

February 2013



z/VM 6.2 Live Guest Relocation with Linux Middleware

z/VM 6.2 Live Guest Relocation with Linux Middleware

Table of Contents

About this publication	3
Authors	3
Introduction	4
z/VM 6.2	4
SSI cluster	4
Relocation concepts	5
Objectives	5
Summary	6
Hardware and software configuration	7
Server hardware	7
System z - LPAR Configuration	7
Storage server - DS8K15	7
IOCDS configuration for FICON CTCs	7
FCTC connections	8
Server software	9
z/VM and Linux configuration	10
z/VM configuration	10
ISFC logical link configuration	10
General	10
Avoidance of write collisions	11
Linux guest configuration	12
z/VM monitor configuration	12
Test approach	12
Test sequence	13
Test workload	13
Test configurations	14
ISFC logical link configurations with "straight" FCTC connections	14
ISFC logical link configurations with "crossed" FCTC connections	15
Striping of FCTC device numbers	15
Test metrics	15
Test results	16
General	16
Performance of the relocation process	16
Baseline test	16
Scaling the number of FCTC connections per ISFC logical link	17
Common observations	28
Linux guest performance during the relocation process	28
Number of memory passes	29
z/VM system load during the relocation process	29
ISFC logical link performance	30
ISFC logical link data rates	30
FCTC connection I/O rates	33
Influence of striping of FCTC devices numbers over FICON channels	34
Recommendations	42
Appendix. IOCDS example	44
References	45
Notices	46
Terms and conditions	48

z/VM 6.2 Live Guest Relocation with Linux Middleware

About this publication

This paper provides results for testing live guest relocation of z/VM[®] virtual machines running Linux[®] on IBM System z[®].

This is the first of a series of papers and it focuses on setup considerations to optimize the relocation speed in a Java[™] environment.

Authors

Michael Johansen

Dr. Juergen Doelle

z/VM 6.2 Live Guest Relocation with Linux Middleware

Introduction

An introduction to z/VM 6.2 concepts and an overview of the tests described in this paper.

z/VM 6.2

Version 6.2 of the IBM z/VM operating system introduced two concepts that provide the functional foundation for the tests presented in this paper.

- z/VM Single System Image (SSI)
- z/VM Live Guest Relocation (LGR)

A z/VM single system image (SSI) cluster is a multi z/VM system environment in which the z/VM systems can be managed as a single resource pool and guests can be moved from one system to another while they are running.

Note: SSI (including the LGR function) is a z/VM 6.2 priced feature that you can order separately.

Note: As of z/VM 6.2 the LGR function is only available for virtual machines running Linux for System z as the guest operating system.

SSI cluster

A z/VM SSI cluster consists of up to four z/VM systems in an Inter-System Facility for Communications (ISFC) collection.

Each z/VM system is a member of the SSI cluster. *Figure 1* shows the basic structure of a cluster with four members.

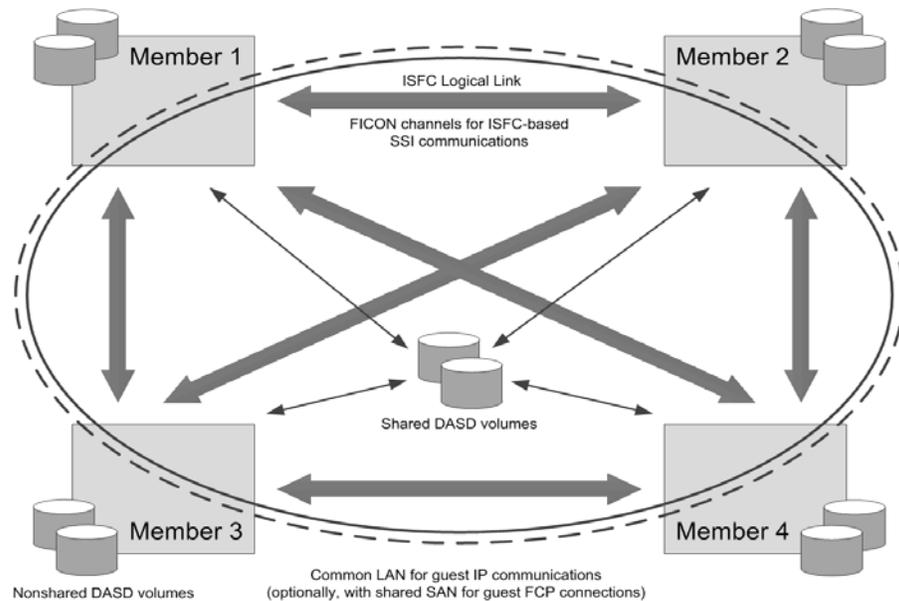


Figure 1: A four member z/VM SSI cluster (from IBM z/VM Planning and Administration)

z/VM 6.2 Live Guest Relocation with Linux Middleware

The SSI cluster is self-managed by CP using ISFC messages that flow across channel-to-channel devices between the members. All members can access shared DASD volumes, the same Ethernet LAN segments, and the same storage area networks (SANs).

The members of an SSI cluster are connected through ISFC logical links. ISFC logical links are z/VM logical elements defined to the z/VM systems composing the SSI cluster. ISFC logical links in turn are configured to use FICON[®] CTC (FCTC) devices on FICON channels that connect the logical partitions hosting the z/VM systems.

Relocation concepts

A z/VM SSI cluster provides a virtual server mobility function called live guest relocation (LGR).

A running virtual server (guest virtual machine) can be relocated from one member to another. Relocating virtual servers can be useful for load balancing and for moving workload off a member system that requires maintenance. After maintenance is applied to a member, guests can be relocated back to that member, thereby allowing maintaining z/VM and to keep Linux virtual servers highly available.

The LGR function of z/VM is designed such that the performance impact on the relocated virtual system is minimized. Ideally, the guest operating system running within the virtual system and its applications to their clients should seem to run continuously; this should be the case regardless of the load situation of z/VM, of the guest operating system and of its applications. In other words, the impact resulting from the live guest relocation operation that are noticeable by external clients should be minimized. Particularly, no client visible error situations should result from a live guest relocation.

The "quiesce time" is a primary metric for the live guest relocation process. It is defined as the small amount of time during which - while the live guest relocation operation continues - the virtual server is effectively stopped.

Other relocation related metrics observable from z/VM are the consumption of resources needed for the relocation process, such as the consumption of CPU and memory, and the I/O rates and data amounts processed by the ISFC logical links, and their FICON CTC devices and FICON channels. Of course the relocation process is impacted if any of these resources is limited.

A metric observable from the perspective of a Linux guest operating system is the steal time. The steal time is the percentage of time that the guest operating system is involuntary wait state while z/VM is doing something else, such as for example performing a live guest relocation.

Objectives

The objective of this project is:

- To show the impact of live guest relocation on a Java application workload that runs in a single guest virtual machine, without any external resource pressure in order to eliminate external factors. The Java workloads are expected to have a specific memory pattern.

Particular focus is on the parameters such as:

- The relocation time that represents the overall time expended by z/VM to perform a live guest relocation
- The quiesce time that is the time during which a relocated virtual system is stopped; this is considered the primary metric for evaluating the performance of the live guest relocation process

z/VM 6.2 Live Guest Relocation with Linux Middleware

- To show the resource requirements for live guest relocations, such as:
 - The speed, data rates and I/O rates of ISFC logical links and their FCTC connections
 - The use of CPU and memory resources by z/VM for the live guest relocation process
 - The impact observable by a guest operating system, such as reduced availability of memory and CPU resources
- To give recommendations for configuring z/VM systems in order to optimize the live guest relocation process

Note: In this paper memory and storage sizes are expressed in multiples of 1024 byte, but data rates are expressed in multiples of 1000 byte.

Table 1 lists the notational conventions applied in this paper, in accordance to IEC 60027 - 2 Amendment 2:

Table 1. Notational conventions

Symbol	Full name	Derivation
KiB	kibibyte	2^{10} byte = 1024 byte
MiB	mebibyte	2^{20} byte = 1048576 byte
GiB	gibibyte	2^{30} byte = 1073741824 byte
bit/s	bit per second	10^0 bit / second
Kbit/s	kilobit per second	10^3 bit / second
Mbit/s	megabit per second	10^6 bit / second
Gbit/s	gigabit per second	10^9 bit / second
B/s	byte per second	10^0 byte / second
KB/s	kilobyte per second	10^3 byte / second
MB/s	megabyte per second	10^6 byte / second
GB/s	gigabyte per second	10^9 byte / second

Summary

The performance behavior of the live guest relocation of virtual machines was evaluated.

For each test, a live guest relocation of a 4 GiB z/VM guest virtual machine was performed while a Java workload was executed under Linux on System z running within the z/VM guest virtual machine. For details about the workload, see *Test workload*.

Depending on the configuration of the ISFC logical link, the live guest relocation took times from 158 down to 14.9 seconds. Furthermore, the quiesce time - the time while the virtual system is stopped during the relocation process and thus is not able to perform work - varied from 21 down to 1.6 seconds.

These variations emphasize the need for carefully configuring the ISFC logical links between the z/VM SSI cluster members, such that high data transfer rates result and minimize the impact of the relocation process on the workload performed by the guest virtual machine.

Given a situation similar to that exhibited by the test environment and the test workload used for the evaluations presented in this paper, the following recommendations should be considered:

- Prefer ISFC logical configurations with more than one FICON channel
 - In our tests the observed relocation and quiesce times scaled about inversely linear with the number of FICON channels

z/VM 6.2 Live Guest Relocation with Linux Middleware

- Consider using FICON channels for ISFC logical links as well as for DASD access if either usage only occurs occasionally; however, the usage of these DASDs should be low such that it does not interfere with z/VM's usage of FCTC connections on the same FICON channels
- Configure four *usable* FCTC connections from each FICON channel used for an ISFC logical link
 - In addition, because z/VM designates one or two FCTC connections for "unidirectional use", add one or two FCTC connections, respectively; for details, see *Recommendations*
- For ISFC logical links connecting z/VM members on LPARs residing on the same CEC, we recommend that you use ISFC logical link configurations with "straight" FCTC connections; for details, see *ISFC logical link configurations with "straight" FCTC connections*.
- If a virtual system is to be relocated to another z/VM member, with the choice of that target z/VM member being either on the same or on a different CEC, and the same set of FICON channels is configured for the ISFC logical links connecting to either target z/VM member, then relocating to the remote system might be faster because in this case the FICON channels are used only for sending the relocation data (instead sending *and* receiving in the local case).

Hardware and software configuration

This section provides details about the hardware and software used in our testing.

Server hardware

This section details the server configuration used in the tests.

System z - LPAR Configuration

For the first set of evaluations an IBM zEnterprise® 196 (z196), Model 2817-M66 was used.

That system was equipped with 66 CPs, 12 SAP and 1328 GiB customer storage. On that system two LPARs - LPAR1 and LPAR2 - were identically configured, as follows:

- 4 shared central processors (CPs)
- 20 GiB central storage
- 2 GiB expanded storage

Two FICON Express4 4Gbit channel cards were used for the switched FCTC connections between the two LPARs.

Storage server - DS8K15

The storage server DS8K15 is an IBM System Storage® DS8800, Model 951 with 8 host adapters connected with 4 FICON Express8 8 GBit channel cards.

IOCDS configuration for FICON CTCs

Initially three FICON channels from a FICON Express4 channel card were available.

In a later phase, we could use a second FICON Express4 channel card. In the IOCDS configuration for the CEC, these three FICON channels were assigned CHPIDs 3C, 3D and 3E; these FICON channels were used to configure FICON CTC (FCTC) connections that connect LPAR1 and LPAR2 in various ways.

z/VM 6.2 Live Guest Relocation with Linux Middleware

FCTC connections

For the purposes of this document, the term "FCTC connection" is used to refer to the pair of FCTC devices that are configured to be connected by means of the IOCDS configuration; for details, see System z, ESCON® and FICON Channel-to-Channel Reference. Occasionally, the term "FCTC device" could mean both "FCTC device" and "FCTC connection".

A graphic representation of the IOCDS based FCTC configuration is shown in *Figure 2*.

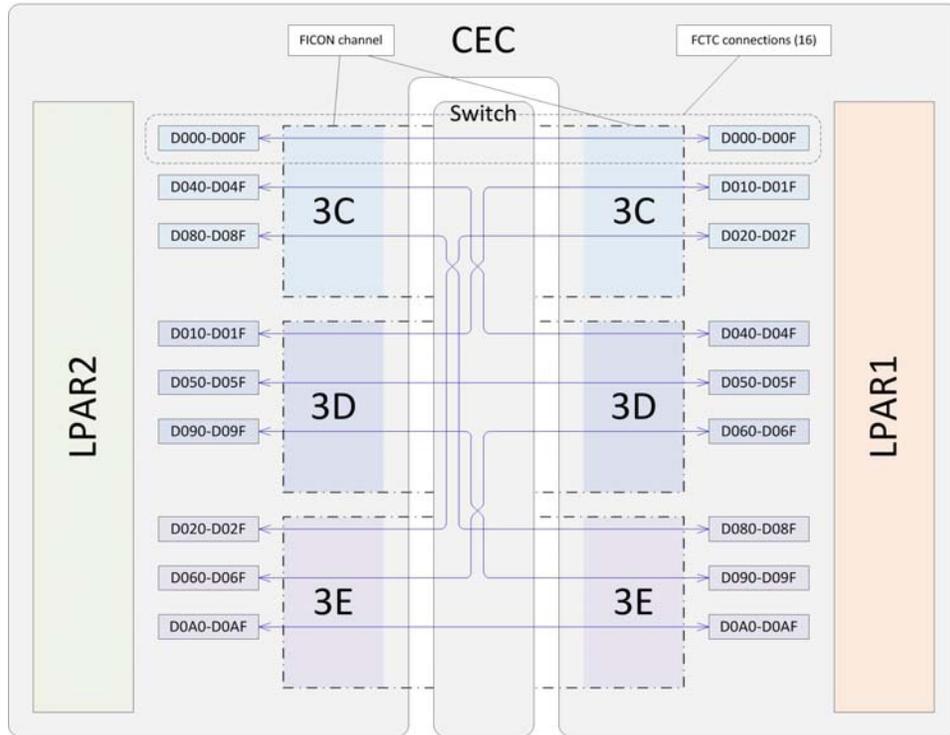


Figure 2. FCTC configuration

For each of the two LPARs, sets of FCTC connections were configured in the IOCDS, such that LPAR1 could potentially connect to LPAR2 through every possible FICON channel combination. Each possible FICON channel combination was configured with 16 FCTC devices on each side, resulting in 144 possible FCTC connections. The recommended convention of assigning the *same device number* for both FCTC devices of an FCTC connection was applied.

The FCTC connection configuration is detailed in *Table 2*.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Table 2. FCTC connection configuration

Source CHPID on LPAR1	FCTC device numbers	Target CHPID on LPAR2
3C	D000-D00F	3C
3D	D010-D01F	3C
3E	D020-D02F	3C
3C	D040-D04F	3D
3D	D050-D05F	3D
3E	D060-D06F	3D
3C	D080-D08F	3E
3D	D090-D09F	3E
3E	D0A0-D0AF	3E

The set of 144 possible FCTC connections presented in *Table 2* is the superset that is available for the z/VM configuration of the ISFC logical link connecting the two z/VM members running in the LPARs. The IOCDS configuration reflected in *Figure 3* and *Table 2* was used for our tests unless otherwise noted.

In addition, a "striping" approach was applied for some test cases. In this case, FCTC device numbers were not assigned in contiguous ranges, but in a cyclic manner, as shown in *Table 3*.

Table 3. "FCTC connection configuration with "striped" device numbers

Source CHPID on LPAR1	FCTC device numbers	Target CHPID on LPAR2
3C	D000, D003, D006, D009, D00C	3C
3D	D001, D004, D007, D00A, D00D	3D
3E	D002, D005, D008, D00B, D00E	3E

If the "striped" IOCDS configuration reflected in *Table 3* was used for a particular test, this is specifically noted.

Finally, a second FICON Express4 channel card was temporarily available for additional tests. The four FICON channels on that card had the CHPID numbers 4A, 4B, 4C and 4D. The IOCDS configuration for this card was done similar to that shown in *Table 2*, using device numbers D600 to D6FF. In addition, an alternative "striped" IOCDS configuration similar to that shown in *Table 3* was used for some tests.

A further IOCDS variant was the addition of 16 more FCTC device number for CHPID 3C, with FCTC device numbers D700-D70F, such that FCTC devices from CHPID 3C were defined at both the low and the high end of the device number range.

Server software

Table 4 shows the operating system software used on the z/VM SSI cluster, and for the guest virtual machines.

Table 4. Server software used

Product	Version/Level
IBM z/VM, (product number 5741-A07)	Version 6, Release 2, service level 1201
SUSE Linux Enterprise Server 11 (s390x)	SLES 11, SP2

z/VM 6.2 Live Guest Relocation with Linux Middleware

The following Java environment was used:

- IBM Java 1.6 SR8 FP1

z/VM and Linux configuration

This section describes the z/VM configuration and the ISFC logical link configuration

z/VM configuration

Within each of the two LPARs, a z/VM member is configured as part of a common SSI cluster.

Each z/VM member is configured to exploit all processors and all of the storage that is configured for the hosting LPAR (see *System z - LPAR Configuration*).

For each z/VM member, four paging devices were configured on the DS8K15 storage server (see *Storage server - DS8K15*). Each paging device had a size of 10016 cyl 3390 (equivalent to an IBM 3390 Model 9 DASD), resulting in a combined paging area of approximately 34 GB (roughly 1.5 times the real storage size available to each z/VM member). While this is slightly less than what is recommended for a typical z/VM installation, for the tests performed for this project z/VM paging was not a factor.

For each z/VM member, two spool devices were configured on the DS8K15 (see *Storage server - DS8K15*). Each spool device had a size of 10016 cyl 3390 (equivalent to an IBM 3390 Model 9 DASD), resulting in a combined spool space of approximately 17 GB. While spool space is configured separately for each member in an SSI cluster, the spool area as a whole is shared by all members.

The z/VM system was connected to an internal TCP/IP network via an OSA Express2 adapter.

ISFC logical link configuration

This section contains general configuration information and also details about how to avoid write collisions.

General

Specific attention was paid to the configuration of the ISFC logical link connecting the two z/VM members.

In general, the z/VM recommendations given in *z/VM CP Planning and Administration*, chapter *Planning the ISFC Network in an SSI Cluster* and those given in the *IBM z/VM Performance Report* (see *Live Guest Relocation and ISFC Improvements*) were followed. However, if necessary for the investigation of corner cases of the ISFC logical link configurations, some of the guidelines were not applied.

An ISFC logical link connecting two z/VM members is configured by selecting up to sixteen FCTC connections. These FCTC connections must have been previously configured in the IOCDS (*IOCDS configuration for FICON CTCs*), such that each FCTC connection in turn connects the two LPARs hosting the z/VM members.

For example, if the FCTC device with device number D000 is configured in IOCDS for LPAR1 and if in the z/VM member running LPAR1 that FCTC device D000 is activated for the ISFC logical link connecting to LPAR2, then a corresponding FCTC device - preferably with the same device number D000 - (the other "end" of the FCTC connection) must in turn be configured for LPAR2 in the IOCDS, and in the z/VM member running in LPAR2 must be activated for the ISFC logical link connecting to LPAR1.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Each side of an ISFC logical link is configured separately by `ACTIVATE ISLINK` statements in the z/VM system configuration file, by executing `ACTIVATE ISLINK` privileged CP commands, or both. Either way, `ACTIVATE ISLINK` requires the FCTC device number(s) as parameter(s). The use of CP commands provides for the ability to dynamically modify the actual configuration of ISFC logical links after IPL.

Once one or more respectively prepared FCTC connections are configured for an ISFC logical link on both of its sides, the z/VM members connected through the ISFC logical link can use the link for communication and data transfer; this includes the case of transporting the data resulting from virtual machines being relocated from one z/VM member to the other.

In the test environment, the z/VM system configuration file contained an initial configuration for the ISFC logical link between the two z/VM members that became effective at IPL time. Then, during the preparation phase of each test case, the ISFC logical link was dynamically reconfigured, thereby establishing the test specific ISFC logical link configuration requirements.

Avoidance of write collisions

A FCTC connection can potentially be used for communication in both directions.

However, if both sides try to send through a particular FCTC connection at the same time, a so called *write collision* occurs. While this is a recoverable situation, recovery consumes considerable processing and I/O resources and hence should be avoided.

In order to avoid write collisions, z/VM implements a specific usage convention for the FCTC connections configured for an ISFC logical link. By means of that usage convention, z/VM internally designates FCTC connections for unidirectional use.

The usage convention depends on the FCTC device numbers and the system names of the connected z/VM members; for details, see *ISFC Improvements*.

In summary, if two z/VM members are connected through an ISFC logical link, then

- The z/VM member with the lexicographically lesser system name¹ uses the FCTC devices starting from the lowest device number, up to higher device numbers
- The z/VM member with the lexicographically greater system name uses the FCTC devices starting from the highest device number, and down to lower device numbers
- If the ISFC logical link is composed of two up to eight FCTC devices, then each side uses at most one FCTC device less than configured, leaving the unused FCTC device for exclusive use of the adjacent side
- If the ISFC logical link is composed of nine up to sixteen FCTC devices, then each side uses at most two FCTC devices less than configured, leaving the unused FCTC devices for exclusive use of the adjacent side

For example, if the z/VM members in LPARs LPAR1 and LPAR2 were connected through an ISFC logical link with the FCTC connections with device numbers D000 through D00F, then LPAR1 (the system with the lexicographically lesser name) would perform its write operations on FCTC devices D000 up to D00D, leaving D00E and D00F exclusively for write operations of LPAR2. On the other hand, LPAR2 would perform its write operations on D00F down to D002, leaving FCTC devices D000 and D001 exclusively for write operations of LPAR1. This is illustrated in *Figure 3*.

z/VM 6.2 Live Guest Relocation with Linux Middleware

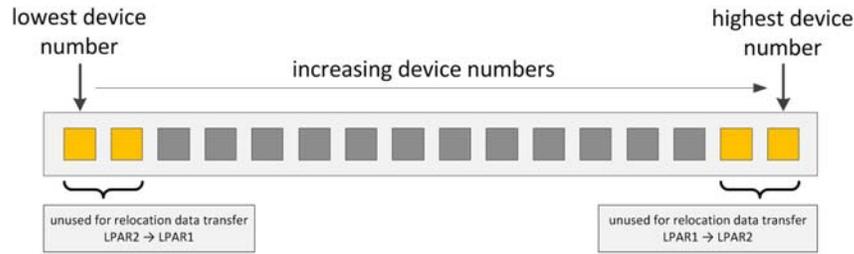


Figure 3. Devices designated by z/VM for unidirectional use, based on their device number

With live guest relocation, asymmetric load on an ISFC logical link is typical, such that at a particular point in time only one side is sending a large amount of data, while the adjacent side at that time typically only sends small amounts of control information back to the sender.

As long as more than one FCTC connection is configured for an ISFC logical link, the two latter of the above rules ensure that there always remains at least one (or two) FCTC device(s) designated for the recipient for sending back control information, thereby - for the typical asymmetric load conditions - minimizing the risk that write collisions occur.

However, these two rules also imply that not all connections of an ISFC logical link are used for transferring relocation data; this is particularly significant

- If the ISFC logical link is composed of only few FCTC connections, and / or
- If the unidirectional FCTC devices are the only ones configured from a particular FICON channel.

Linux guest configuration

The Linux guest virtual machine was configured as described in this section.

- 4 virtual CPUs
- 4096 MiB virtual memory
- One full pack minidisk on a FICON attached 3390 Model 27 DASD, with two partitions:
 - Partition 1 for the root file system, with a size of 19,999 MiB , and an ext3 file system
 - Partition 2 as a swap partition, with a size of 1,128 MiB
- NIC attached to a VSWITCH (z/VM Virtual Switch).

z/VM monitor configuration

Monitor event and sample data collection was enabled for the processor, storage, user, I/O, network, ISFC and SSI domains.

Specific monitoring of the FCTC devices in use for the particular test case was configured. The monitor sample interval was set to six seconds.

Test approach

In this project the performance of z/VM live guest relocation was evaluated.

For comparable results, a specific test sequence was defined and repeatedly executed for variations of a number of test configurations. The test sequence is detailed in *Test sequence*, the test configurations and their variations are detailed in *Test configurations*.

z/VM 6.2 Live Guest Relocation with Linux Middleware

The objective of the test sequence was performing and observing a z/VM live guest relocation while a specific test workload was executed under Linux on System z running in the guest virtual machine that was relocated. The test workload is detailed in *Test workload*.

The performance was evaluated by comparing values of specific metrics. The metric definitions are detailed in *Test metrics*

Test sequence

For each test variation, the sequence of major control steps described in this section was performed.

The steps were as follows:

1. The z/VMs on both test LPARs are IPLed as the only LPARs running on the CEC.
2. The ISFC logical link between the z/VM members is reconfigured according to the requirements of the test variation
3. Monitoring is started
4. The guest used for the test is started on one z/VM
5. The workload is started within the Linux guest operating system with a 10 minutes runtime
6. The workload execution is allowed to stabilize, in order to produce a comparable processor and memory usage pattern
7. After 7 minutes, the guest virtual machine is relocated to the other z/VM, while continuing to execute the workload
8. After the termination of the workload the test ends.
9. Data gathered from each run:
 - Workload specific performance data (e.g. throughput)
 - z/VM monitor data
 - Linux sysstat tool output (sadc/sar)

Test workload

A Java workload was executed within a Linux on system z operating system member running within the z/VM guest virtual machine that was relocated.

The workload was customized for producing a constant pressure on the CPU and memory resources such that the z/VM system was loaded up to but not beyond its respective capacities. This constant workload provides for the comparability of the relocation operations.

The Java setup parameters are listed in *Table 5*.

Table 5. Java Setup parameters

Parameter	Value
Java heap size	3200 MiB
Java garbage collection algorithm	optthroughput

The heap size was determined such that a maximum amount of the memory available within the Linux guest operating system was claimed by the workload.

The `optthroughput` garbage collection algorithm was selected because it causes intensive memory usage during the garbage collection phases. This was intended to produce a memory usage pattern with a large amount of updated pages spread over the whole Java heap size.

The execution time of 600s was determined such that on the source z/VM member the workload produced a stable resource consumption after an initialization phase, and, after the live guest relocation, on the target z/VM member again resumed a stable resource consumption.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Aside from this Linux guest virtual machine, no other significant workloads were performed within the z/VM systems.

Test configurations

For the connectivity between the z/VM members several different ISFC logical link configurations were tested.

A principal distinction between two categories of ISFC logical link configurations was made:

- ISFC logical link configurations with "straight" FCTC connections (*described in ISFC logical link configurations with "straight" FCTC connections*)

The rationale for "straight" FCTC connections was the idea that FICON channels might be able to optimize the data flow - at least to some extent - by not passing all data through the FICON switch.

- ISFC logical link configurations with "crossed" FCTC connections (*described in ISFC logical link configurations with "crossed" FCTC connections*)

The rationale for "crossed" ISFC logical link configurations were the ideas

- That send operations initiated on one FICON channel resulted in receive operations on a different FICON channel, and
- That the data flow had to pass through the switch in every case.

Any kind of ISFC logical link configuration is achieved by configuring only such FCTC devices into the ISFC logical link configuration that previously in the IOCDs were configured to connect the LPARs respectively.

ISFC logical link configurations with "straight" FCTC connections

Figure 4 shows various ISFC logical link configurations with "straight" FCTC connections. The same scheme as in Figure 2 is applied, but for brevity, the FCTC devices and the encompassing CEC are omitted.

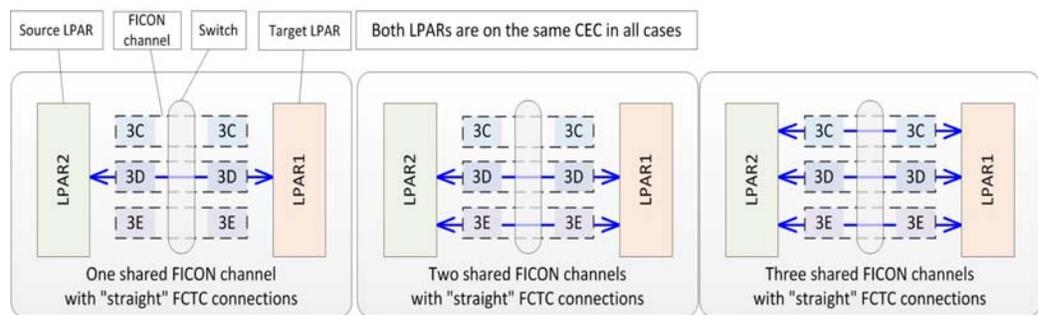


Figure 4. ISFC logical link configurations with "straight" FCTC connections

ISFC logical link configurations with "straight" FCTC connections for any of the FCTC connections composing the ISFC logical link, the FCTC devices at both ends of the FCTC connection are configured in the IOCDs on the *same* FICON channel.

It is important to realize that this is only possible when both LPARs reside on the same CEC and thus can *share* the FICON channel. In other words, "straight" FCTC connections always share FICON channels.

z/VM 6.2 Live Guest Relocation with Linux Middleware

ISFC logical link configurations with "crossed" FCTC connections

Figure 5 shows various ISFC logical link configurations with "crossed" FCTC connections. Again the same scheme as in Figure 2 is applied, but for brevity, the FCTC devices and the encompassing CEC are omitted.

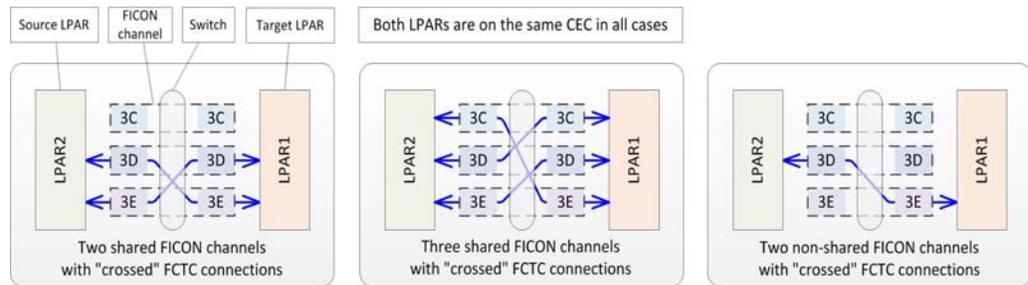


Figure 5. ISFC logical link configurations with "crossed" FCTC connections

With ISFC logical link configurations with "crossed" FCTC connections for any of the FCTC connections composing the ISFC logical link, the FCTC devices at both ends correspond to *different* FICON channels.

ISFC logical link configurations with "crossed" FCTC connections can be *shared* or *non-shared*:

- With "crossed" *shared* FCTC connections, the FICON channels are used by both LPARs at the same time. However, opposed to the "straight" case, data that is sent by one LPAR over one FICON channel is received by the other LPAR on a different FICON channel.
- With "crossed" *non-shared* FCTC connections, each FICON channel is exclusively used by only one LPAR. This emulates the case where the LPARs reside on different CECs.

Striping of FCTC device numbers

Striping of FCTC device numbers was configured as an alternative to typical approach of configuring sequential ranges of FCTC device numbers with identical connection properties.

For more information see *IOCDS configuration for FICON CTCs*.

The rationale behind configuring FCTC device numbers in a striped manner was:

- Exploring the influence of the FCTC device numbers on the distribution of the I/O load on configured FCTC devices and FICON channels, and
- Exploring whether a more granular control could be executed with respect to the selection of FCTC devices that z/VM dedicates for unidirectional use (see *Avoidance of write collisions*).

Test metrics

The performance of live guest relocations was evaluated using the metrics outlined in this section.

Relocation time

The elapsed time of the complete relocation process

Quiesce time

The elapsed time while the relocated system is quiesced; during this time, the guest virtual system is prevented from performing any work. The quiesce time is the most prominent metric when evaluating the quality of the guest relocation process.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Number of memory passes

The number of passes that z/VM performs over the virtual memory of a guest virtual machine during a live guest relocation. Because the guest virtual machine continues its work while the live guest relocation is performed, that work may cause changes of memory content that was already transferred in a previous pass. Consequently, the modified memory content must be transferred in another memory pass.

Amount of memory transferred during the quiesce time

The amount of memory that was transferred while the guest virtual machine was quiesced.

z/VM CPU usage

The relative percentage of CPU used by z/VM while performing a live guest relocation.

The performance of ISFC logical links was evaluated through the following parameters:

ISFC logical link peak write data rate

The maximum write data rate observed during a monitor interval during the live guest relocation.

FCTC connection I/O rate

The number of I/O requests per second [SSCH/s] that were performed by the FCTC device.

Note: The amount of data transferred per I/O request is not known.

FCTC device busy rate

Percent of elapsed time during which the device was busy with an I/O operation.

The performance of the Linux guest virtual system was evaluated through the following parameters:

Steal time percentage

Percentage of time spent in involuntary wait by the virtual CPU or CPUs while the hypervisor was doing something else, such as servicing another virtual processor or performing a live guest relocation.

Test results

General

The test results are presented under two major topics.

- *Performance of the relocation process* provides details about the relocation and quiesce times achieved with different ISFC logical link configurations
- *ISFC logical link performance* provides details about the data rates and the I/O rates observed for the ISFC logical link configurations

Performance of the relocation process

Baseline data was first gathered by running a test without a live guest relocation, this was followed by other tests where live guest relocation took place, and these subsequent tests provide information about the relocation and quiesce times achieved with different ISFC logical link configurations.

Baseline test

A base line test was performed without a live guest relocation taking place.

z/VM 6.2 Live Guest Relocation with Linux Middleware

The test was run for ten minutes, producing a specific throughput that was normalized to 1. The relative throughput of other test cases is presented as a percentage of this normalized throughput.

Scaling the number of FCTC connections per ISFC logical link

ISFC logical link configuration with one shared FICON channel and "straight" FCTC connections

Figure 6 shows an ISFC logical link configuration with one shared FICON channel and "straight" FCTC connections.

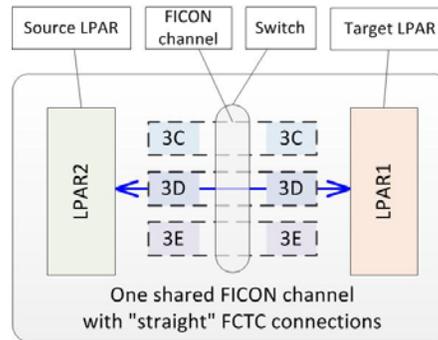


Figure 6. ISFC logical link configuration with one shared FICON channel and "straight" FCTC connections

Figure 7 shows how the relocation and the quiesce times scale in this case when the amount of FCTC connections configured for the ISFC logical link is varied.

One shared FICON channel with "straight" FCTC connections - Relocation and quiesce times

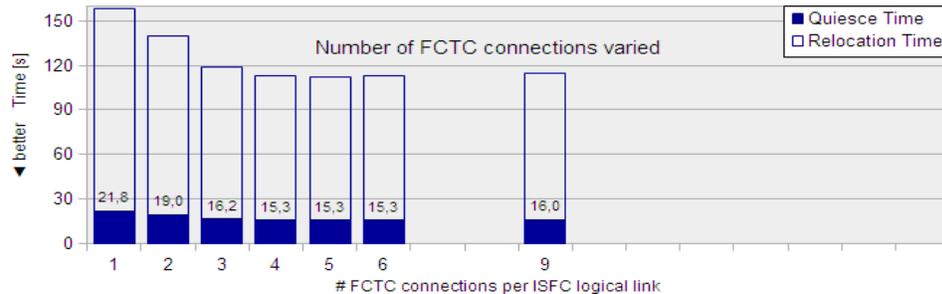


Figure 7. ISFC logical link configuration with one shared FICON channel and "straight" FCTC connections - Relocation and quiesce times

Observation

Figure 7 shows that the relocation and quiesce times improve about 30 %, as the number of FCTC connections is increased from one to four; if more than four FCTC connections are used, the times slightly degrade.

Conclusion

With four or more FCTC connections, no further improvements are achieved; in fact, with nine FCTC connections even a degradation of the relocation and quiesce times occurs.

z/VM 6.2 Live Guest Relocation with Linux Middleware

If one shared FICON channel with "straight" FCTC connections is used, configuring four FCTC connections for the ISFC logical link provides for shortest relocation and quiesce times. Too many FCTC connections can have a negative impact.

Note that the cause of the improvements resulting from adding the third FCTC device differs from that of adding the second FCTC device. For explanation, recall that z/VM designates one FCTC connection for unidirectional use in each direction when between two and eight FCTC connections are configured for an ISFC logical link (see *ISFC logical link configuration*):

- With the addition of the second FCTC connection, the first FCTC connection is designated for use by the receiving LPAR2, and only the second FCTC connection is used for transferring relocation data; however, now the write collisions - which inevitably occur in the one FCTC connection case - are avoided.
- With the addition of the third FCTC connection, still one FCTC connection is designated for unidirectional use, but the number of FCTC connections used for data transfer is doubled (two instead of one).

ISFC logical link configuration with two shared FICON channels and "straight" FCTC connections

Figure 8 shows an ISFC logical link configuration with two shared FICON channels and "straight" FCTC connections.

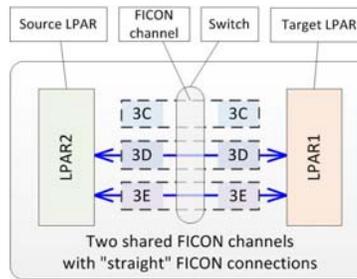


Figure 8. ISFC logical link configuration with two shared FICON channels and "straight" FCTC connections

Figure 9 shows how the relocation and the quiesce times scale in this case when the amount of FCTC connections configured for the ISFC logical link is varied.

Two shared FICON channels with "straight" FCTC connections - Relocation and quiesce times

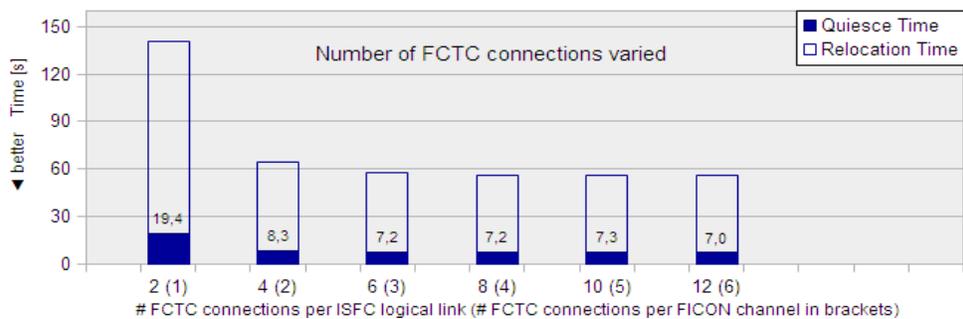


Figure 9. ISFC logical link configuration with two shared FICON channels and "straight" FCTC connections- Relocation and quiesce times

z/VM 6.2 Live Guest Relocation with Linux Middleware

Observation

There is a large improvement when scaling from two FCTC connections to four, in which case the quiesce time is reduced by more than the half. Again using more than four FCTC connections per FICON channel (with more than eight FCTC connections configured for the ISFC logical link) does not cause significant improvements.

Conclusion

Comparing the cases with four FCTC connections when one or two shared FICON channels and "straight" FCTC connections are used (compare case "4" in *Figure 7* with case "4 (2)" in *Figure 9*), it can be seen that the addition of the second FICON channel causes more than 100% improvement, with the relocation and quiesce times being more than halved.

If only two FCTC connections are configured for the ISFC logical link (one per FICON channel), the relocation and transfer times are similar to those with one shared FICON channel and "straight" FCTC connections (compare case "2" in *Figure 7* with case "2 (1)" in *Figure 9*). As already described in *ISFC logical link configuration with one shared FICON channel and "straight" FCTC connections*, this again is caused by the way z/VM designates one FCTC connection for unidirectional use (see *Avoidance of write collisions*). Consequently, when an ISFC logical link configuration with two shared FICON channels and "straight" FCTC connections is used, two or more FCTC connections per FICON channel should be configured into the ISFC logical link in order to minimize the impact of the uni-directionally used FCTC connection.

Configuring more than four FCTC connections does not cause significant improvements.

ISFC logical link configuration with three shared FICON channels and "straight" FCTC connections

Figure 10 shows an ISFC logical link configuration with three shared FICON channels and "straight" FCTC connections

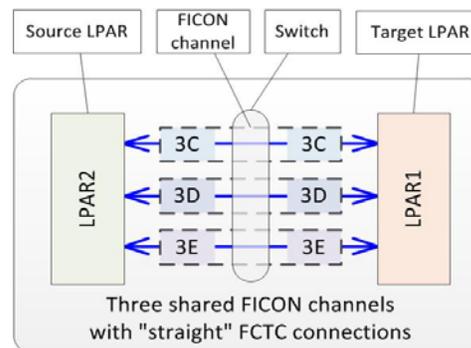


Figure 10. ISFC logical link configuration with three shared FICON channels and "straight" FCTC connections

Figure 11 shows how the relocation and the quiesce times scale in this case when the amount of FCTC connections configured for the ISFC logical link is varied.

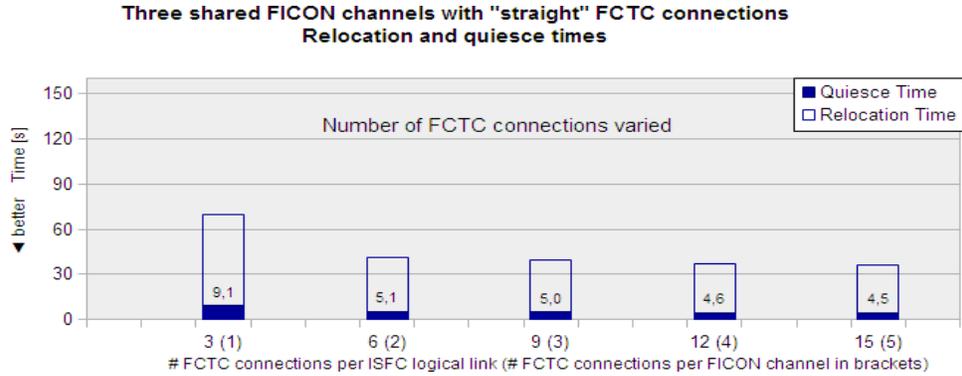


Figure 11. ISFC logical link configuration with three shared FICON channels and "straight" FCTC connections - Relocation and quiesce time

Observation

With three shared FICON channels and "straight" FCTC connections, the best result is achieved when four or five FCTC device pairs on both sides of each FICON channel are used. The relocation and quiesce times are further reduced as compared to ISFC logical link configurations with less FICON channels.

Conclusion

Comparing the times resulting from six FCTC devices with its equivalent when only one FICON channel is used (compare case "6 (2)" in Figure 11 with case "6" in Figure 7), it can be seen that the improvement for the quiesce times when using three FICON channels is in the range of 300 percent (or a reduction of 2/3).

Again, using more than four FCTC connections per FICON channel (with more than twelve FCTC connections configured for the ISFC logical link here) does not cause further improvements.

ISFC logical link configuration with two shared FICON channels and "crossed" FCTC connections

Figure 12 shows an ISFC logical link configuration with two shared FICON channels and "crossed" FCTC connections.

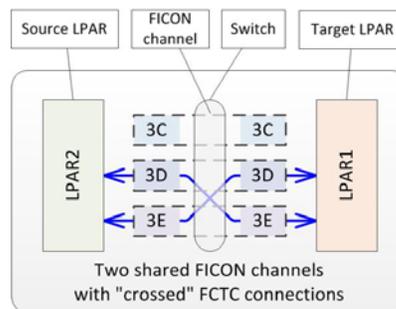


Figure 12. ISFC logical link configuration with two shared FICON channels and "crossed" FCTC connections

z/VM 6.2 Live Guest Relocation with Linux Middleware

Figure 13 shows how the relocation and the quiesce times scale in this case when the amount of FCTC connections configured for the ISFC logical link is varied.

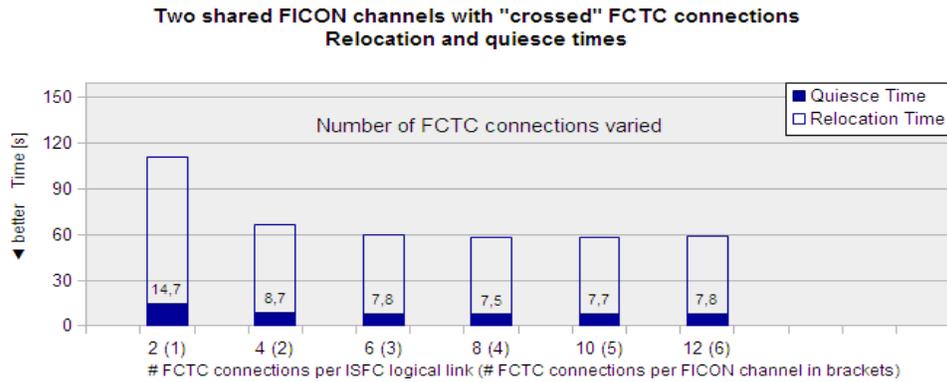


Figure 13. ISFC logical link configuration with two shared FICON channels and "crossed" FCTC connections - Relocation and quiesce times

Observation

With two shared FICON channels and "crossed" FCTC connections, the best result is achieved with four FCTC connections per FICON channel. In comparison to the configuration using two shared FICON "straight" FCTC connections (see *ISFC logical link configuration with two shared FICON channels and "straight" FCTC connections*), the relocation times are slightly inferior, and the quiesce times are mostly around 7% longer, except for the two FCTC connection case, which is around 25% shorter.

Conclusion

This result is structurally similar to that achieved by using two shared FICON channels and "straight" FCTC connections (compare *Figure 13* with *Figure 9*). However, in general, the transfer times for the crossed case are slightly higher than those achieved in the "straight" case. One possible explanation for this is that - because both LPARs were defined on the same CEC and thus share and use the same FICON channels - in the "straight" case the data flow could at least to some extent be handled by a FICON channel alone, without being passed through the FICON switch.

A noteworthy exception is that the times with two FCTC connections configured for the ISFC logical link for the "crossed" case are slightly better than those achieved in the "straight" case (see *Figure 9*). The explanation is that - while in both cases only one FCTC connection is used for relocation data transfer - in the crossed case even with only one FCTC connection configured for the ISFC logical link already two FICON channels are used: Data is sent through one FICON channel, and is received through the other FICON channel.

Again, using more than four FCTC connections per FICON channel does not cause further improvements.

z/VM 6.2 Live Guest Relocation with Linux Middleware

ISFC logical link configuration with three shared FICON channels and "crossed" FCTC connections

Figure 14 shows an ISFC logical link configuration with three shared FICON channels and "crossed" FCTC connections.

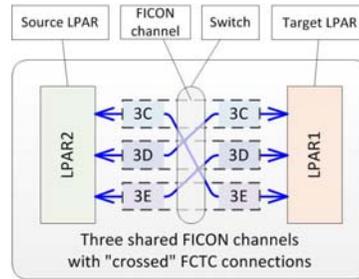


Figure 14. ISFC logical link configuration with three shared FICON channels and "crossed" FCTC connections

Figure 15 shows how the relocation and the quiesce times scale in this case when the amount of FCTC connections configured for the ISFC logical link is varied.

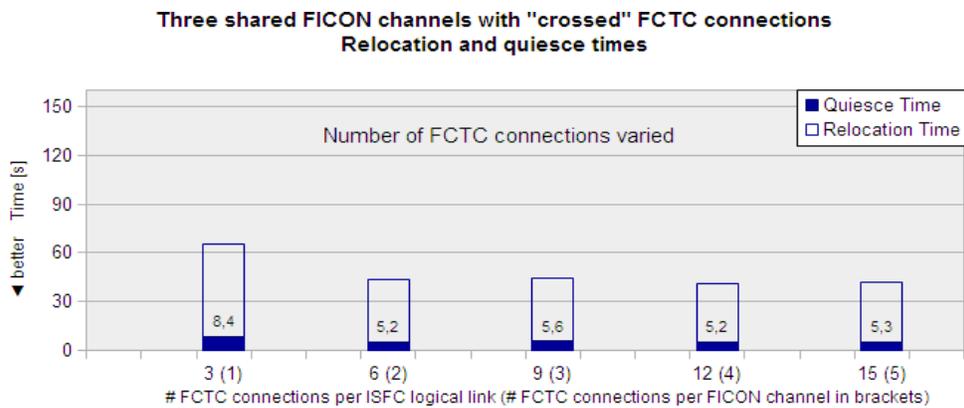


Figure 15. ISFC logical link configuration with three shared FICON channels and "crossed" FCTC connections - Relocation and quiesce times

Observation

When three FICON channels are used in a crossed manner, the shortest relocation and quiesce times are already achieved with two and with four FCTC connections per FICON channel. Compared to the corresponding "straight" setup (compare Figure 15 with Figure 11), the relocation times are about 11% longer, except for the case with one FCTC connection per FICON channel, which is 8% shorter.

Conclusion

An explanation for the generally better results of the "straight" configurations is given in *ISFC logical link configuration with two shared FICON channels and "crossed" FCTC connections*.

The absence of an improvement when adding the third FCTC connection per FICON channel probably again is caused by the way z/VM designates certain FCTC connections for unidirectional use (see *Avoidance of write collisions*).

z/VM 6.2 Live Guest Relocation with Linux Middleware

The FCTC connection usage for the cases "6 (2)" and "9 (3)" is detailed *Table 6*. The relationship between FCTC device numbers and FCTC connections can be tracked by means of *Figure 2*.

Table 6. FCTC connection usage for scenarios "6 (2)" and "9 (3)"

Configuration	FCTC connections					
	# configured per		designated for use by LPAR1	used for data transfer LPAR2 LPAR1		
	Link	CHPID		3D/3C	3E/3D	3C/3E
"6 (2)"	6	2	D010	D011	D060, D061	D080, D081
"9 (3)"	9	3	D010, D011	D012	D060, D061, D061	D080, D081, D082

As described in *Avoidance of write collisions*, z/VM dedicates one or two of the FCTC connections with the lowest device numbers for use by the system with the lexicographically lower name, which is LPAR2 in this case. With more than eight FCTC connections, two FCTC connections are designated for use by LPAR1.

Table 6 shows the effect of this approach by example of the configurations "6 (2)" and "9 (3)":

- With configuration "6 (2)", from each of the FCTC connections on FICON channels 3E/3D and 3C/3E, *two* are used for relocation data transfer
- With configuration "9 (3)", from each of the FCTC connections on FICON channels 3E/3D and 3C/3E, *three* are used for relocation data transfer
- In both cases, from the FCTC connection on FICON channels 3D/3C, only one FCTC connection is used for relocation data transfer, leaving about one 15 to 20 % of those FICON channels capacity unused.

Apparently, the additional third FCTC connection on FICON channels 3E/3D and 3C/3E did not have a positive effect.

ISFC logical link configuration with two non-shared FICON channels and "crossed" FCTC connections

Figure 16 shows an ISFC logical link configuration with two non-shared FICON channels and "crossed" FCTC connections.

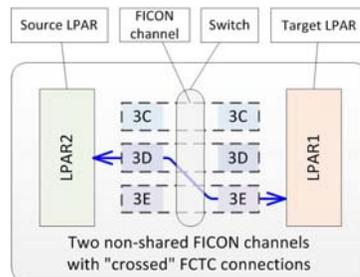


Figure 16. ISFC logical link configuration with two non-shared FICON channels and "crossed" FCTC connections

z/VM 6.2 Live Guest Relocation with Linux Middleware

Figure 17 shows how the relocation and the quiesce times scale in this case when the amount of FCTC connections configured for the ISFC logical link is varied. Recall that "non-shared" implies that each FICON channel is exclusively used by either LPAR. For the configuration shown, the FICON channel with CHPID 3D is exclusively used by LPAR2, and the FICON channel with CHPID 3E is used exclusively by LPAR1.

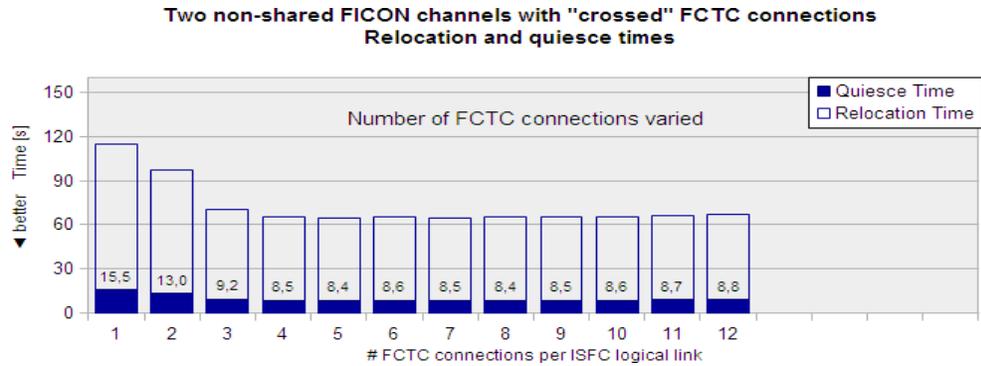


Figure 17. ISFC logical link configuration with two non-shared FICON channels and "crossed" FCTC connections - Relocation and quiesce times

Observation

In comparison to the configuration using two shared FICON channels and "crossed" FCTC connections (compare Figure 17 with Figure 13), the relocation and quiesce times are slightly better if only few FCTC connections are used, but slightly inferior if larger amounts of FCTC connections are used. More than four FCTC connections do not cause further improvement, very large number of FCTC connections even cause a slight degradation.

Conclusion

For the interpretation of this result it is important to note that each FICON channel is only used by one LPAR: The sending LPAR (LPAR2) uses one FICON channel for sending the relocation data, the receiving LPAR (LPAR1) uses the other FICON channel for receiving the data. Because each FICON channel is only used in one direction, this comprises a uni-directional usage of the FICON channels.

This is different from the corresponding crossed shared ISFC logical link configuration (see *ISFC logical link configuration with two shared FICON channels and "crossed" FCTC connections*). With that shared configuration, LPAR2 in fact uses both FICON channels for sending, and LPAR1 simultaneously uses both FICON channels for receiving. Because both FICON channels are simultaneously used for both, sending and receiving data, this comprises a bi-directional usage of the FICON channels.

Comparing the results of the shared and non-shared case with two FICON channels (compare Figure 17 with Figure 13), it can be seen that the relocation and quiesce times using the non-shared configuration are slightly inferior to those using the shared configuration (if an adequate number of FCTC connections is configured). A possible explanation might be the uni-directional versus the bi-directional use of the FICON channels. This is more closely investigated with the inspection of the related ISFC logical link data rates presented in *ISFC logical link performance*.

z/VM 6.2 Live Guest Relocation with Linux Middleware

ISFC logical link configuration with four non-shared FICON channels and "crossed" FCTC connections

Figure 18 shows an ISFC logical link configuration with four non-shared FICON channels and "crossed" FCTC connections.

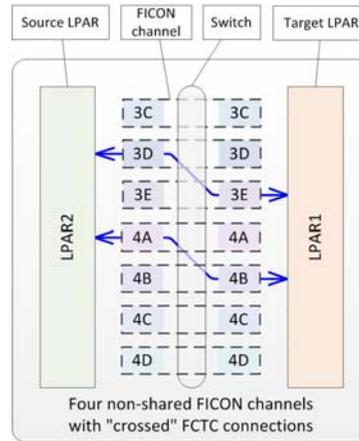


Figure 18. ISFC logical link configuration with four non-shared FICON channels and "crossed" FCTC connections

Figure 19 shows how the relocation and the quiesce times scale in this case the amount of FCTC connections configured for the ISFC logical link is varied.

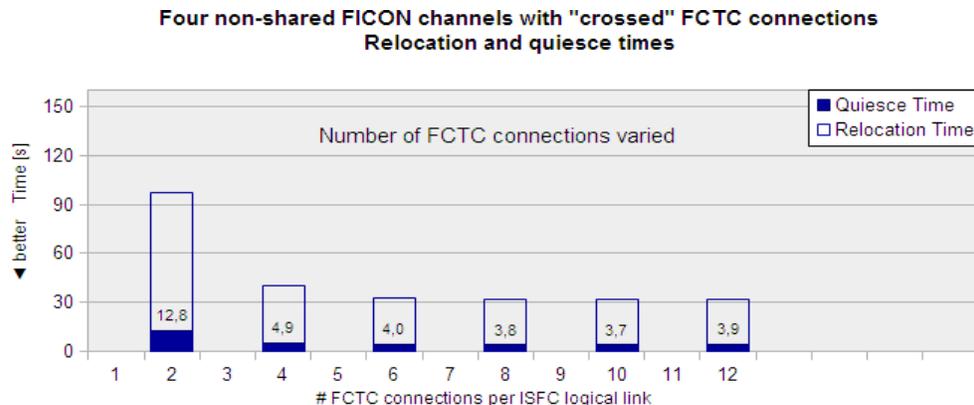


Figure 19. Four non-shared FICON channels and "crossed" FCTC connections - Relocation and quiesce times

Observation

In comparison to the ISFC logical link configuration using only two non-shared FICON channels and "crossed" FCTC connections (compare Figure 19 with Figure 17), the relocation and quiesce times are more than halved if three or more FCTC connections per FICON channel are configured.

Conclusion

As already observed for the shared ISFC logical link configurations, the relocation and quiesce times scale about inversely linear with the number of FICON channels as soon as an adequate number of FCTC connections is configured per FICON channel.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Scaling the number of FICON channels per ISFC logical link

The results so far show very clearly that the amount of FICON channels is the most important parameter to speed up the relocation process. This section addresses the question as to whether the relocation and quiesce times can be even more reduced if the amount of FICON channels is further increased.

Figure 20 shows the scaling of the relocation and quiesce times when up to seven FICON channels and "straight" FCTC connections are used.

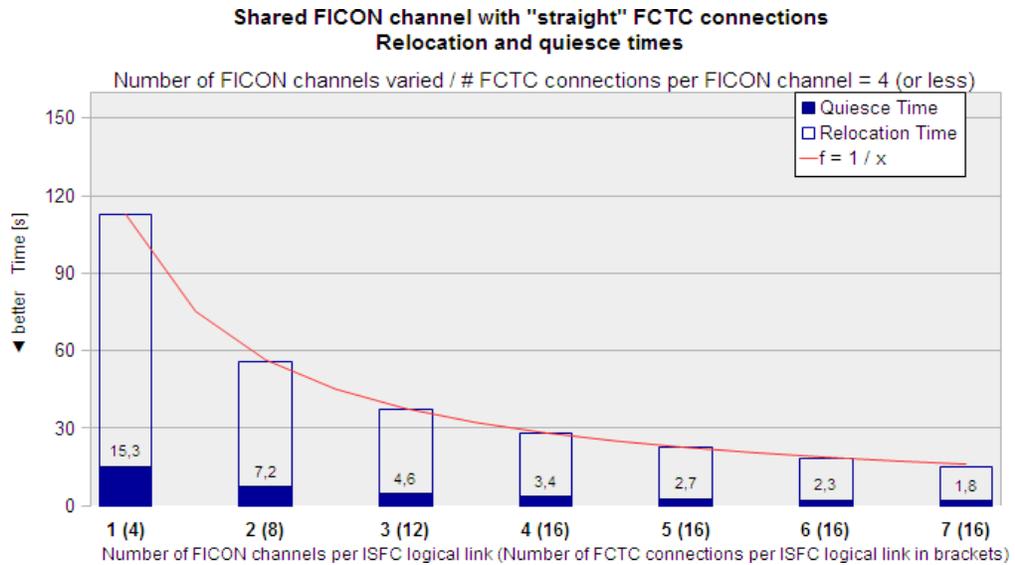


Figure 20. ISFC logical link configuration with shared FICON channels and "straight" FCTC connections- Relocation and quiesce times

For the test cases reflected in Figure 20, the number of FCTC connections per FICON channel were configured as follows:

- For configurations with four or less FICON channels, four FCTC connections per FICON channel were configured; the number of FCTC connections for the ISFC logical link overall is the sum of the FCTC connections from the FICON channels.
- With five FICON channels or more, always the maximum of sixteen FCTC connections were configured for the ISFC logical link. These were distributed over the FICON channels, as follows:
- With five FICON channels, four FICON channels were configured with three FCTC connections each, and one FICON channel was configured with four FCTC connections.
- With six FICON channels, four FICON channels were configured with three FCTC connections each, and two FICON channels were configured with two FCTC connections each.
- With seven FICON channels, five FICON channels were configured with two FCTC connections each, and two FICON channels were configured with three FCTC connections each.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Observation

The relocation as well as the quiesce time scale inversely linear with the number of FICON channels from which FCTC connections are configured into the ISFC logical link. In other words, if the (normalized) relocation time is 1 when only one FICON channel is used, it is about 1/2 with two FICON channels used, 1/3 with three FICON channels used, and so forth.

Conclusion

As long as a sufficient number of FCTC connections from each FICON channel is used, the observed inverse linear dependency can be expected. That means that the performance of the relocation process scales linear with the amount of FICON channels. For example, the relocation time when four FICON channels are used is four times faster than with one FICON channel.

Influence of striping of FCTC devices numbers over FICON channels

Figure 21 shows how the relocation and the quiesce times scale when three shared FICON channels are used, but the FCTC device numbers are assigned in a "striped" manner (see *ISFC logical link configuration* on how to assign FCTC device numbers to FICON channels in a "striped" manner). The amount of FCTC connections configured for the ISFC logical link is varied.

**Three shared FICON channels with "straight" FCTC connections and "striped" FCTC devices
Relocation and quiesce times**

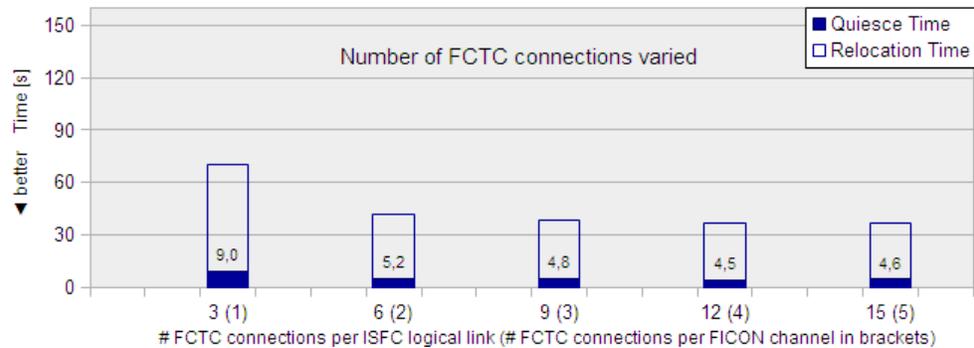


Figure 21. Three shared FICON channels and "straight" FCTC connections with "striped" FCTC devices - Relocation time and quiesce time

Observation

The relocation and quiesce time quickly converge towards a constant value as two or more FCTC connections per ISFC logical link are used.

Conclusion

When comparing the results from ISFC logical link configuration using FCTC connections with "striped" FCTC device number assignment with those of the corresponding configuration using FCTC connections with a sequential FCTC device number assignment (compare *Figure 21* with *Figure 11*), no significant impact can be observed.

However, the effects of striping FCTC device numbers become more apparent when observing the data rates achieved by an ISFC logical link. This is detailed in *Influence of striping of FCTC devices numbers over FICON channels*.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Common observations

Some observations about Linux guest performance during relocation, the number of memory passes and the z/VM system load during the relocation process.

Linux guest performance during the relocation process

Linux performance was monitored by means of the system activity data collector (sadc).

In most cases, a ten second sampling interval was used. However, in order to show the effects of the guest relocation process, for this particular test case a two second sampling interval was used. In order to make the effects of the guest relocation process most apparent, for this test case an ISFC logical link configuration was chosen that causes longer relocation and quiesce times.

Figure 22 shows the steal time percentage observed during the relocation process.

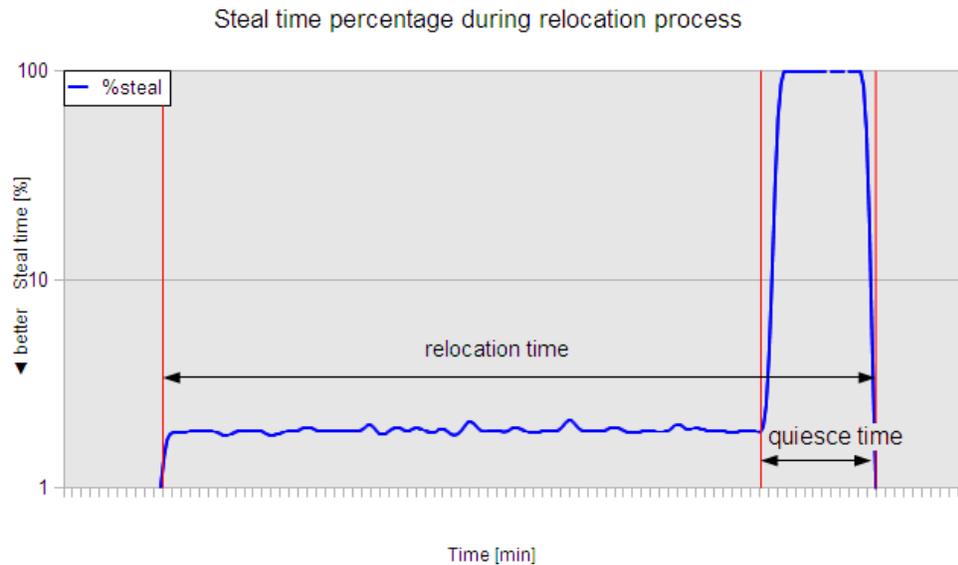


Figure 22. Steal time percentage during the relocation process

Observation

In Figure 22, a logarithmic scale was chosen for the ordinate in order to highlight the observed effects. Figure 22 shows the steal time percentage over the elapsed time for the relocation process. Initially, the steal time percentage stays at very low values (below 1%). It then rises to and stays at slightly higher values for some time (about 2 to 2.5%), and eventually rises sharply to and stays at a value of 100% for some time. Finally, the steal time percentage falls back to the initial very low value.

Conclusion

The initial constant phase reflects the situation before the live guest relocation is started. Because the Linux guest is the only workload of the source z/VM system, and because the z/VM installation has plenty resources available in terms of processing power and memory, at that time the Linux guest virtual machine receives all required resources, and is able to execute the test workload without constraints imposed by the virtualization platform.

As the live guest relocation process starts, a slightly increased steal time percentage is observed. This is because in that phase z/VM requires some resources to perform the initial memory passes of the live guest relocation.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Then, during the quiesce time, the Linux guest is no longer dispatched by z/VM, and the effective steal percentage is 100%.

After the relocation process finishes, the Linux guest now is running as the only workload of the target z/VM system, and the steal time percentage drops back to its initial very low value.

Number of memory passes

In all our tests, we observed that the number of memory passes performed by z/VM during the relocation process was *eight*.

See *Test metrics* for more information about memory passes. This is attributed to the memory access pattern of the test workload that accesses memory in a random pattern with a more or less constant rate over time, resulting in a more or less constant working set over time as well.

The observed number of eight memory passes confirms the statement made in the z/VM performance report (see *Live Guest Relocation*) that if an application is not making progress toward the quiesce threshold, or in other words, the amount of memory transferred with each memory pass does not decrease consistently, then z/VM will complete the live guest relocation in eight memory passes.

z/VM system load during the relocation process

Figure 23 shows the typical CPU usage by z/VM during the live guest relocation. In all cases shown in *Figure 23*, an optimal number of FCTC connections per FICON channel are configured. The number of FICON channels is varied from one to seven.

These values were obtained using the "Processor Activity" log (PROCLOG, FCX144) of the z/VM Performance Toolkit. In *Figure 23*, the values for CPU 0 are shown; the other CPUs exhibit similar usage patterns, but with slightly less values.

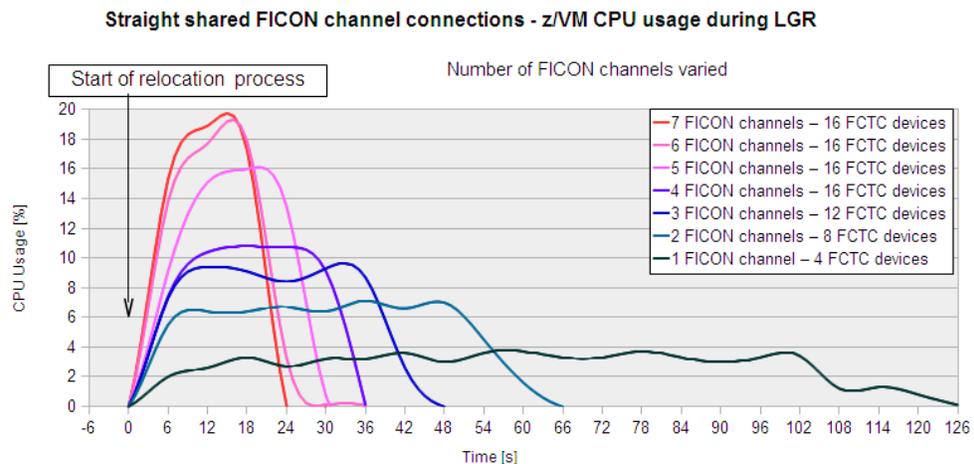


Figure 23. z/VM usage of CPU 0 during live guest relocation for various ISFC logical link configurations

Observation

The peak CPU usage afforded by z/VM for performing the live guest relocation increases with the amount of FICON channels involved; however, at the same time the relocation times decrease.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Conclusion

The area below each graph is an indicator for the overall CPU consumption spent by z/VM for the relocation process. For the configuration depicted in *Figure 23*, the largest area results when one FICON channel is used, and a relative minimum is assumed when four FICON channels are used. This might be interpreted as a hint that - in the context of the given workload and the general setup of the test environment - the four FICON channel configuration provides for the most economical approach with respect to the z/VM CPU consumption for the relocation process.

ISFC logical link performance

This section provides details about the data rates and the I/O rates observed for the ISFC logical link configurations.

ISFC logical link data rates

Figure 24 shows the maximum data rates of the ISFC logical link observed both of its sides during the guest relocation process. The maximum data rates for several ISFC logical link configurations are shown, and for each of them the number of FCTC connections is varied.

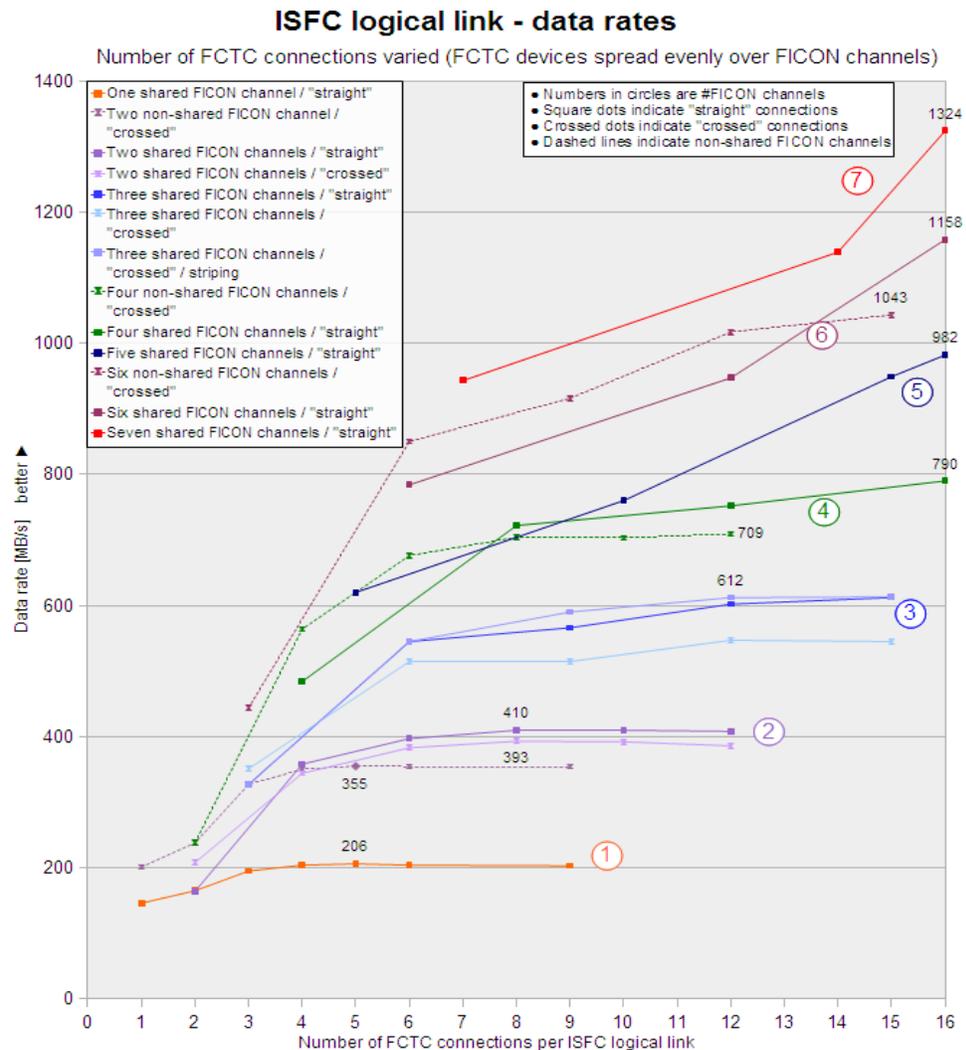


Figure 24. ISFC logical link - write data rates

Observation

For the cases where up to four FICON channels are used, the write data rates settles for a more or less constant value as soon as four or more FCTC connections are configured per FICON channel. For the cases with five and more FICON channels, the data rates continue to rise as more FCTC connections are used. However, in these cases the limit of sixteen FCTC devices for the ISFC logical link prohibits that four FCTC connections are configured for all of the used FICON channels.

Conclusion

FICON Express4 channels are specified to support 4 Gbit/s link speeds. A practical exploitation is published in the *IBM System z9[®] I/O and FICON Express4 Channel Performance* paper (see FICON Express4 Channel Performance): The results shown there were obtained under optimal conditions and show a "full-duplex" FICON channel data rate of 350 Gbit/s for a "large sequential read/write mix", and a "half-duplex" FICON channel data rate of 330 Gbit/s for a "Large sequential all reads or all writes" use-case.

Subsequently, a respective maximum *byte* data rate is approximated by dividing the specified bit rate of four Gbit/s by a factor of 10 (considering 8 bits per byte, and assuming 2 bits overhead) - yielding an approximate maximum byte data rate of 400 GB/s.

It is important to realize that for the result of all the *shared* configurations shown in *Figure 24* the FICON channel data transfer capacities are effectively shared between the sending and the receiving LPAR. In other words, for the shared configurations, every FICON channel is simultaneously used for both sending and receiving data.

This simultaneous use is assumed to be a possible reason why in these cases the maximum write byte data rates per FICON channel shown in *Figure 24* converge on a value of slightly more than 200 GB/s - which is about half of the maximum data rate per FICON channel that was previously calculated. With the shared configurations, at the same time on the receiving side a read data rate also of slightly more than 200 GB/s is achieved.

Thus, the combined read / write data rate is slightly more than 400 GB/s per FICON channel. This is a very good result that exceeds the expectations expressed in the z/VM performance report (see *ISFC Improvements*).

On the other hand, for the non-shared configurations, FICON channels are exclusively used by one of the LPARs. Also, in these cases each FICON channel is mostly used for data transfer in one primary direction. These non-shared configurations comprise a situation similar to that when the two LPARs reside on different CECs.

In our tests using the non-shared configurations, the maximum data rates per FICON channel reach values around 355 GB/s per FICON channel if one or two FICON channels are used, and a value of about 348 GB/s per FICON channel if three FICON channels are used. This about 89% (or 87%) of the maximum data rate per FICON channel that was previously calculated, and therefore is in the range of the expectation of 85% expressed in the z/VM performance report.

The ISFC logical link configuration variants with "crossed" FCTC connections achieve data rates that are up to 10% inferior to their "straight" counterparts. One possible explanation for this behavior is that with "straight" FCTC connections- because both LPARs reside on the same CEC and thus use the same FICON channels in one data transfer operation - the data flow could at least to some extent be handled by the FICON channel alone, without being passed through the switch. Regardless, also with "crossed" FCTC connections, the achieved data rates still are

z/VM 6.2 Live Guest Relocation with Linux Middleware

slightly above the expectation stated in the z/VM performance report and those expressed in the "IBM System z9 I/O and FICON Express4 Channel Performance" paper.

The slight dent observed for the performance of the non-striped variants of the ISFC logical configurations with three shared FICON channels and three to four "straight" FCTC connections (the two lower blue graphs) is caused by the way z/VM designates FCTC connections for unidirectional use; this is further substantiated in *FCTC connection I/O rates*.

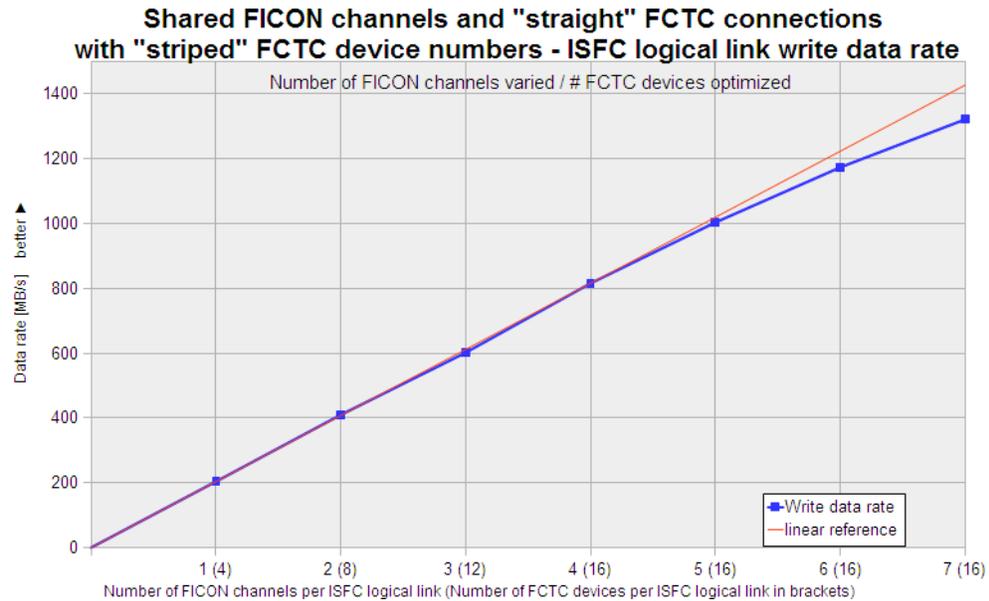


Figure 25. ISFC logical link configuration with shared FICON channels and "straight" FCTC connections with "striped" FCTC device numbers - ISFC logical link write data rate

Observation

Figure 25 shows the ISFC logical link write data rate as a function of the number of FICON channels that were used. The graph stays linear until up to four FICON channels are used, with a rate of about 204 MB/s per FICON channel. Beyond that, the graph slightly flattens.

Conclusion

Given that the same type of FICON card is used for all FCTC connections (FICON Express4 cards were used in all our test cases), and with the number of FCTC connections per FICON channel configured according to the recommendations given in *Recommendations*, the data transfer rates of the ISFC logical link scale almost linearly.

The flattening of the graph when more than four FICON channels are used is assumed to result from the limit of sixteen FCTC connections per ISFC logical link. This limit in turn limits the number of FCTC connections at least on some FICON channels to be less than four. However, if less than four FCTC connections are used on a FICON channel, that FICON channel's transport capacity is no longer fully exploited, as will be explained in more detail in *FCTC connection I/O rates*.

z/VM 6.2 Live Guest Relocation with Linux Middleware

FCTC connection I/O rates

Figure 26 shows the I/O rates per FCTC connection achieved during the relocation process for the case that two shared FICON channels and five "straight" FCTC connections are used. The FCTC connection I/O rates are measured in SSCH/s (Start Subchannel per second). The I/O rate data values were obtained using the "General I/O Device Performance" log (DEVLOG, FCX168) of the z/VM Performance Toolkit.

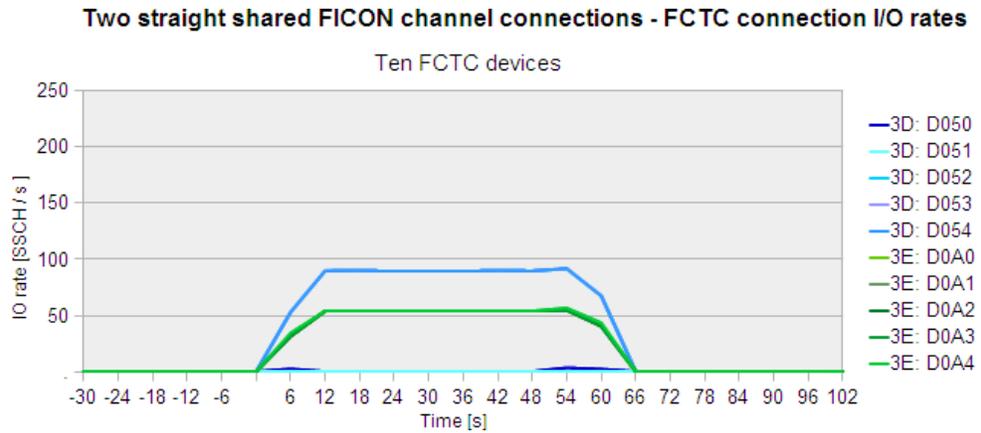


Figure 26. ISFC logical link configuration with two shared FICON channels and five "straight" FCTC connections: FCTC connection I/O rates

Observation

The first two FCTC connections (coinciding two bottom blue graphs, lowest device numbers D050 and D051) are not used for relocation data transfer in this case because of z/VM dedicating these FCTC connections for unidirectional use (see *Avoidance of write collisions*).

The remaining three FCTC connections (coinciding three topmost blue graphs, device numbers D052 to D054) on the same FICON channel (CHPID 3D) are loaded with a higher I/O rate than the FCTC connections (coinciding five middle green graphs, device numbers D0A0 to D0A4) on the other the FICON channel (CHPID 3E).

Conclusion

The relationship between the lower and the higher I/O rate per FCTC connection is inversely proportional to the number of FCTC connections effectively used (here 5:3). The average I/O rate per CHPID is 270 SSCH/s for both FICON channels (CHPID 3D: $3 * 90$ SSCH/s; CHPID 3E: $5 * 54$ SSCH/s).

For an explanation, recall that the test workload causes a CPU heavy load situation, but no shortage of memory or paging resources in both the sending and the receiving z/VM system. It appears that in this situation z/VM tries to issue I/O operations evenly across the FICON channels of which FCTC connections are configured for an ISFC logical link, regardless of how many FCTC connections are actually used for data transfer. In this case, even while the three FCTC connections on the FICON channel with CHPID 3D are loaded with a higher data rate, this does not cause a significant reduction of the overall ISFC logical link performance.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Influence of striping of FCTC devices numbers over FICON channels

Striping FCTC device numbers over FICON channels means that in the IOCDS configuration not ranges of FCTC devices are defined on a particular FICON channel, but that instead FCTC device numbers are assigned in a "striped" manner. For details, see *IOCDS configuration for FICON CTCs*.

Three shared FICON channels with three "straight" FCTC connections each: FCTC connection I/O rates

Figure 27 shows the I/O rates per FCTC connection achieved on the sending side (LPAR2) during the relocation process for the case that three shared FICON channels are used with three "straight" FCTC connections each. The FCTC device numbers are assigned in a sequential manner in this case.

In Figure 27, the FCTC connection I/O rates are measured in SSCH/s (Start Subchannel per second), using the left scale. For each FICON channel a different color was used, as follows:

- Orange was used for the data rates of the FCTC connections defined on the FICON channel with CHPID 3C
- Green was used for the data rates of the FCTC connections defined on the FICON channel with CHPID 3D
- Blue was used for the data rates of the FCTC connections defined on the FICON channel with CHPID 3E

In addition, Figure 27 shows the link data rate (LDR) for the complete ISFC logical link as a dashed black line. The link data rate is measured in MB/s, using the right scale.

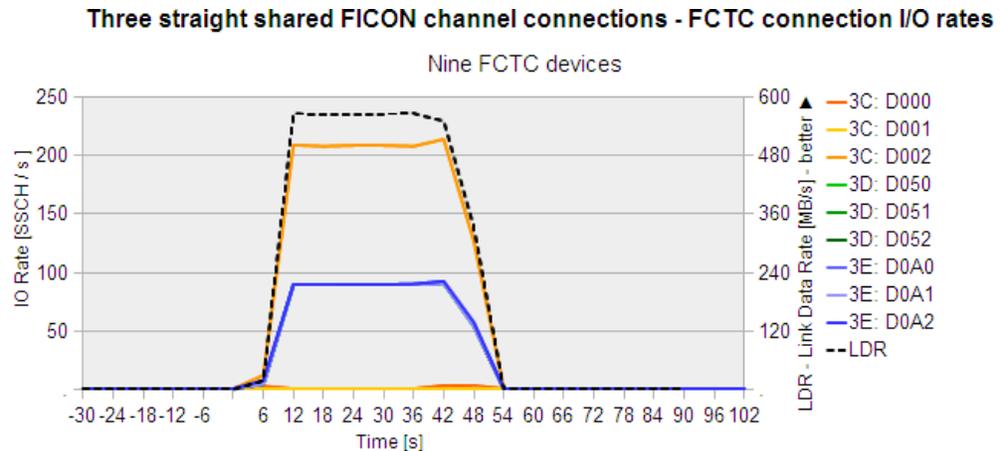


Figure 27. Three shared FICON channels with three "straight" FCTC connections each: FCTC connection I/O rates

Observation

The FCTC connections with the lowest device numbers - D000 and D001, both on the FICON channel with CHPID 3C - are designated by z/VM for unidirectional use by receiving LPAR1. For these FCTC connections only very low I/O rates are observed (coinciding bottom red and yellow graphs). While not shown in Figure 27, the device busy percentages in these cases stay at 0%.

The highest I/O rate with 214 SSCH/s is observed for FCTC connection with device number D002 (topmost orange graph); while exhibiting that high value, the device busy percentage (not

z/VM 6.2 Live Guest Relocation with Linux Middleware

shown) for this FCTC connection stays at about 100%. Because the FCTC connection with device number D002 is the only connection on the FICON channel with CHPID 3C, the channel I/O rate for that FICON channel is also 214 SSCH/s here.

For the FCTC connections on the FICON channels with CHPIDs 3D and 3E, in all cases an I/O rate of about 90 SSCH/s is observed (coinciding middle green and blue graphs). For each of these FCTC connections a device busy percentage of about 100% (not shown). For the FICON channels with CHPIDs 3D and 3E the channel I/O rate is 270 SSCH/s each ($3 * 90$ SSCH/s).

The added I/O rate for the ISFC link is 754 SSCH/s (214 SSCH/s + $2 * 270$ SSCH/s). In addition, *Figure 27* shows that during the relocation process the write data rate for the ISFC logical link (black dashed graph) almost constantly stays at about 565 MB/s.

Three shared FICON channels with three FCTC connections each and "striped" FCTC device numbers

Figure 28 also shows the I/O rates per FCTC connection achieved on the sending side (LPAR2) during the relocation process for the case that three FICON channels are used with 3 FCTC connections each; however, here the device numbers were assigned in the IOCDS in a "striped" manner.

In *Figure 28*, the same color assignment as in *Figure 27* is used (3C: orange; 3D: green; 3E: blue). However, because here the device numbers were assigned in a "striped" manner, the device numbers corresponding to one FICON channel are not arranged in contiguous ranges; instead, every third FCTC device corresponds to the same FICON channel.

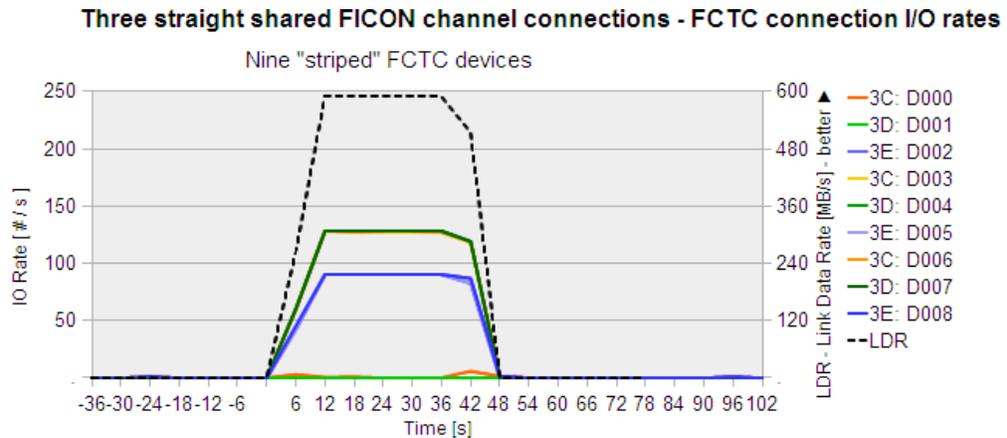


Figure 28. Three shared FICON channels with three FCTC connections each and "striped" FCTC devices: FCTC connection I/O rates

Observation

Again the FCTC connections with the lowest device numbers - D000 and D001 - are designated for unidirectional use by receiving LPAR1. But because of the "striped" device number assignment in the IOCDS now on different FICON channels (with CHPIDs 3C and 3D). For these FCTC connections only very low I/O rates are observed (bottom coinciding orange and green graphs), along with a (not shown) device busy rate of 0%. But as a result of the "striped" device number assignment, the designated FCTC connections now are on two different FICON channels.

z/VM 6.2 Live Guest Relocation with Linux Middleware

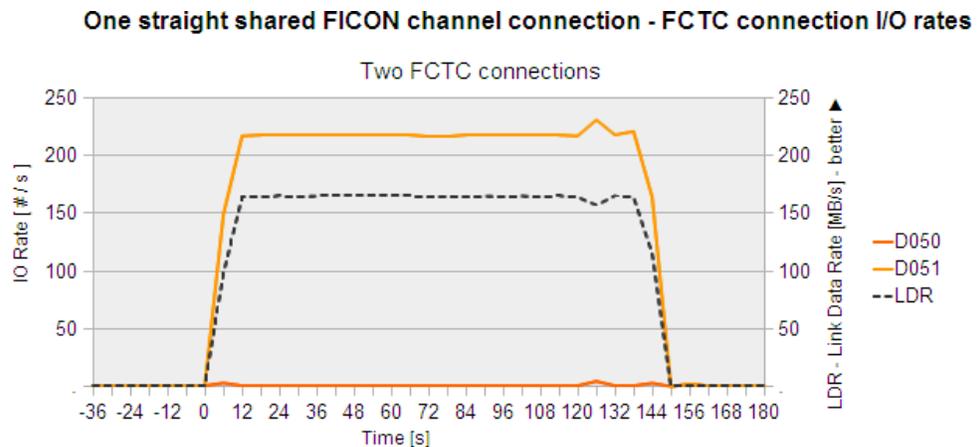
For the remaining FCTC connections on these FICON channels (D003 and D006 on the FICON channel with CHPID 3C, D004 and D007 on the FICON channel with CHPID 3D, topmost coinciding orange and green graphs) an I/O rate of about 128 SSCH/s (along with a 100% device busy rate) is observed. In sum, the channel I/O rate for the FICON channels with CHPIDs 3C and 3D is 256 SSH/s here.

For the three FCTC connections on the FICON channel with CHPID 3E (device numbers D002, D005 and D008, middle coinciding blue graphs) an I/O rate of about 90 SSCH/s is observed (along with a 100% busy rate). In sum, the channel I/O rate for the FICON channel with CHPID 3E is 270 SSCH/s here.

The added I/O rate for the ISFC link is 782 SSCH/s ($2 * 256 \text{ SSCH/s} + 270 \text{ SSCH/s}$), which is about 3.7% higher than for the non-striped case shown in *Figure 27*. In addition, *Figure 28* shows that the write data rate for the ISFC logical link (black dashed graph) almost constantly stays at about 590 MB/s during the relocation process, which is about 4.4% higher than for the non-striped case.

One shared FICON channel with two FCTC connections: FCTC connection I/O rates

For comparison, *Figure 29* shows the I/O rates per FCTC connection achieved during the relocation process for the case that one FICON channel is used with two FCTC connections.



**Figure 29. One shared FICON channel with two FCTC connections:
FCTC connection I/O rates**

Observation

The FCTC connection with the lower device number exhibits a very low device I/O rate in this case. In addition, *Figure 29* shows that while I/O is performed the write data rate for the ISFC logical link (black dashed graph) almost constantly stays at about 165 MB/s.

Conclusions from influence of striping of FCTC devices numbers over FICON channels

Figure 27 shows the case of an ISFC logical link configuration using shared FICON channels and "straight" FCTC connections with sequentially assigned device numbers. The two FCTC connections with the lowest device numbers (D000 and D001, lower orange graphs) are not used for relocation data transfer because of the way z/VM designates certain FCTC connections for unidirectional use (see *Avoidance of write collisions*). Due to the sequential device number assignment, both of these FCTC connections are from the same FICON channel with CHPID 3C.

z/VM 6.2 Live Guest Relocation with Linux Middleware

The remaining only FCTC connection (device number D002) on the FICON channel with CHPID 3C is loaded with a significantly higher I/O rate than the FCTC connections on the other FICON channels (CHPIDs 3D and 3E). In fact, this single FCTC connection I/O rate is similar to that shown in *Figure 29*. There, one FCTC connection (D051) of a two FCTC connection / single FICON channel configuration is used for the relocation data transfer, while the other FCTC connection (device number D050) is designated for use by the communication partner by z/VM. In other words, only one FCTC connection on that FICON channel is used for relocation data transfer. *Figure 29* also conveys that the data rate of the ISFC logical link in this situation is about 165 MB/s. This is about 20% less than the maximal possible data transfer rate per FICON channel of about 206 MB/s that is achievable with one FICON channel if sufficient FCTC connections are used (see *Figure 24*). Apparently, the use of only one FCTC connection from a FICON channel for data transfer prevents exploiting all of the usable bandwidth from that FICON channel.

Figure 28 shows the case of an ISFC logical link configuration also using shared FICON channels with "straight" FCTC connections, but with device numbers assigned in a striped manner. Again the two FCTC connections with the lowest device numbers (D000 and D001, lower orange and lower green graph) are not used for relocation data transfer. However, with a "striped" device number assignment in the IOCDS, now the two "unidirectional" connections are from two different FICON channels. With two FCTC connections from each of these FICON channels now being used for data transfer, hitting the "one connection bandwidth limit" as observed in the non-striped case is attenuated. For the ISFC logical link overall, a higher throughput results than in the non-striped case.

In summary, for an ISFC logical link configuration with shared FICON channels and "straight" FCTC connections where three FCTC connections with contiguous device numbers are configured per FICON channel, the first two FCTC connections are designated for unidirectional use by z/VM, causing the first FICON channels data transfer capacity not being fully used for data transfer. It was shown that a better ISFC logical link throughput can be achieved by distributing the "unidirectional" FCTC connections on two different FICON channels, because then the remaining FCTC connections on these FICON channels are used for data transfer.

Seven FICON channels with sixteen "straight" FCTC connections: FCTC connection I/O rates

Figure 30 shows the I/O rates per FCTC connection achieved during the relocation process for the case that ISFC logical link configuration with sixteen "straight" FCTC connections on seven FICON channels is used. Four FCTC connections are configured on the FICON channel with CHPID 3C; two FCTC connections are configured on each of the other six FICON channels. The orange and red graphs show the I/O rates for the four FCTC connections on the FICON channel with CHPID 3C; two of them are the lowest graphs, and the other two are the highest graphs. The I/O rates of all other FCTC connections are shown in blue.

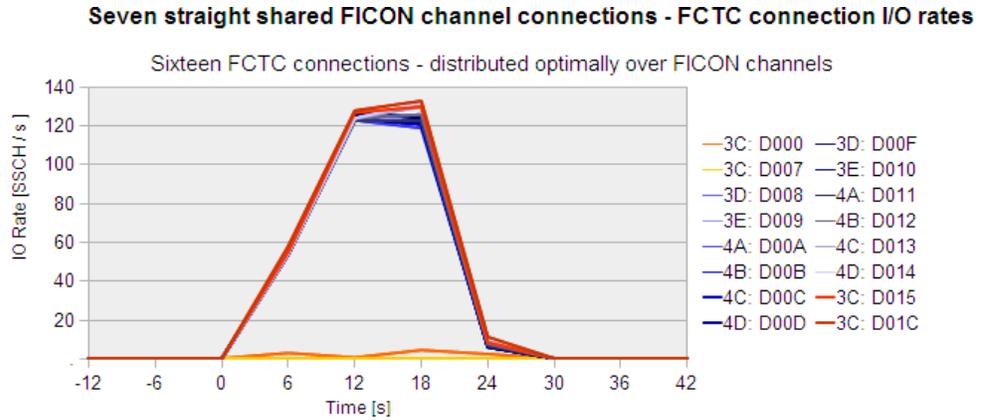


Figure 30. ISFC logical link configuration with seven shared FICON channels and "straight" FCTC connections - FCTC connection I/O rates

Observation

The I/O rates for the FCTC devices corresponding to the FICON channel with CHPID 3C exhibit a bi-fold behavior: The FCTC connections with device numbers D000 and D007 are almost not used (bottom orange and yellow graph); the FCTC connections with device numbers D015 and D01C (topmost red graphs) are used with a similar I/O rate than the FCTC connections corresponding to the other FICON channels.

Conclusion

The first two FCTC connections with device numbers D000 and D007 (lower orange graphs, lowest device numbers) are not used for relocation data transfer from BOER3740 to BOER3739 because of the way z/VM dedicates certain FCTC connections for unidirectional use (see *Avoidance of write collisions*).

For the ISFC logical link configuration used for this example, the FCTC connections with the two highest and the two lowest device numbers were selected from the same FICON channel (CHPID 3C), resulting in a symmetrical ISFC logical link configuration; the remaining FCTC connections were selected from the other six FICON channels, and with device numbers within the range in between.

As a consequence, the four FCTC connections on the FICON channel with CHPID 3C are designated by z/VM for unidirectional use (two in either direction). For the relocation process this ISFC logical link configuration results in an almost uniform distribution of I/O operations over fourteen FCTC connections and seven FICON channels, with each FICON channel being used with two FCTC connections.

This ISFC logical link configuration yielded the most optimal relocation process achieved in our project. This is caused by uniform distribution of the I/O load over all available FICON channels, including the FICON channel on which the FCTC connections designated for unidirectional use are assigned.

FCTC connection I/O rates for non-shared ISFC logical link configurations

This section shows FCTC I/O rates for one, two and three non-shared FICON channels.

One non-shared FICON channels with twelve FCTC connections each

z/VM 6.2 Live Guest Relocation with Linux Middleware

Figure 31 shows the I/O rates per FCTC connection for the one FICON channel that is used by the sending LPAR2. The receiving LPAR1 uses a different FICON channel. The color coding for FICON channel with CHPID 3E is orange.

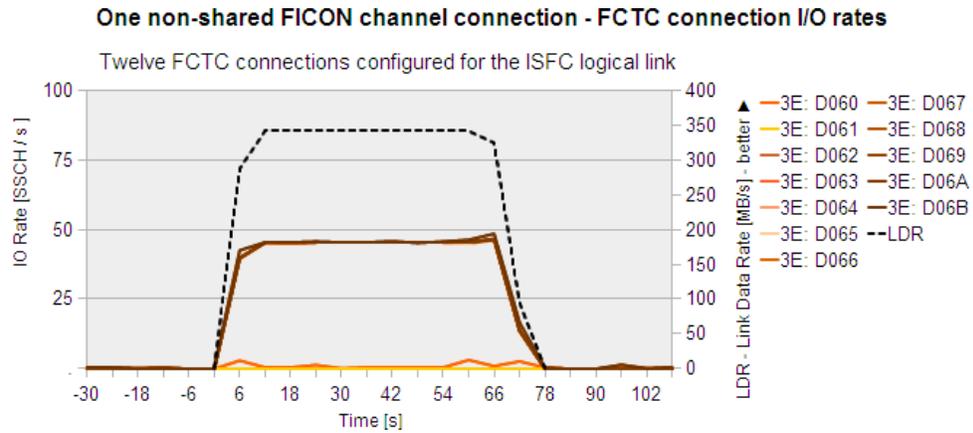


Figure 31. One non-shared FICON channels with twelve FCTC connections each:
FCTC connection I/O rates

Observation

Two FCTC connections are designated by z/VM for unidirectional use (lower orange graphs). The other FCTC connections uniformly exhibit I/O rates of about 45 SSCH/s (coinciding orange graphs), resulting in a combined I/O rate for the ISFC logical link of about 450 SSCH/s ($10 * 45$ SSCH/s). The link data rate is about 342 MB/s.

Two non-shared FICON channels with six FCTC connections each

Figure 32 shows the I/O rates per FCTC connection for the two FICON channels that are used by the sending LPAR2. The receiving LPAR1 uses two different FICON channels. The color coding for FICON channel with CHPID 3E is orange, that for the FICON channel with CHPID 4C is blue.

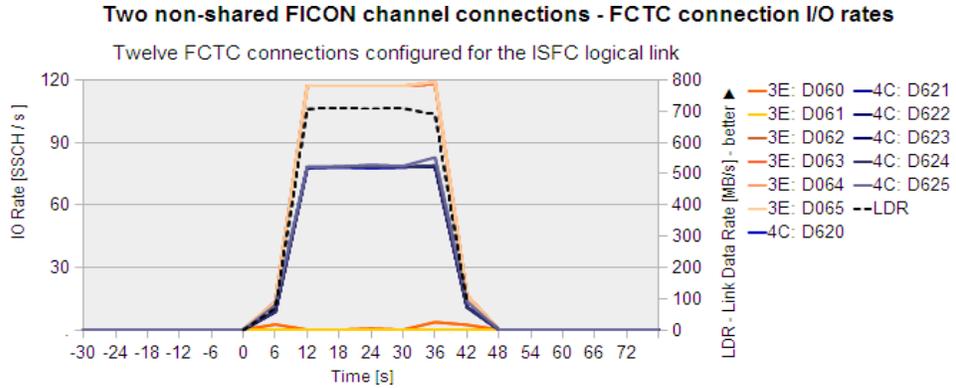


Figure 32. Two non-shared FICON channels with six FCTC connections each: FCTC connection I/O rates

Observation

Two FCTC connections are designated by z/VM for unidirectional use (lower orange graphs), both on the FICON channel with CHPID 3E. The other FCTC connections on that FICON channel uniformly exhibit I/O rates of about 117 SSCH/s each (coinciding topmost orange graphs), resulting in a combined I/O rate for the FICON channel with CHPID 3E of about 468 SSCH/s ($4 * 117 \text{ SSCH/s}$). The FCTC connections on the FICON channel with CHPID 4C uniformly exhibit I/O rates of about 78 SSCH/s each (coinciding middle blue graphs), resulting in a combined I/O rate for the FICON channel with CHPID 4C of about 468 SSCH/s ($6 * 78 \text{ SSCH/s}$). The combined I/O rate of the ISFC logical link is 936 SSCH/s, which is a little bit more than twice the I/O rate achieved with only one non-shared FICON channel. The link data rate is about 708 MB/s, which again is a little bit more than twice the data rate achieved with only one non-shared FICON channel.

Three non-shared FICON channels

Figure 33 shows the I/O rates per connection for the three FICON channels that are used by the sending LPAR2. The receiving LPAR1 uses three different FICON channels. The color coding for FICON channel with CHPID 3E is orange, that for the FICON channel with CHPID 4C is blue, and that for the FICON channel with CHPID 4D is green.

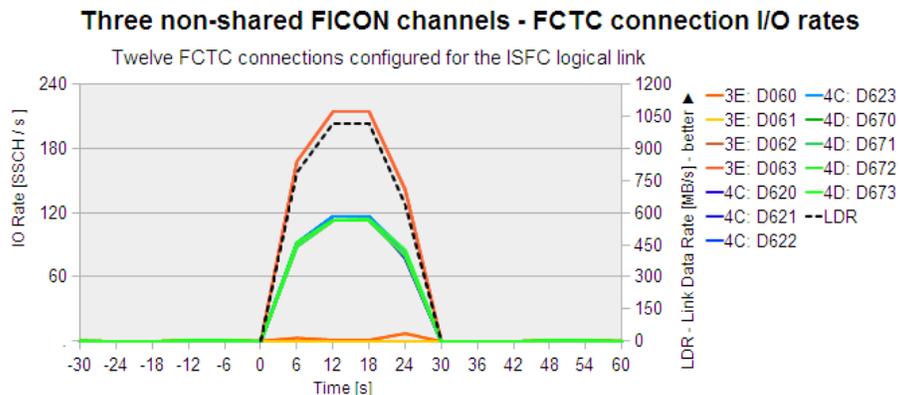


Figure 33. Three non-shared FICON channels: FCTC connection I/O rates

z/VM 6.2 Live Guest Relocation with Linux Middleware

Observation

Two FCTC connections are designated by z/VM for unidirectional use (lower orange graphs), both on the FICON channel with CHPID 3E. The other FCTC connections on that FICON channel uniformly exhibit I/O rates of about 214 SSCH/s (coinciding middle blue graphs), resulting in a combined I/O rate for the FICON channel with CHPID 3E of about 428 SSCH/s ($2 * 214$ SSCH/s). The FCTC connections on the FICON channels with CHPID 4C and 4D (coinciding middle blue and green graphs) exhibit I/O rates of about 115 SSCH/s each (coinciding middle blue graphs), resulting in a combined I/O rate for each of the FICON channel with CHPID 4C and 4D of about 460 SSCH/s ($4 * 115$ SSCH/s). The combined I/O rate of the ISFC logical link is 1348 SSCH/s, which is about three times the I/O rate achieved with only one non-shared FICON channel. The link data rate is about 1017 MB/s, which is a little bit more than three times the data rate achieved with only one non-shared FICON channel.

Conclusion

As with their shared counterparts, the I/O rates and data rates for non-shared ISFC logical link configurations scale linearly with the amount of FICON channels in use. The effects resulting from some FCTC connections being designated for unidirectional use by z/VM are similar.

Given a fixed amount of FICON channels being used for an ISFC logical link, the data rates for a "remote" link (the z/VM members reside in LPARs on *different* CECs), are almost double as high as those for a "local" link (the z/VM members reside in LPARs on the *same* CEC).

For example, compare *Figure 33* (non-shared, three FICON channels) with *Figure 28* (shared, three FICON channels): In both cases, three FICON channels are used for the ISFC logical link configuration. The averaged data rate per FICON channel is about 339 MB/s ($1017/3$ MB/s) for the non-shared, send only case, and averaged send data rate per FICON channel for the shared case is 197 MB/s ($595/3$ MB/s).

Possible reasons for the non-shared case not reaching twice the data rate of the shared case might be

- For the configuration with shared FICON channels and "straight" FCTC connections, that the FICON channels might optimize at least some of the data transfer by transferring data directly and not passing it through the switch
- That the FICON channels in the non-shared case are used almost only in one direction (send only), while in the shared case the FICON channels are used equally in either direction (send and receive).

Insofar, the results observed here are similar to those published in the "IBM System z9 I/O and FICON Express4 Channel Performance" paper (see FICON Express4 Channel Performance). There a data rate of 350 MB/s per FICON channel was observed for a use-case exhibiting a "large sequential read/write mix" (bi-directional), as opposed to 330 MB/s per FICON channel for a use-case exhibiting "Large sequential all reads or all writes" (uni-directional).

z/VM 6.2 Live Guest Relocation with Linux Middleware

Recommendations

For the configuration of an ISFC logical link, in general, we confirm the recommendations of the z/VM performance report (see ISFC Improvements) which states that four FCTC devices per FICON channel should be configured.

However, in addition, the effects resulting from z/VM's convention from using certain FCTC devices uni-directionally (see *Avoidance of write collisions*) should be taken into consideration.

For that reason, we extend the recommendation of the z/VM performance report, and recommend using the following procedure to determine the set of FCTC devices used for an ISFC logical link:

1. Determine NC, the number of FICON channels that are available for being used in context of the ISFC logical link
2. Determine NDC, the number of devices per FICON channel to be used, as follows:
 - Initially, assume $NDC = 4$
 - If $NDC * NC > 14$, reduce NDC such that $NDC * NC \leq 14$
3. Configure the IOCDS for the FCTCs, as follows:
 - Select one FICON channel to hold the "unidirectional" devices
 - Determine a device number range that is large enough to provide device numbers for all needed FCTC devices. Unless device number constraints exist in your installation, consider 16 devices for each "normal" FICON channel, and two times 16 devices for the FICON channel selected to hold the "unidirectional" devices. While only few of all these FCTC devices will actually be used for the ISFC logical link, the subdivision into subranges provides for a better structure of the IOCDS configuration.
 - For the unidirectional FICON channel, configure two ranges of FCTC devices, with one "low" range at the start, and one "high" range at the end of the determined device number range
 - For each of the other FICON channels, configure one range of FCTC devices, with device numbers from within the determined device number range
4. In z/VM, configure the ISFC logical link, as follows:
 - Configure 2 FCTC devices from the "low" range(s) configured in step 3.
 - Configure NDC FCTC devices for each of the other FICON channels
 - Configure 2 FCTC devices from the "high" range(s) configured in step 3.
 - If NDC is less than four, and not yet sixteen FCTC devices are configured, add additional FCTC devices in a symmetrical manner, until sixteen FCTC devices are configured.

As an example, consider the use of four FICON channels with CHPIDs 4A, 4B, 4C and 4D:

- Select the FICON channel with CHPID 4A to hold the unidirectional devices
- Determine NDC as 3 (because $3 * 4 = 12 < 14$)
- In the IOCDS, for example define FCTC devices as follows:
 - D000 to D00F on 4A
 - D010 to D01F on 4B
 - D020 to D02F on 4C
 - D030 to D03F on 4D
 - D040 to D04F on 4A
- Configure the ISFC logical link, in a first step with 3 FCTC devices per FICON channel:

```
D000 D001
D010 D011 D012
D020 D021 D022
D030 D031 D032
D040 D041
```

z/VM 6.2 Live Guest Relocation with Linux Middleware

- Because not yet sixteen FCTC devices are configured, and for some FICON channels less than four FCTC devices are usable in either direction, add more FCTC devices in a symmetrical and equal manner:

```
D000 D001 D002
D010 D011 D012
D020 D021 D022 D023
D030 D031 D032
D040 D041 D042
```

- With this ISFC logical link configuration, in either direction four FCTC devices are available data transfer from each of the FICON channels with CHPIDs 4A and 4C, while three FCTC devices are available from each of the FICON channels with CHPIDs 4B and 4D. The device numbers in *italics* are the newly added ones, the underlined devices numbers are those of the FCTC devices selected by z/VM for unidirectional use.

As another example, consider the use of two FICON channels with CHPIDs 4A and 4B:

- Select 4A to hold the unidirectional devices
- Determine NDC as 4 (because $4 * 2 = 8 < 14$)
- In the IOCDs, for example define FCTC devices as follows:
- D000 to D00F on 4A
- D010 to D01F on 4B
- D020 to D02F on 4A
- Configure the ISFC logical link, in a first step with 4 FCTC devices per FICON channel:

```
D000 D001
D010 D011 D012 D013
D020 D021
```

- Because not yet sixteen FCTC devices are configured, and for some FICON channels less than four FCTC devices are usable in either direction, add more FCTC devices in a symmetrical and equal manner:

```
D000 D001 D002
D010 D011 D012 D013
D020 D021 D022
```

- With this ISFC logical link configuration, in either direction four FCTC devices are available for data transfer from each of FICON channels 4A and 4B. The device numbers in *italics* are newly added, the underlined devices numbers are those of the FCTC devices selected by z/VM for unidirectional use.

Another recommendation applies if the connected LPARs reside on the same CEC, and FCTC devices from more than one FICON channel are configured for the connecting ISFC logical link. In this case n ISFC logical link configuration with "straight" (see *ISFC logical link configurations with "straight" FCTC connections*) or "crossed" (see *ISFC logical link configurations with "crossed" FCTC connections*) FCTC connections is possible in the IOCDs. Because our measurements show that "straight" FCTC connections result in slightly better transfer times and data rates than "crossed" FCTC connections, our recommendation is to use "straight" FCTC connections.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Appendix. IOCDs example

```
ID      SYSTEM=(2817,1),LSYSTEM=SYS1, *
      TOK=('SY1',...,'12-08-10','hh:mm:ss','SYS4','IODFxx')
RESOURCE PARTITION=((LPAR1,1),(LPAR2,2)))
CHPID  PATH=(CSS(2),3C),SHARED,SWITCH=ss,PCHID=pc1,TYPE=FC, *
      PARTITION=((CSS(2),(LPAR1,LPAR2),=))
CHPID  PATH=(CSS(2),3D),SHARED,SWITCH=ss,PCHID=pc2,TYPE=FC, *
      PARTITION=((CSS(2),(LPAR1,LPAR2),=))
CHPID  PATH=(CSS(2),3E),SHARED,SWITCH=ss,PCHID=pc3,TYPE=FC, *
      PARTITION=((CSS(2),(LPAR1,LPAR2),=))
CHPID  PATH=(CSS(2),3F),SHARED,SWITCH=ss,PCHID=pc4,TYPE=FC, *
      PARTITION=((CSS(2),(LPAR1,LPAR2),=))
*
*
* 16 "STRAIGHT" FCTC connections between LPAR1 and LPAR2 on CHPID 3C
*
* CNTLUNIT D000 on CHPID 3C "points to" LPAR2 (LINK=18, CUADD=2)
* CNTLUNIT CUNUMBR=D000,PATH=((CSS(2),3C)),UNITADD=((00,256)), *
*   LINK=((CSS(2),18)),CUADD=2,UNIT=FCTC
*
* DEVICES D000-D00F on LPAR1 are on D000 (which "points to" LPAR2)
* IODEVICE ADDRESS=(D000,016),CUNUMBR=(D000),STADET=Y, *
*   PARTITION=((CSS(2),LPAR1)),UNIT=FCTC
*
* CNTLUNIT D001 also on CHPID 3C "points to" LPAR1 (LINK=18, CUADD=1)
* CNTLUNIT CUNUMBR=D001,PATH=((CSS(2),3C)),UNITADD=((00,256)), *
*   LINK=((CSS(2),18)),CUADD=1,UNIT=FCTC
*
* DEVICES D000-D00F on LPAR2 are on D001 (which "points to" LPAR1)
* IODEVICE ADDRESS=(D000,016),CUNUMBR=(D001),STADET=Y, *
*   PARTITION=((CSS(2),LPAR2)),UNIT=FCTC
*
*
* 16 "CROSSED" FCTC connections between LPAR1 and LPAR2
* on CHPIDs 3C and 3D
*
* CNTLUNIT D010 on CHPID 3C "points to" CHPID 3D/LPAR2 (LINK=19, CUADD=2)
* CNTLUNIT CUNUMBR=D010,PATH=((CSS(2),3C)),UNITADD=((00,256)), *
*   LINK=((CSS(2),19)),CUADD=2,UNIT=FCTC
*
* DEVICES D010-D01F on LPAR1 are on D010 (which "points to" LPAR2)
* IODEVICE ADDRESS=(D010,016),CUNUMBR=(D010),STADET=Y, *
*   PARTITION=((CSS(2),LPAR1)),UNIT=FCTC
*
* CNTLUNIT D011 on CHPID 3D "points to" CHPID 3C/LPAR1 (LINK=18, CUADD=1)
* CNTLUNIT CUNUMBR=D011,PATH=((CSS(2),3C)),UNITADD=((00,256)), *
*   LINK=((CSS(2),18)),CUADD=1,UNIT=FCTC
*
* DEVICES D010-D01F on LPAR2 are on D011 (which "points to" LPAR1)
* IODEVICE ADDRESS=(D010,016),CUNUMBR=(D011),STADET=Y, *
*   PARTITION=((CSS(2),LPAR2)),UNIT=FCTC
```

z/VM 6.2 Live Guest Relocation with Linux Middleware

References

- *IBM z/VM CP Planning and Administration*, SC24-6178-02:
<http://publibz.boulder.ibm.com/epubs/pdf/hcsg0c10.pdf>
- *IBM z/VM Getting Started with Linux on System z*, SC24-6194-01:
<http://publibz.boulder.ibm.com/epubs/pdf/hcsx0c10.pdf>
- *IBM System z, ESCON and FICON Channel-to-Channel Reference*, SB10-7034-05:
ibm.com/support/docview.wss?uid=isg29fafd2510f68d17285256ced0051213e&aid=1
- Live Guest Relocation, IBM Web publication *z/VM Performance Report*
<http://www.vm.ibm.com/perf/reports/zvm/html/620lgr.html>
- ISFC Improvements IBM Web publication *z/VM Performance Report*
<http://www.vm.ibm.com/perf/reports/zvm/html/620isfc.html>
- FICON Express4 Channel Performance IBM Web publication *IBM System z9 I/O and FICON Express4 Channel Performance*
<ftp://ftp.software.ibm.com/common/ssi/sa/wh/n/zsw03005usen/ZSW03005USEN.PDF>

z/VM 6.2 Live Guest Relocation with Linux Middleware

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 grant 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.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

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.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

z/VM 6.2 Live Guest Relocation with Linux Middleware

IBM Corporation
Software Interoperability Coordinator, Department 49XA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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 is for planning purposes only. The information herein is subject to change before the products described become available.

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.

z/VM 6.2 Live Guest Relocation with Linux Middleware

Terms and conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

Personal Use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of the manufacturer.

Commercial Use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of the manufacturer.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any data, software or other intellectual property contained therein.

The manufacturer reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by the manufacturer, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

THE MANUFACTURER MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THESE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

z/VM 6.2 Live Guest Relocation with Linux Middleware



® Copyright IBM Corporation 2013
IBM Systems and Technology Group
Route 100
Somers, New York 10589
U.S.A.
Produced in the United States of America,
02/2013

IBM, IBM logo, ESCON, FICON, System Storage, System z System z9, WebSphere, zEnterprise, and z/VM are trademarks or registered trademarks of the International Business Machines Corporation.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

InfiniBand and InfiniBand Trade Association are registered trademarks of the InfiniBand Trade Association. Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

ZSW3247-USEN-00