473.2 208 0.0 0.0
DB2G_LOCK1 LOCK *ALL 25955 57 1140 481 0.0 0.0
                P01 12190 60 900.8 339 0.0 0.0
                P08 13765 55 239.4 1015 0.0 0.0

```

Figure 2-4 DB2 lock structure with duplexing

The CPU utilization of CF1 increased from 11.1% to 55% (an increase of 44%). Because all updates for a system-managed duplex structure are also driven to the CF with the duplex copy of the structure, the CPU utilization of CF6 also increased, from 31.9% to 79.5% (an

increase of 46.6%). This is a little higher than the expectation that a system-managed duplexed CF request consumes between four and five times the CF CPU of a simplex request. A possible explanation of this disparity is that the tests used CFlevel 13. CFlevel 14 featured an enhancement that was specifically intended to decrease CF CPU utilization when using system-managed duplexing. At the time of writing, a suite of tests to measure the difference between simplex and duplex requests when using CFlevel 14 is being planned. This Redpaper will be updated to reflect the results of this test when they are available.

Because of the increase in response time for the synchronous requests, the CPU utilization for the jobs has increased by two and a half times. The general guideline is that duplexing a lock structure quadruples the z/OS CPU time for synchronous duplexed requests. Given that not all the job CPU time was spent processing CF requests, the results in Table 2-6 are roughly in line with expectations. For more information about the considerations for, and performance impacts of, system-managed structure duplexing, refer to “*System Managed CF Structure Duplexing*”, available on the Web at:

<http://www.ibm.com/servers/eserver/zseries/library/techpapers/pdf/gm130103.pdf>

Table 2-6 Comparison of simplex and duplex runs

Jobname	CPU time change for duplex run
DBU1P106	+141%
DBU2P106	+148%
DBU3P106	+145%
DBU4P106	+145%
DBU5P106	+149%
DBU6P106	+155%
DBU7P106	+153%
DBU1P806	+141%
DBU2P806	+138%
DBU3P806	+142%
DBU4P806	+135%
DBU5P806	+140%
DBU6P806	+144%
DBU7P806	+145%

Interestingly, despite the higher response times and the higher CF CPU utilization, the number of DB2 requests processed was actually nearly identical to the simplex case. As you would expect, the increase in the request rate to CF6 (29271 per second) is almost entirely due to the number of requests per second sent to the secondary lock structure (28902 per second).

Summary

Depending on your the workload and service level requirements of your installation, the performance impact of duplexing the DB2 lock structure might not be acceptable. If this is the case, the lock structure should reside in an external CF. The external CFs should contain all the structures that have a failure isolation requirement, like the DB2 lock structure.

Because of the high lock rate of the customer applications, the customer determined that the impact on z/OS CPU utilization that would result from duplexing the DB2 lock was unacceptable. Based on this test, the customer decided that the more appropriate strategy for them was to continue with their use of external CFs for structures with a failure-isolation requirement. However, the use of local ICFs could still be useful for isolating external CF traffic from other structures, for example.

If an installation uses an external CF to house the (simplex) lock structure, the other CF in the configuration can be either external or internal. If the lock structure fails over from the external CF to an internal CF, the installation has a loss of failure-isolation/robustness during that failover period, but that might be acceptable. If there were multiple external CFs, failure-isolation/robustness could be maintained even in the failover scenario.

2.2.3 CF processor impact

The next test was to measure the impact of CF CPU utilization on CF request response times. For the test, we ran seven update jobs on P01 and seven update jobs on P08 against three GBPs using ICBs and ICs as links. Data sharing was in place. We used the following structures:

- ▶ DB2G_LOCK1
- ▶ DB2G_SCA
- ▶ DB2G_GBP1
- ▶ DB2G_GBP2
- ▶ DB2G_GBP20

The ICB run documented in 2.2.1, “ICB versus ISC for lock structures” on page 8 was used as the base measurement for this test. The results of that run are repeated here in Table 2-7 and Figure 2-5 on page 19.

Table 2-7 Summary of job execution using CFs with ICB links and two engines

Step/Jobname	Elapsed	CPU Time	SRB Time
DBU1P105	10:49	50.68	0.18
DBU2P105	10:16	50.49	0.17
DBU3P105	10:03	50.58	0.18
DBU4P105	10:05	50.56	0.17
DBU5P105	7:36	48.85	0.10
DBU6P105	8:18	49	0.10
DBU7P105	8:07	48.69	0.10
DBU1P805	10:44	53.75	0.18
DBU2P805	10:24	53.32	0.15
DBU3P805	10:16	52.69	0.17
DBU4P805	10:15	53.54	0.17
DBU5P805	7:18	52.79	0.06
DBU6P805	8:03	51.67	0.06
DBU7P805	7:54	51.29	0.06

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
---- Coupling Facility ----
Name  Type  Model Level  Util% Defined Effect  Request  -- Storage --
CF1   2084  B16  13  11.1  2  2.0  27592  8126M  6868M
CF6   2084  B16  13  31.9  2  2.0  44135  8126M  6917M

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
CF: CF1
Type ST System  --- Sync ---  ----- Async -----
Rate Avg Rate Avg Chng Del
Serv Serv % %
Structure Name
DB2G_LOCK1  LOCK  *ALL  27314  7  139.7  441  0.0  0.0
                P01  13631  6  70.5  439  0.0  0.0
                P08  13684  9  69.3  444  0.0  0.0

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
CF: CF6
Type ST System  --- Sync ---  ----- Async -----
Rate Avg Rate Avg Chng Del
Serv Serv % %
Structure Name
DB2G_GBP1  CACHE  *ALL  6145  13  213.2  222  0.0  0.0
                P01  3010  14  109.3  224  0.0  0.0
                P08  3136  12  103.9  221  0.0  0.0
DB2G_GBP2  CACHE  *ALL  12646  12  430.4  249  0.0  0.0
                P01  6141  14  219.1  252  0.0  0.0
                P08  6505  10  211.3  246  0.0  0.0
DB2G_GBP20  CACHE  *ALL  23958  12  840.1  249  0.0  0.0
                P01  11873  13  439.4  250  0.0  0.0
                P08  12085  10  400.7  248  0.0  0.0

```

Figure 2-5 CF report for 2-engine CF configuration

As you can see, the percentage of asynchronous lock requests is insignificant, the CF processor utilization is low (11% and 31.9%), and the service time for the lock requests varies from 6 microseconds to 9 microseconds.

The second run was made after configuring one engine offline in each CF, which increased the CF processor utilization, to determine the impact, if any, on the CF service times. The results are summarized in Table 2-8 and Figure 2-6 on page 20. Changes were made to the configuration after the ICB run, meaning that job elapsed times could not reliably be compared from one run to another. These changes did not, however, impact the accuracy of the CPU time that was charged to the jobs.

Table 2-8 Summary of the Jobs run with a CF processor configured offline

Step/Jobname	Elapsed	CPU Time	SRB Time
DBU1P116	8:13	58.23	0.37
DBU2P116	8:54	57.40	0.31
DBU3P116	8:58	57.93	0.35
DBU4P116	9:02	57.50	0.33
DBU5P116	7:52	56.36	0.29
DBU6P116	8:27	56.18	0.28

Step/Jobname	Elapsed	CPU Time	SRB Time
DBU7P116	8:14	56.22	0.28
DBU1P816	8:28	N/A ¹	0.28
DBU2P816	8:41	N/A ¹	0.23
DBU3P816	8:54	N/A ¹	0.23
DBU4P816	8:58	N/A ¹	0.22
DBU5P816	7:43	58.89	0.16
DBU6P816	8:17	59.37	0.15
DBU7P816	8:04	59.28	0.15

¹The CPU time information for these jobs was mislaid.

```

RMF V1R2  CF Overview      - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/11/04  Time: 12.10.00  Range: 540  Sec
---- Coupling Facility ----- Processor ----- Request  -- Storage --
Name      Type  Model Level  Util% Defined Effect  Rate      Size  Avail
CF1       2084   B16   13    22.7  1  1.0    30969   8126M  4123M
CF6       2084   B16   13    61.4  1  1.0    47659   8126M  3254M
  
```

```

RMF V1R2  CF Activity      - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/11/04  Time: 12.10.00  Range: 540  Sec
CF: CF1      Type  ST System
Structure Name
DB2G_LOCK1   LOCK  *ALL
              P01
              P08
  
```

Structure Name	Type	ST	System	--- Sync ---		----- Async -----			
				Rate	Avg Serv	Rate	Avg Serv	Chng %	Del %
DB2G_LOCK1	LOCK		*ALL	29872	10	152.7	369	0.0	0.0
			P01	14809	9	76.7	360	0.0	0.0
			P08	15063	12	76.0	379	0.0	0.0

```

RMF V1R2  CF Activity      - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/11/04  Time: 12.10.00  Range: 540  Sec
CF: CF6      Type  ST System
Structure Name
DB2G_GBPO    CACHE *ALL
              P01
              P08
DB2G_GBP1    CACHE *ALL
              P01
              P08
DB2G_GBP2    CACHE *ALL
              P01
              P08
DB2G_GBP20   CACHE *ALL
              P01
              P08
  
```

Structure Name	Type	ST	System	--- Sync ---		----- Async -----			
				Rate	Avg Serv	Rate	Avg Serv	Chng %	Del %
DB2G_GBPO	CACHE		*ALL	<0.1	22	0.1	146	0.0	0.0
			P01	<0.1	23	0.1	149	0.0	0.0
			P08	<0.1	11	<0.1	37	0.0	0.0
DB2G_GBP1	CACHE		*ALL	6035	27	838.3	148	0.0	0.0
			P01	2909	29	495.1	144	0.0	0.0
			P08	3127	26	343.2	153	0.0	0.0
DB2G_GBP2	CACHE		*ALL	12394	27	1715	152	0.0	0.0
			P01	5948	29	1017	151	0.0	0.0
			P08	6445	24	697.4	153	0.0	0.0
DB2G_GBP20	CACHE		*ALL	22859	28	3309	152	0.0	0.0
			P01	11747	30	2020	152	0.0	0.0
			P08	11111	26	1289	153	0.0	0.0

Figure 2-6 DB2 structure service time with a CF processor offline

Note that the number of requests to each CF is marginally higher than the two-engine scenario and that the CF CPU utilization is approximately doubled; this exactly what you would expect considering that half the capacity was removed.

Specifically, we can observe an increase in the percentage of requests that are asynchronous and in the service times for all requests. While the CF CPU utilization of CF1 doubled, the response times for CF1 (the less busy CF) increased by about 50%. The response time impact was larger for the busier CF (CF6), increasing by about 100%. This behavior is expected; as CPU utilization increases, queuing increases, meaning that response times go up exponentially as you get closer to 100% utilization.

An interesting side effect of the higher CF CPU utilization is higher CPU utilization by the jobs that are using those CFs. Because a large portion of CF requests are still synchronous, the longer response time for those requests is reflected in CPU times that are roughly 15% higher than the base case.

It is also worth mentioning that CFs with a small number of engines are more sensitive to high utilization than those with a higher number of engines. Figure 2-7 is a chart that shows the theoretical response time at various utilizations for varying numbers of engines and the results of some actual measurements. Note that there is a fairly close correspondence between the two. You can see that, in the CF with just one engine, there was a steady increase in service times from relatively low utilizations. The CF with two engines was able to deliver acceptable response times up to significantly higher utilizations, and CFs with four or more engines delivered acceptable response times that were nearly flat up to utilizations above 60%. Looking back on the test with hindsight, it probably would have been more representative to keep the number of engines the same, and double the workload. By taking one engine offline, the results reflect the higher utilization *and* the effect of using one engine rather than two.

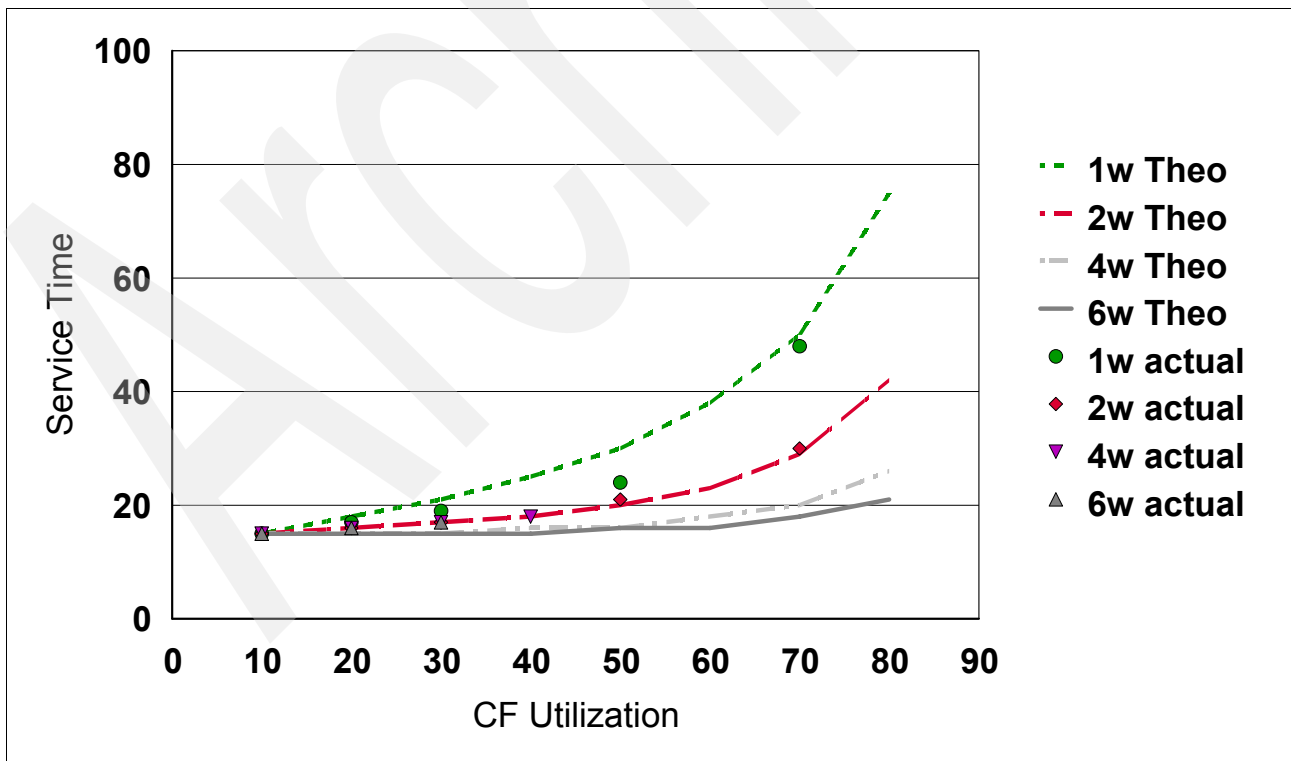


Figure 2-7 Response times versus utilization for varying numbers of CF engines

Summary

The primary intent of this run was to understand the impact of CF CPU utilization on service times. There are three considerations for acceptable CF CPU utilization:

- ▶ Utilization should not reach a level where response times are impacted to an unacceptable degree.
- ▶ Increased CF CPU utilization results in increased CF response times, which in turn results in increased CPU utilization in the address spaces that are using those structures.
- ▶ There should be enough spare capacity in each CF to handle the load from the other CF in case of a planned or unplanned CF outage. Normally, with simplex structures, this rule for failover generally translates into 50% utilization or lower. However, with duplexed structures, you might still have sufficient capacity for failover situations even if both CFs are over 50% utilized because the utilization of the duplexed structure decreases when it reverts to simplex mode.

This test showed that even at utilizations in the 60% range, the delivered response times were still very good. However, in the test here, if one CF had failed, the remaining CF would have been running over 85% busy. This is probably a bit higher than you would wish to run in a production environment.

There are two important messages here. First, the utilization of each CF should be monitored to ensure that it does not approach the range where service times become unacceptable. Perhaps more importantly, the sum utilization of the two CFs should be monitored to ensure it does not exceed an installation-specified threshold, probably in the 75% to 85% range.

2.3 DB2 GBP structures

The following tests were conducted to understand how the GBP performance varies depending on the CF configuration.

2.3.1 ICB versus ISC for GBP structures

The objective of this test was to determine the performance impact of different types of CF links on GBP performance. We ran seven update jobs on P01 and seven update jobs on P08 against three GBPs. Data sharing was in place. We used the following structures:

- ▶ DB2G_LOCK1
- ▶ DB2G_SCA
- ▶ DB2G_GBP1
- ▶ DB2G_GBP2
- ▶ DB2G_GBP20

The first run used ICB links to connect the z/OS images to the CF with the GBP structures. Table 2-9 and Figure 2-8 on page 23 summarize the results of the base run (in fact, these are the same runs that were discussed in 2.2.1, “ICB versus ISC for lock structures” on page 8).

Table 2-9 Summary of job execution using ICB links

Step/Jobname	Link Type for CF1	Elapsed	CPU Time	SRB Time
DBU1P105	IC link	10:49	50.68	0.18
DBU2P105	IC link	10:16	50.49	0.17
DBU3P105	IC link	10:03	50.58	0.18

Step/Jobname	Link Type for CF1	Elapsed	CPU Time	SRB Time
DBU4P105	IC link	10:05	50.56	0.17
DBU5P105	IC link	7:36	48.85	0.10
DBU6P105	IC link	8:18	49	0.10
DBU7P105	IC link	8:07	48.69	0.10
DBU1P805	ICB link	10:44	53.75	0.18
DBU2P805	ICB link	10:24	53.32	0.15
DBU3P805	ICB link	10:16	52.69	0.17
DBU4P805	ICB link	10:15	53.54	0.17
DBU5P805	ICB link	7:18	52.79	0.06
DBU6P805	ICB link	8:03	51.67	0.06
DBU7P805	ICB link	7:54	51.29	0.06

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
---- Coupling Facility ----- Processor Request  -- Storage --
Name      Type  Model Level  Util% Defined Effect  Rate  Size Avail
CF1       2084  B16  13  11.1  2  2.0  27592  8126M  6868M
CF6       2084  B16  13  31.9  2  2.0  44135  8126M  6917M

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
CF: CF1      Type ST System  --- Sync ---  --- Async ---
              Rate Avg  Rate Avg Chng Del
              Serv  Serv %  %
Structure Name
DB2G_LOCK1   LOCK  *ALL  27314  7  139.7  441  0.0  0.0
              P01  13631  6  70.5  439  0.0  0.0
              P08  13684  9  69.3  444  0.0  0.0

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.02.00  Range: 540  Sec
CF: CF6      Type ST System  --- Sync ---  --- Async ---
              Rate Avg  Rate Avg Chng Del
              Serv  Serv %  %
Structure Name
DB2G_GBP1   CACHE *ALL  6145  13  213.2  222  0.0  0.0
              P01  3010  14  109.3  224  0.0  0.0
              P08  3136  12  103.9  221  0.0  0.0
DB2G_GBP2   CACHE *ALL  12646  12  430.4  249  0.0  0.0
              P01  6141  14  219.1  252  0.0  0.0
              P08  6505  10  211.3  246  0.0  0.0
DB2G_GBP20  CACHE *ALL  23958  12  840.1  249  0.0  0.0
              P01  11873  13  439.4  250  0.0  0.0
              P08  12085  10  400.7  248  0.0  0.0

```

Figure 2-8 DB2 GBP structures using ICB links

Note that the overall percentage of asynchronous GBP requests is insignificant (96.6% of GBP requests were synchronous). Even for system P01, which used the ICB rather than IC links, the percent of asynchronous requests is only 3.5%.

The response time for the synchronous GBP requests varied between 10 and 14 microseconds. The response time for the asynchronous GBP requests varies between 220 and 250 microseconds. The CPU utilization of the CF with the GBP structures was 31.9%.

The second run used ISC links instead of the ICB connections. The same jobs as those used in the ICB test were run. Table 2-10 provides a summary of the job execution details, and Figure 2-9 on page 25 contains the information about the CF performance during the run. Changes were made to the configuration during the test, meaning that job elapsed times could not reliably be compared from one run to another. These changes did not, however, impact the accuracy of the CPU time that was charged to the jobs.

Table 2-10 Summary of job execution using ISC links

Step/Jobname	Link Type for CF1	Elapsed	CPU Time	SRB Time
DBU1P106	IC link	8:58	54.77	0.78
DBU2P106	IC link	9:02	54.42	0.78
DBU3P106	IC link	9:21	54.45	0.77
DBU4P106	IC link	9:21	54.54	0.78
DBU5P106	IC link	8:39	53.27	0.72
DBU6P106	IC link	9:11	53.64	0.73
DBU7P106	IC link	9:09	53.31	0.72
DBU1P806	ICB link	8:41	1:12.18	0.11
DBU2P806	ICB link	8:46	1:12.96	0.11
DBU3P806	ICB link	9:05	1:12.00	0.10
DBU4P806	ICB link	9:06	1:12.82	0.10
DBU5P806	ICB link	8:24	1:12.34	0.05
DBU6P806	ICB link	8:56	1:12.32	0.06
DBU7P806	ICB link	8:54	1:12.38	0.06

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.31.00  Range: 540  Sec
---- Coupling Facility ----  ---- Processor ----  Request  -- Storage --
Name  Type  Model Level  Util% Defined Effect  Rate  Size  Avail
CF1   2084   B16  13  15.0  2  2.0  30565  8126M  6868M
CF6   2084   B16  13  43.6  2  2.0  44865  8126M  6917M

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.31.00  Range: 540  Sec
CF: CF1      Type  ST System  --- Sync ---  ----- Async -----
                                Rate  Avg      Rate  Avg  Chng  Del
                                Serv  %       Serv  %   %    %
Structure Name
DB2G_LOCK1  LOCK  *ALL  29600  15  156.6  421  0.0  0.0
                                P01  14456  6   76.6  349  0.0  0.0
                                P08  15144  23  80.0  490  0.0  0.0

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 540  Systems: 2  Date: 06/08/04  Time: 14.31.00  Range: 540  Sec
CF: CF6      Type  ST System  --- Sync ---  ----- Async -----
                                Rate  Avg      Rate  Avg  Chng  Del
                                Serv  %       Serv  %   %    %
Structure Name
DB2G_GBP1   CACHE *ALL  3837  20  2678  100  0.0  0.0
                                P01  496.1  43  2556  95  0.0  0.0
                                P08  3341  17  121.7  219  0.0  0.0
DB2G_GBP2   CACHE *ALL  7897  17  5571  101  0.0  0.0
                                P01  1076  43  5320  95  0.0  0.0
                                P08  6821  13  250.9  242  0.0  0.0
DB2G_GBP20  CACHE *ALL  14804  18  10480  100  0.0  0.0
                                P01  1951  43  10010  94  0.0  0.0
                                P08  12853  15  470.4  232  0.0  0.0

```

Figure 2-9 DB2 GBP structures using ISC links

There are a number of interesting results in the report:

- ▶ The percentage of asynchronous requests has increased from 3.5% for system P01 when using ICB links to 84.2% when using ISC.

This shift from synchronous to asynchronous requests is a result of the heuristic asynchronous algorithm that was introduced in z/OS 1.2. This converts CF requests from synchronous to asynchronous when the response times exceed a CPC-specific threshold.

- ▶ The response time for the synchronous requests coming from P08 is stable. The response time for the synchronous requests coming from P01 (which is now using ISC rather than ICB links) has increased from 14 microseconds to 43 microseconds. Note that this increase is larger than the increase in lock response time (which increased from 9 to 23 microseconds).

The larger impact on synchronous response time for GBP requests compared to lock requests is due to the fact that GBP requests are larger (they include data; the lock requests do not) and therefore they are more affected by the lower bandwidth of the ISC links.

- ▶ The asynchronous response time for P08 is about the same, but the asynchronous response time from system P01 has dropped from over 200 microseconds down to 95 microseconds.

Because of their asynchronous nature, there is normally a delay between the time that the CF sends a response to an asynchronous request and when XES processes that request. After XES processes an asynchronous request, it checks to see if any more requests have

completed before it releases the CP back to the dispatcher. As the rate of asynchronous requests increases, the likelihood of XES successfully finding another completed request also increases, resulting in the reduced average asynchronous response time that we witnessed for system P01.

- ▶ The CPU utilization of CF6 has increased from 31.9% using ICB links to 43.6% using ISC links, despite the fact that the total number of requests to that CF are nearly the same. Note that the two CFs were running CFLevel 13. At that CF level, time that the CF spent waiting for a data transfer to complete was reported in RMF as busy time. Because the ISC links are slower than ICB links, both CFs spent more time waiting for the transfers to complete in this test. Also, because the lower bandwidth of the ISC links especially impacts data transfers such as those to the DB2 GBP structures, the utilization of CF6 was impacted more than CF1.

Unfortunately, we did not have the opportunity to re-run this test with CFLevel 14; however, we believe that the higher CF Level would have eliminated the difference in CF CPU utilization that we saw in this test.

Even for customers with z900 technology, where CFLevel 13 is the highest available CF level, we believe that most people would not see as significant a difference between ICB and ISC links. Most CFs contain a mixture of lock, list, and cache structures, so the effect seen here is much more dramatic than normal.

Summary

These tests reinforce the finding of the lock structure ICB versus ISC run—specifically, that ICB links deliver significantly better response time for the same CF load and should be used wherever possible. The higher bandwidth of ICB benefits GBP structures more than it does the lock structure. Wherever possible, structures containing data (list and cache structures, for example) should be placed in CFs that are accessed using IC or ICB links.

As CPC speeds increase with each new generation, there will be a tendency to have more and more requests run asynchronously. If your applications require very short response times, the use of ICB links improves the likelihood of more requests remaining synchronous.

2.3.2 User-managed duplexing

The final DB2-related test was to measure the impact of DB2 user-managed GBP structure duplexing. User-managed duplexing is very different from system-managed duplexing and has a very different performance profile. Duplexing the GBP structures is highly recommended as a way to avoid GRECP processing in DB2 if the CF with a simplex GBP structure were to fail.

For this test, we ran a single DB2 job against a duplexed GBP structure. The CF with the structure was connected to the z/OS with the DB2 job through ICB links. Data sharing was in place and inter-DB2 read/write interest was maintained by running a SPUFI job in the other system. A similar test was run using the other system (P08) with similar results. The following DB2 structures were used:

- ▶ DB2G_LOCK1
- ▶ DB2G_SCA
- ▶ DB2G_GBP20 (Duplexed using User-Managed duplexing)

First, we ran a job to obtain a baseline measurement. The statistics for this job are shown in Table 2-11 on page 27. The CF performance information is shown in Figure 2-10 on page 27.

Table 2-11 Job statistics for DB2 GBP base measurement job

Step/Jobname	Link Type for CF6	Elapsed	CPU Time	SRB Time
DBU1P113	ICB link	4:07	34.71	0.08

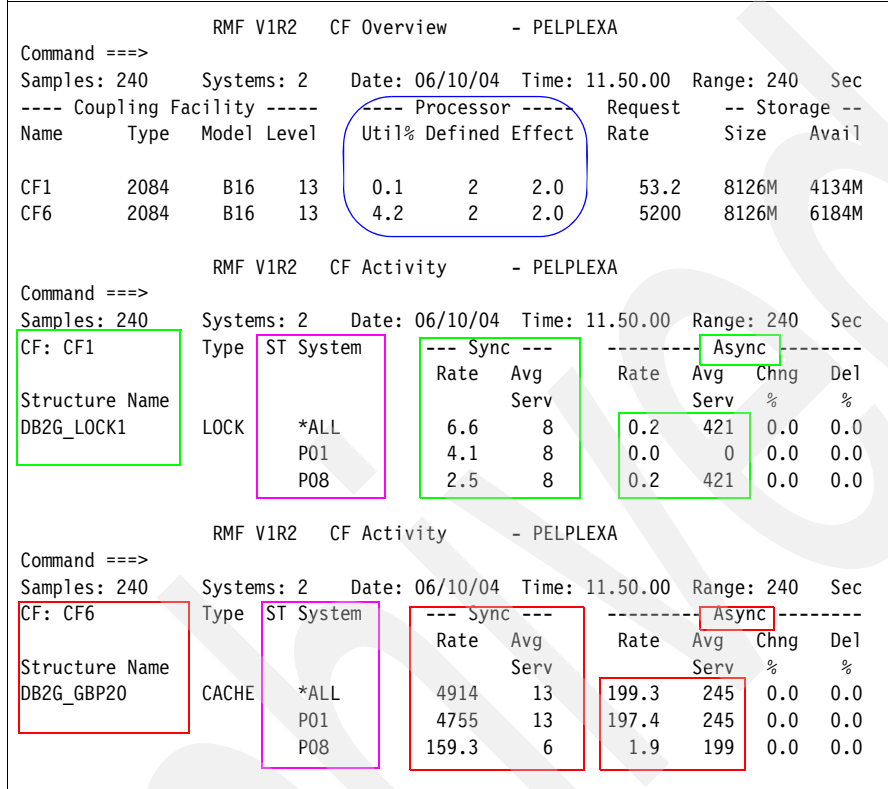


Figure 2-10 DB2 GBP in simplex mode

The DB2_GBP20 structure was then duplexed and the test rerun. The statistics for the job using the structure in duplex mode are in Table 2-12 and the CF performance information is shown in Figure 2-11 on page 28.

Table 2-12 Job statistics for DB2 GBP duplex measurement job

Step/Jobname	Link Type for CF6	Elapsed	CPU Time	SRB Time
DBU1P113	ICB link	4:07	33.75	0.09

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 240  Systems: 2  Date: 06/10/04  Time: 11.25.00  Range: 240  Sec
---- Coupling Facility ---- Processor Request -- Storage --
Name  Type  Model Level  Util% Defined Effect  Rate  Size  Avail
CF1   2084   B16  13    1.6    2    2.0    2136  8126M  3938M
CF6   2084   B16  13    3.7    2    2.0    5014  8126M  6184M

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 240  Systems: 2  Date: 06/10/04  Time: 11.25.00  Range: 240  Sec
CF: CF1
Type ST System  --- Sync ---  --- Async ---
Rate Avg Rate Avg Chng Del
Serv Serv % %
Structure Name
DB2G_GBP20 SEC CACHE *ALL 0.1 13 2053 58 0.0 0.0
P01 0.0 0 2053 58 0.0 0.0
P08 0.1 13 0.0 0 0.0 0.0
DB2G_LOCK1 LOCK *ALL 26.8 9 0.2 191 0.0 0.0
P01 23.9 10 0.0 0 0.0 0.0
P08 2.9 8 0.2 191 0.0 0.0

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 240  Systems: 2  Date: 06/10/04  Time: 11.25.00  Range: 240  Sec
CF: CF6
Type ST System  --- Sync ---  --- Async ---
Rate Avg Rate Avg Chng Del
Serv Serv % %
Structure Name
DB2G_GBP20 PRIM CACHE *ALL 4706 13 187.9 222 0.0 0.0
P01 4706 13 187.9 222 0.0 0.0
P08 0.2 9 <0.1 147 0.0 0.0

```

Figure 2-11 DB2 buffer pool application duplexing

There are a number of interesting aspects to this performance information:

- ▶ In the base run, CF6 was 4.2% busy doing 5200 requests per second. In the duplex test, the request rate was down marginally to 5014 and the CPU utilization was also down marginally to 3.7%. In other words, duplexing the structure had no impact on CPU utilization of the CF with the primary structure.
- ▶ The utilization of CF1 was .1% in the base run, handling just 53 requests a second. In the duplex run, the lock request rate rose a bit and there were 2053 GBP requests per second. The utilization of CF1 increases to 1.6%. This equates to roughly the same amount of CF CPU time for each request to the secondary structure as was observed for requests to the primary structure. You can see how this is very different to the CF CPU utilization effect of system-managed duplexing.
- ▶ The response time for the GBP structure in simplex mode was 13 microseconds. The response time for the structure in duplex mode was also 13 microseconds.
- ▶ All accesses to the secondary copy of the structure are asynchronous. This is the design of DB2 GBP duplexing. For selected updates to the primary structure, DB2 issues an asynchronous update to the secondary structure, immediately followed by a synchronous request to the primary structure so that both updates run in parallel.
- ▶ The number of requests to the secondary structure is a little more than half the number of requests to the primary structure. DB2 only mirrors data updates to the secondary structure, minimizing the number of requests, and hence the overhead of duplexing the structure. The number of updates to the secondary structure depends on how write-intensive the workload is.

Duplexing a GBP structure is enabled by defining the structure in the CFRM policy as either DUPLEX(ENABLED) or DUPLEX(ALLOWED). No changes are required in DB2.

Summary

The performance impact of enabling duplexing for the GBP structure was minimal, and so was the impact on total CF CPU utilization (across both CFs).

DB2 GBP duplexing is highly recommended. It consumes very little CPU, yet it can avoid the impact of a running recovery procedure if data were to be lost.

Archived

Archived

WebSphere MQ queue sharing structures

In this chapter, we describe the tests that we ran in the WebSphere MQ environment.

When you review the WebSphere MQ test results, it is important to keep several points in mind:

- ▶ The programs used during the tests performed only a single function, either the PUT activity or the GET activity. This allowed the cloning of the GET jobs, which can be useful in simulating multiple listeners or keeping message buildup in the structures at an acceptable level.
- ▶ RMF Monitor III was used to present reports that narrowed the intervals selected for generating the utilization and activity rates. When idle time is present in an interval, the rates are affected and can cause deviations in reported utilization.
- ▶ The intent of these tests was to reduce noise/activity to achieve pure numbers. This would have been impossible in a production environment.
- ▶ The bulk of the results are based on 200-byte message payloads for the following reasons:
 - Applications consist of consumer applications and provider applications.
 - Think of the messages in terms of a requestor and a replier. Request messages often provide key data that is used to access and format the reply. The tendency is for requests to be small and replies to be large. This is why we focused on a small message size.
 - Requests are more likely candidates for shared queues if the message size is manageable and minimal.
 - Replies need to be directed to the requestor (affinity).

Reply messages are usually large, and therefore are less likely candidates for shared queue implementations. This implies that a specific WebSphere MQ Queue Manager connection must be used to obtain the reply. It is much more difficult to justify the use of shared queues for large reply messages, which is the reason the bulk of the testing was performed for a small message size.

The service provider can exist in multiple LPARs, justifying the need for a shared queue for availability and ease of management.

Archived

3.1 MQ Application structures

The goal of these tests was to understand the IBM 2084 machine family performance in a sysplex environment that is supported by an IBM 2084 CF. All these tests were run using WebSphere MQ 5.3.

We ran controlled tests with a minimum of contention because we wanted to achieve the following goals:

- ▶ Understand the affects of using various link types. The testing configuration allowed us to use various combinations of ISC, ICB, and IC for local ICFs.
- ▶ Develop an enhanced understanding of the factors that trigger the use of the MQ Admin structure.
- ▶ Understand the impact CF CPU utilization has on service time.

3.2 MQ Application and Admin structures in simplex

The tests were run with one job putting the messages over two application structures to generate activity against the Admin structure and with two jobs retrieving messages from a single queue. The retrieving jobs ran slower than the job writing the messages.

The tests were kept relatively short, so the rates and utilization numbers were affected by short idle time. For this reason, we focused on the structure service times, because the processor utilization and the activity rates were affected by lack of activity.

3.2.1 ISC links

This test consisted of a one-PUT program and a two-GET program running on LPAR P01. The CF structures participating in the test were:

- ▶ QSG1APPLLARGE1: Application in CF1
- ▶ QSG1APPLMEDIUM1: Application in CF1
- ▶ QSG1CSQ_ADMIN: Admin in CF6

The programs used were GRHMQGET to retrieve the messages, and GRHMQPT2 to put messages on the two separate structures. The message size is 200 bytes.

The first result, as shown in Figure 3-1 on page 34, documents a run with the jobs running on the P01 LPAR, connecting to the MQ application structures through IC links and to the Admin structure through ISC links. This test set the base for further comparisons.

```

RMF V1R2 CF Overview - PELPLEXA
Command ==>
Samples: 120 Systems: 2 Date: 06/10/04 Time: 10.42.00 Range: 120 Sec
---- Coupling Facility ----
Name Type Model Level Util% Defined Effect Request Rate Size Avail
CF1 2084 B16 13 19.5 2 2.0 25191 8126M 3938M
CF6 2084 B16 13 0.6 2 2.0 275.7 8126M 6184M

RMF V1R2 CF Activity - PELPLEXA
Command ==>
Samples: 120 Systems: 2 Date: 06/10/04 Time: 10.42.00 Range: 120 Sec
CF: CF1 Type ST System --- Sync --- Async ---
Rate Avg Rate Avg Chng Del
Serv % %
Structure Name
QSG1APPLLARGE1 LIST *ALL 12516 13 31.1 799 0.0 0.0
P01 12512 13 24.4 959 0.0 0.0
P08 4.8 33 6.7 219 0.0 0.0
QSG1APPLMEDIUM1 LIST *ALL 12510 13 27.8 850 0.0 0.0
P01 12507 13 24.5 938 0.0 0.0
P08 2.7 28 3.4 216 0.0 0.0

RMF V1R2 CF Activity - PELPLEXA
Command ==>
Samples: 120 Systems: 2 Date: 06/10/04 Time: 10.42.00 Range: 120 Sec
CF: CF6 Type ST System --- Sync --- Async ---
Rate Avg Rate Avg Chng Del
Serv % %
Structure Name
QSG1CSQ_ADMIN LIST *ALL 3.1 29 253.1 101 0.0 0.0
ISC P01 1.0 64 253.1 101 0.0 0.0
P08 2.1 13 0.0 0 0.0 0.0

```

Figure 3-1 MQ structure

In the second run, the jobs were moved from the P01 LPAR to the P08 LPAR. These jobs used ISC links to access the application structures on CF1 (instead of IC as in the previous test) and IC links to connect to the Admin structure on CF6. Figure 3-2 on page 35 summarizes the CF service times.

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 180  Systems: 2  Date: 06/10/04  Time: 10.56.00  Range: 180  Sec
---- Coupling Facility ----  ---- Processor ----  Request  -- Storage --
Name      Type  Model Level  Util% Defined Effect  Rate      Size  Avail
CF1       2084   B16   13   30.5   2   2.0   13015  8126M  3938M
CF6       2084   B16   13   0.3    2   2.0   206.8  8126M  6184M

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 180  Systems: 2  Date: 06/10/04  Time: 10.56.00  Range: 180  Sec
CF: CF1      Type  ST System  --- Sync ---  ----- Async -----
                Rate  Avg  Rate  Avg  Chng  Del
                Serv  Serv  %    %
Structure Name
QSG1APLLARGE1  LIST  *ALL  64.3  34  6476  84  0.0  0.0
                P01  28.4  13  0.0   0  0.0  0.0
                P08  35.9  51  6476  84  0.0  0.0
QSG1APPLMEDIUM1  LIST  *ALL  44.6  41  6468  83  0.0  0.0
                P01  14.3  12  0.0   0  0.0  0.0
                P08  30.2  55  6468  83  0.0  0.0

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 180  Systems: 2  Date: 06/10/04  Time: 10.56.00  Range: 180  Sec
CF: CF6      Type  ST System  --- Sync ---  ----- Async -----
                Rate  Avg  Rate  Avg  Chng  Del
                Serv  Serv  %    %
Structure Name
QSG1CSQ_ADMIN  LIST  *ALL  126.2  12  19.2  223  0.0  0.0
                P01  0.1  74  18.5  200  0.0  0.0
                P08  126.2  12  0.7  814  0.0  0.0

```

Figure 3-2 MQ structures using ISC links

Summary

This test was run to verify the performance relationship between the CF service time and the CF link configuration. What you can observe in this run is that the processor utilization increased when ISC links were used instead of IC links. In the second run, the activity on CF1 moves to asynchronous and the overall throughput is negatively impacted. Also the service time increases, even though it is still good for the asynchronous requests for the application structures.

3.2.2 ICB links

This test consisted of a one-PUT program and a two-GET program running on LPAR P01. The CF structures participating in the test were:

- ▶ QSG1APLLARGE: Application in CF1
- ▶ QSG1APPLMEDIUM: Application in CF1
- ▶ QSG1CSQ_ADMIN: Admin in CF6

The programs used were GRHMQGET to retrieve the messages and GRHMQPT2 to put messages on the two separate structures. The message size is 200 bytes.

In this run, we focused on the Admin structure. The configuration was changed to use ICB links to connect to the Admin structure on the CF6 and to use IC links to connect to the application structures on CF1.

Figure 3-3 summarizes the CF service times.

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 120  Systems: 2  Date: 06/10/04  Time: 11.56.00  Range: 120  Sec
---- Coupling Facility ----
Name  Type  Model Level  Util% Defined Effect  Request  -- Storage --
Rate  Size  Avail
CF1   2084  B16  13  18.6  2  2.0  24191  8126M  4134M
CF6   2084  B16  13  0.4   2  2.0  335.8  8126M  6059M

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 120  Systems: 2  Date: 06/10/04  Time: 11.56.00  Range: 120  Sec
CF: CF1  IC  Type ST System  --- Sync ---  ----- Async -----
Rate  Avg  Rate  Avg  Chng  Del
Serv  Serv  %    %
Structure Name
QSG1APLLARGE1  LIST  *ALL  12079  13  23.6  952  0.0  0.0
P01  12070  13  23.6  952  0.0  0.0
P08  9.2  17  0.0  0  0.0  0.0
QSG1APPLMEDIUM1  LIST  *ALL  12510  13  24.4  924  0.0  0.0
P01  12505  13  24.4  924  0.0  0.0
P08  4.5  17  0.0  0  0.0  0.0

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 120  Systems: 2  Date: 06/10/04  Time: 11.56.00  Range: 120  Sec
CF: CF6  Type ST System  --- Sync ---  ----- Async -----
Rate  Avg  Rate  Avg  Chng  Del
Serv  Serv  %    %
Structure Name
QSG1CSQ_ADMIN  LIST  *ALL  248.5  18  4.3  398  0.0  0.0
ICB  P01  246.9  18  4.3  398  0.0  0.0
P08  1.5  13  0.0  0  0.0  0.0
  
```

Figure 3-3 MQ structures using IC links

In the second run, the jobs were moved from the P01 LPAR to the P08 LPAR. The jobs used ICB links to access the application structures on CF1 and IC links to connect to the Admin structure on CF6. Figure 3-4 on page 37 shows a summary of the CF service times.

```

RMF V1R2 CF Overview - PELPLEXA
Command ==>
Samples: 60 Systems: 2 Date: 06/10/04 Time: 12.00.00 Range: 60 Sec
---- Coupling Facility ----
Name Type Model Level Util% Defined Effect Request Rate Size Avail
CF1 2084 B16 13 31.2 2 2.0 32773 8126M 4134M
CF6 2084 B16 13 0.4 2 2.0 400.6 8126M 6059M

RMF V1R2 CF Activity - PELPLEXA
Command ==>
Samples: 60 Systems: 2 Date: 06/10/04 Time: 12.00.00 Range: 60 Sec
CF: CF1 ICB Type ST System --- Sync --- Async ---
Rate Avg Rate Avg Chng Del
Serv % %
Structure Name
QSG1APPLARGE1 LIST *ALL 14660 21 31.1 901 0.0 0.0
P01 1.8 11 0.0 0 0.0 0.0
P08 14658 21 31.1 901 0.0 0.0
QSG1APPLMEDIUM1 LIST *ALL 15049 20 31.6 886 0.0 0.0
P01 0.8 11 0.0 0 0.0 0.0
P08 15048 20 31.6 886 0.0 0.0

RMF V1R2 CF Activity - PELPLEXA
Command ==>
Samples: 60 Systems: 2 Date: 06/10/04 Time: 12.00.00 Range: 60 Sec
CF: CF6 Type ST System --- Sync --- Async ---
Rate Avg Rate Avg Chng Del
Serv % %
Structure Name
QSG1CSQ_ADMIN LIST *ALL 306.7 11 1.8 937 0.0 0.0
P01 1.1 19 0.0 0 0.0 0.0
P08 305.6 11 1.8 937 0.0 0.0
IC

```

Figure 3-4 MQ structures using ICB links

Summary

This test was run to verify the performance relationship between the CF service time and the CF link configuration. If we focus on the Admin structure, we can see the effect of using different links. In Figure 3-1 on page 34, we used ISC links for the Admin structure with an average response time of 64 microseconds and most of the requests processed as asynchronous; in Figure 3-3 on page 36, you can observe that using ICB links provides a better service time and that most of the CF requests against the Admin structure are now processed as synchronous requests. In the last run described in Figure 3-4, the Admin structure is accessed through IC links with similar results. Link type affects service times and the mode of access to the structures.

3.3 MQ Admin structures in duplex

The Admin structure is a single point of failure to a queue sharing group because all members must restart to resume data sharing if the Admin structure is lost.

The following runs were used to understand the impact of duplexing the Admin structure with different link configurations.

3.3.1 ISC links

This test consisted of a one-PUT program and a two-GET program running on LPAR P01. The CF structures participating in the test were:

- ▶ QSG1APPLLARGE1: Application in CF1
- ▶ QSG1APPLMEDIUM1: Application in CF1
- ▶ QSG1CSQ_ADMIN: Admin in CF6 (duplex in CF1)

The programs used were GRHMQGET to retrieve the messages and GRHMQPT2 to put messages on the two separate structures. The message size is 200 bytes.

In this run, the same workload and same hardware that were used in the base run were used. System-managed duplexing for the Admin structure was enabled. Figure 3-5 reports the performance data about the run.

```

RMF V1R2  CF Overview  - PELPLEXA
Command ==>
Samples: 120  Systems: 2  Date: 06/10/04  Time: 09.27.00  Range: 120  Sec
---- Coupling Facility -----
Name      Type  Model Level  Util% Defined Effect  Request Rate  -- Storage -- Size Avail
CF1       2084  B16  13    21.3    2    2.0    26132  8126M  4123M
CF6       2084  B16  13    1.1    2    2.0    377.7  8126M  6184M
    
```

```

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 120  Systems: 2  Date: 06/10/04  Time: 09.27.00  Range: 120  Sec
CF: CF1      Type  ST System  --- Sync ---  ----- Async -----
                Rate  Avg  Rate  Avg  Chng  Del
Structure Name  Serv  %    %
QSG1APPLLARGE1 LIST  *ALL  12514  13  29.6  822  0.0  0.0
                P01  12510  13  24.5  958  0.0  0.0
                P08  3.6  27  5.1  176  0.0  0.0
QSG1APPLMEDIUM1 LIST  *ALL  12508  13  27.1  865  0.0  0.0
                P01  12506  13  24.4  939  0.0  0.0
                P08  2.0  34  2.7  183  0.0  0.0
QSG1CSQ_ADMIN  LIST  *ALL  2.1  73  251.3  125  0.0  0.0
                P01  1.0  77  250.8  125  0.0  0.0
                P08  1.1  70  0.5  204  0.0  0.0
    
```

```

RMF V1R2  CF Activity  - PELPLEXA
Command ==>
Samples: 120  Systems: 2  Date: 06/10/04  Time: 09.27.00  Range: 120  Sec
CF: CF6      Type  ST System  --- Sync ---  ----- Async -----
                Rate  Avg  Rate  Avg  Chng  Del
Structure Name  Serv  %    %
QSG1CSQ_ADMIN  LIST  *ALL  2.1  67  251.3  131  0.0  0.0
                ISC  P01  1.0  85  250.8  130  0.0  0.0
                P08  1.1  50  0.5  194  0.0  0.0
    
```

Figure 3-5 MQ structures using ISC links - duplexing

From this report, you can see that system-managed duplexing has been enabled on the Admin structure. Asynchronous access for the Admin structure with the ISC links is used. You can also see that the prim/sec request rates are identical, which is typical for an MQ structure where everything is duplexed.

Summary

MQ availability is sensitive to a failure in the Admin structure. All the sharing group members must be restarted for an Admin structure failure. If so, you can use system-managed duplexing for this structure. In this run, you can see that the service time increases from 74 to 85 microseconds, but it might still be acceptable to your installation based on the activity rate.

3.3.2 ICB links

This test consisted of a one-PUT program and a two-GET program running on LPAR P08. The CF structures participating in the test were:

- ▶ QSG1APLLARGE1: Application in CF1
- ▶ QSG1APPLMEDIUM1: Application in CF1
- ▶ QSG1CSQ_ADMIN: Admin in CF6 (duplex in CF1)

Duplexing was enabled for the Admin structure in the CF1. The programs used were GRHMQGET to retrieve the messages and GRHMQPT2 to put messages on the two separate structures. The message size is 200 bytes.

The run was made using ICB links from LPAR P08 to CF1 with the MQ application structures and IC links to connect to the Admin structure in CF6. Figure 3-6 on page 40 shows a summary of the CF service times.

```

RMF V1R2 CF Overview - PELPLEXA
Command ==>
Samples: 60 Systems: 2 Date: 06/10/04 Time: 12.05.00 Range: 60 Sec
---- Coupling Facility ---- Processor Request -- Storage --
Name Type Model Level Util% Defined Effect Rate Size Avail
CF1 2084 B16 13 30.0 2 2.0 30881 8126M 4123M
CF6 2084 B16 13 0.9 2 2.0 337.0 8126M 6059M

RMF V1R2 CF Activity - PELPLEXA
Command ==>
Samples: 60 Systems: 2 Date: 06/10/04 Time: 12.05.00 Range: 60 Sec
CF: CF1 ICB Type ST System --- Sync --- Async ---
Rate Avg Rate Avg Chng Del
Serv Serv % %
Structure Name
QSG1APLLARGE1 LIST *ALL 14178 21 27.7 979 0.0 0.0
P01 4.6 12 0.0 0 0.0 0.0
P08 14173 21 27.7 979 0.0 0.0
QSG1APPLMEDIUM1 LIST *ALL 14555 20 28.3 961 0.0 0.0
P01 3.0 17 0.0 0 0.0 0.0
P08 14552 20 28.3 961 0.0 0.0
QSG1CSQ_ADMIN LIST *ALL 295.9 42 3.6 1826 0.0 0.0
P01 0.3 39 1.4 169 0.0 0.0
P08 295.5 42 2.2 2863 0.0 0.0

RMF V1R2 CF Activity - PELPLEXA
Command ==>
Samples: 60 Systems: 2 Date: 06/10/04 Time: 12.05.00 Range: 60 Sec
CF: CF6 Type ST System --- Sync --- Async ---
Rate Avg Rate Avg Chng Del
Serv Serv % %
Structure Name
QSG1CSQ_ADMIN LIST *ALL 295.9 36 3.6 1822 0.0 0.0
P01 0.3 40 1.4 170 0.0 0.0
P08 295.5 36 2.2 2856 0.0 0.0

```

Figure 3-6 MQ structures using ICB links - simplex

From this report, you can see that system-managed duplexing has been enabled on the Admin structure and then synchronous access is used for the Admin structure even when duplexed.

Summary

This run shows that duplexing the Admin structure can be achieved if your installation is looking for high availability and that the cost and frequency of the Admin structure makes any decision not to duplex a difficult one to support. ICB makes duplexing much faster, and the MQ Admin structure has data so the higher ICB bandwidth helps there too.

The performance results that were obtained from the duplexing implementation for the Shared Queue indicates that these structures are good candidates for local ICF implementations.



Results

This chapter describes the solution that the insurance company adopted based on the benchmark run and the lessons learned during the execution of the projects. These general lessons can be used for any installation.

4.1 Configuration selected

After running the test cases, the company concluded that the most suitable configuration for their workload consisted of:

- ▶ 4 CECs with 5 LPARs for the production environment
 - Different 2084 models with different number of engines
 - 1 B16 machine
 - 3 C24 machines
- ▶ One CEC 2084-B16 with multiple LPARs for the test environment
- ▶ Two 2084 A08 CFs with two engines each
- ▶ As we moved to 2084s, the link architecture was also upgraded to ICB4 links

By moving from an R06 CF to 2084 external ICFs with two engines each, this company had the expectations that the CF requests would switch to synchronous access from asynchronous, that the service times would be drastically reduced, and that a reduction in CPU utilization from 55% to between 17% and 19% would be achieved.

After updating the configuration, some of the expectations became true:

- ▶ The CF requests moved heavily from asynchronous access to synchronous.
- ▶ The service time was drastically reduced. For example, the service times to DB2 buffer pools improved from 250 to 300 microseconds to 35 microseconds when using ISCs and 13 microseconds when using ICB4.
- ▶ The CPU utilization was reduced from about 55% to about 25%. In fact, external CFs have been heavily using the CF links as communication media, abandoning the CTCs. This resulted in an increase in the CF activity of 30%.
- ▶ On the other hand, it was observed that the transaction elapsed time improved from 15% to 25%, although the transaction CPU consumption remained consistent.

In addition, the following elements were noticed:

- ▶ ICB links showed the first synchronous to asynchronous change event, indicating that the links are busy when a synchronous request is issued. However, the number of these events has been extremely low.
- ▶ Connectivity concerns arose because the link requirements were using almost the entire capacity of the first book STI ports. There can be a benefit to using IC links from the local z/OS image to free up two STIs in addition to any open STIs in the single book. This issue needs to be reviewed because there might be a need to upgrade the CFs to a B16 model for STI ports even though only 40% of two engines is used.
- ▶ The traffic directed now to the local CF make it possible to keep the configuration to two engines in each of the external CFs.

4.2 Lessons learned

The following lessons and considerations were learned about this installation in the process of taking part in this project:

- ▶ The intended direction to implement only local ICFs in the future has been changed. The DB2 lock structure is an example of a structure that should be located either in an external ICF or duplexed for fault tolerance.

- ▶ An external CF with a local CF as backup might be an acceptable configuration because you do not necessarily need two external CFs.
- ▶ Service time for CF requests is affected by the type of the link and the CF utilization. These are the elements that you should carefully plan in your environment.
- ▶ The performance aspects of the IC links, coupled with the benefits of freeing the STI ports, enforce the conclusion that the introduction of local ICFs remains an important direction in the future strategy for the installation, and it will be used to enhance their external ICF configuration.
- ▶ When duplexing of structures is desired, ICB links should be used when possible.
- ▶ The selection of the structures that are targeted for the external ICFs should be based on requirements for recoverability, performance criticality attributes, or both.
- ▶ Local ICFs can be used as a means of protecting the structures selected to be on the external ICFs. For example, the DB2 lock structure is suited for an external CF for the following reasons:
 - Performance impacts of duplexing could not be tolerated.
 - Placing the DB2 lock structure on a local ICF would require that duplexing be established because a single hardware failure could lose the lock information in the connected IRLM and the ICF structure.
 - Local ICFs could be used to off load lower level tasks that tolerate failures (XCF for instance), protecting the processor capacity on the external ICF and providing ICB relief (from the 16 ICB total link perspective).

Archived



Coupling Facility Performance: A Real World Perspective



**Comparison of
different Coupling
Facility
configurations**

**Considerations for
system-managed
duplexing**

**Focus on DB2 and
WebSphere MQ**

This Redpaper can help you understand the Coupling Facility technology and how different configuration options can affect your workload. It describes how to configure your Coupling Facility to enhance performance and high availability in a zSeries Parallel Sysplex environment.

The information presented is based on the results of a benchmark by a large IBM client at the IBM Benchmark Center in Poughkeepsie, New York. The company decided to run benchmarks to verify how the different configurations fit with their workload and obtain the best performance and availability results.

**INTERNATIONAL
TECHNICAL
SUPPORT
ORGANIZATION**

**BUILDING TECHNICAL
INFORMATION BASED ON
PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:
ibm.com/redbooks**