



z/OS Parallel Sysplex Update

Session 8694
Mark A. Brooks
mabrook@us.ibm.com



In this session, the speaker will provide updates on Parallel Sysplex, including the latest hardware (zEnterprise 196) and coupling link technology (infiniband), Coupling Facility Control Code (CFCC Level 17), and recent z/OS enhancements (z/OS 1.12) and a preview of things to come (z/OS 1.13).



Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

IBM*	FICON*	System x*
IBM (logo)*	IMS	System z*
ibm.com*	Parallel Sysplex®	System z9®
AIX*	POWER7	System z10
BladeCenter*	ProtecTIER*	Tivoli*
DataPower*	RACF*	WebSphere*
CICS*	Rational*	XIV*
DB2*	Redbooks®	zEnterprise
DS4000*	Sysplex Timer®	z/OS*
ESCON®	System Storage	z/VM*
		z9®

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

IBM, z/OS, Predictive Failure Analysis, DB2, Parallel Sysplex, Tivoli, RACF, System z, WebSphere, Language Environment, zSeries, CICS, System x, AIX, BladeCenter and PartnerWorld are registered trademarks of IBM Corporation in the United States, other countries, or both.
 DFSMSHsm, z9, DFSMSmm, DFSMSdtp, DFSMSdes, DFSMS, DFS, DFSORT, IMS, and RMF are trademarks of IBM Corporation in the United States, other countries, or both.
 Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. 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.
 InfiniBand is a trademark and service mark of the InfiniBand Trade Association.
 UNIX is a registered trademark of The Open Group in the United States and other countries.
 Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

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.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.


All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

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

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the 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.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

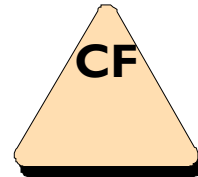


2

**For a complete list of IBM Trademarks, see
www.ibm.com/legal/copytrade.shtml**

Agenda

- Hardware Updates
 - **CFCC Level 17**
 - CFCC Level 16
 - Parallel Sysplex InfiniBand Links
- z/OS Updates
 - Sysplex Failure Management
 - z/OS V1R12
 - z/OS V1R13 preview
- Summary



3

CFLEVEL 17



- **IBM zEnterprise™ 196 (z196)**
 - Available September 2010
- Up to 2047 structures
- Up to 255 connectors per structure
 - z/OS imposes smaller limits for lock structures and serialized list structures
- Prerequisites
 - z/OS V1.10 or later with PTF for OA32807
 - z/VM V5.4 for guest virtual coupling



4

zEnterprise 196 servers with Coupling Facility Control Code (CFCC) Level 17 supports **up to 2047 structures per Coupling Facility (CF) image**, up from the prior limit of 1023. This allows you to define a larger number of data sharing groups, which can help when a large number of structures must be defined, such as to support SAP configurations or to enable large Parallel Sysplex configurations to be merged. This function requires z/OS 1.10 or later with the PTFs for APAR OA32807.

zEnterprise 196 servers with CFCC Level 17 supports **more connectors to list and lock structures**. XES and CFCC already support 255 connectors to cache structures. With this new support XES also supports up to 247 connectors to a lock structure, 127 connectors to a serialized list structure, and 255 connectors to an unserialized list structure. This support requires z/OS 1.10 or later with the PTFs for APAR OA32807.

z/OS V1.12 supports **larger Coupling Facility (CF) structures**. The maximum size you can specify for a CF structure is increased from slightly below 100 GB (99,999,999 KB) to 1 TB. Also, the CFRM policy utility (IXCMIAPU) is updated to allow you to specify structure sizes in units of KB, MB, GB, and TB. These changes improve both Parallel Sysplex CF structure scalability and ease of use.

CFLEVEL 17 ...



- CF Diagnostics
 - Non-disruptive dumping
 - Improved diagnostics (coordinated capture)
- Prerequisites
 - z/OS V1.12
 - z/OS 1.10 or 1.11 with PTFs for OA31387

SHARE
in Anaheim
2011

5

z/OS V1.12, and with the PTFs for OA31387 installed, z/OS V1.10 and z/OS V1.11, in conjunction with z196 servers and Coupling Facility control code (CFCC) Level 17, can capture Coupling Facility (CF) data nondisruptively in some circumstances, allowing the CF to continue operating. This new function is intended to help improve Parallel Sysplex availability when it is necessary to capture CF data. The CF uses a pre-staged dump capture area to avoid collateral effects observable by z/OS, such as message time-outs (observed as interface control checks) or loss of connectivity.


Before the introduction of this support, there were two ways to capture diagnostic information from the CF. The most useful way required a disruptive CF hard dump, requiring a reboot of the CF and lossconn recovery (rebuilds, etc.) from z/OS. Understandably, customers are rarely willing to install diagnostic CFCC code loads to produce such a dump. A secondary method is a CFCC soft dump, which produces a “blurry”, unserialized picture because CF activity is not quiesced while the dump is captured.

Support for serialized non-disruptive CF dumps is introduced with CFLEVEL 17. Dumps can be triggered in several ways:

- CFCC has a number of detection points, mostly where it can be recognized that duplexing is about to break.
- z/OS can explicitly request a CF dump.
- When requested by z/OS, the link can recognize an impending timeout and trigger collection of CF, z/OS, and link data. Thus one can obtain a coherent set of problem determination data from the CF, z/OS, and the connecting links (coordinated capture)

The Support Element has been enhanced to provide storage for up to 10 dumps vs. the current 2.


CFCC has implemented a 5-minute refractory period. Once a non-disruptive dump has been taken, CFCC will not take another within the refractory period, no matter what the trigger. This minimizes the likelihood of getting a flood of dumps for the same incident, for example when a duplex break causes many commands against the affected structure to observe the built-in internal triggers.



CFLEVEL 17 ...

Migration

- z196 DR86 contains CFCC Level 17 support
 - In general, get to most current LIC levels
- Use CF Sizer website to check/update structure sizes:
 - CF structure sizes may increase when migrating to CFCC Level 17 from earlier levels due to additional CFCC controls
 - IBM's testers saw 0-4% growth from CFLEVEL=16
- Note that CFCC image size is now 512MB



6

CF Level 17 is expected to have marginal impact to structure sizes. Representative, properly sized structures used by IBM testers grew between 0 and 4% when allocated in a coupling facility running CFLEVEL 17 vs a coupling facility running CFLEVEL 16. These results may not apply to your environment. Thus IBM suggests using the CF Sizer to size structures whenever a new CF level is installed. The CF Sizer can be found at this web site:

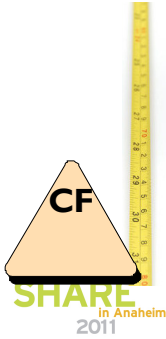

www.ibm.com/systems/support/z/cfsizer/index.html

The storage requirement for the CFCC image size is now 512MB (previously 128MB). This is independent of any structure size growth.

CF Sizer Enhancements

- Improved sizings
 - IMS, DB2, XCF
- Additional structures
 - IBM Session Manager, InfoSphere Classic
- Usability improvements

www.ibm.com/systems/support/z/cfsizer/



7

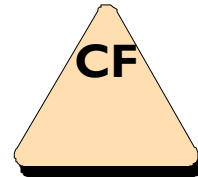
The web-based CFSIZER tool is designed to help you perform coupling facility structure sizing calculations.

Concurrently with z/OS V1.12 availability, CFSIZER provides:

- More accurate size calculations for the IMS Operations Management (OM) Audit, Resource, and Overflow Sequential Access Method (OSAM) structures
- Calculated sizes for IBM Session Manager (ISM) Sysplex User structures
- Improved sizes for XCF signaling structures
- Calculated sizes for InfoSphere™ Classic control and filter structures
- Improved sizes for DB2 SCA structures
- Various usability improvements to the CFSIZER Web pages, including consolidated structure input pages to the OEM cache structure page.

Agenda

- Hardware Updates
 - CFCC Level 17
 - **CFCC Level 16**
 - Parallel Sysplex InfiniBand Links
- z/OS Updates
 - Sysplex Failure Management
 - z/OS V1R12
 - z/OS V1R13 preview
- Summary



CFLEVEL 16



- System z10, October 2008
- CF Duplexing Protocol Enhancements for improved duplex response time
- CF Notification Enhancements to avoid false schedules for Shared Message Queue exploiters
- CF Storage increment size increase 512KB → 1 MB
- Prerequisite:
 - z/OS V1R6 or later with PTFs for APAR OA25130
 - z/OS V1R11 or later

SHARE
in Anaheim
2011

9

CFLEVEL 16 provides enhancements that may improve the response time of duplexed requests and may reduce the overhead of running certain applications that exploit shared message queues.

CFLEVEL 16 increased the size of the blocks of storage that are managed by the CF. This change will likely change the allocated structure sizes relative to prior CFLEVELs and necessitate updates to the structure sizes specified in the Coupling Facility Resource Manager (CFRM) policy.

z/OS support for CFLEVEL 16 is available at z/OS V1R6 and up.

CFLEVEL 16 – Asynchronous RTC

- Designed to improve duplexed request response time
 - Depends on structure's usage of duplexed CF requests
 - Improvements vary with distance
- Requires pairs of CFs
 - CFCC Level 16 or later
 - z10 or z196 servers

10

SHARE
in Anaheim
2011

CF (Coupling Facility) requests are likely to have longer service times when issued against a duplexed structure vs. a simplex structure. The duplexed command pairs must execute in a coordinated fashion within the two coupling facilities, a process that requires the exchange of signals between the two CFs. The exchanging of these signals, especially at large distances, takes time. The impact on coupling efficiency for a particular application will vary according to rate at which CF requests are generated, as well as the relative proportion of requests that modify structure objects (writes) to requests that don't (reads).

The coupling facilities containing the duplexed instances of the structure exchange Ready To Execute (RTE) signals to coordinate starting of the duplexed command pair. After completing the command, the coupling facilities exchange Ready To Complete (RTC) signals to commit the results.

Asynchronous RTC allows a duplexed request to complete without waiting for the RTC exchange between coupling facilities to complete. Rather than eliminating the RTC exchange, the RTC exchange will occur asynchronous to the completion of the request – improving the response time for the request. The improvement for any particular customer will of course depend on the particulars of the workload and its access patterns for the structure of interest. In general, one could see roughly 10-15% improvement in response time for duplexed requests. In general, as the distance between the coupling facilities increases, the improvement in response time will be more noticeable.

Use of asynchronous RTC does not impact (reduce) host CPU utilization.

The two coupling facilities containing the duplexed instances of the structure must be running CFCC Level 16 (for z10) and/or CFCC Level 17 (for z196).

CFLEVEL 16 – Sublist Notification

- Avoid false schedules for shared queue exploiters
 - IMS Shared Message Queue
 - MQ Shared Queues
- Empty to non-empty state change notification sent to one connector in round robin fashion
- If no response in (time period), then send to rest of the connectors

11


SHARE
Technology • Connections • Results

SHARE
in Anaheim
2011

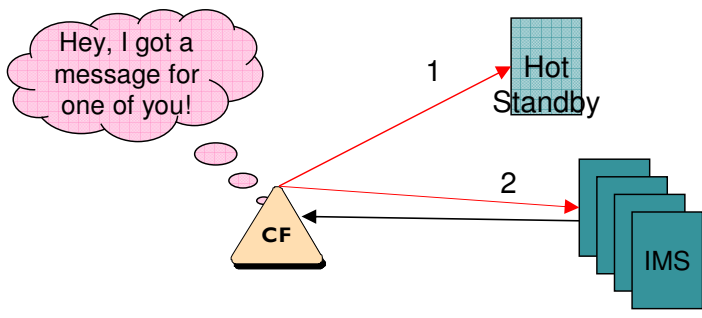
The sublist notification protocol would benefit and be of general interest to Parallel Sysplex customers, particularly those who make use of the existing sublist notification function for CF Keyed List Structures. Exploiters of sublist monitoring, such as MQ Shared Queues and IMS Shared Message Queue, can transparently reduce their scheduling overhead with the enhanced sublist notification mechanism.

This Sublist Notification protocol identifies in a round robin fashion a monitored instance of a sublist transition. The identified instance is notified of the sublist transition immediately without delay. Any other monitored instances may be notified of the sublist transition after a specific delay time. This allows time for the first notified monitored instance to process the work before the other monitored instances are notified. If the first notified monitored instance can process the work, the other monitored instances may not be notified. This prevents the other monitored instances from unnecessarily trying to process work that has already been handled.

CFLEVEL 16 – Sublist Notification ...




- But ... in hindsight, round robin notification may not be so hot for a Hot Standby environment
 - Standby does not process the notifications
 - CF eventually notifies next in line
 - But the delay may be intolerable
- APAR OA30994 provides new controls to either tune or disable the protocol, on a structure by structure basis



The diagram illustrates the notification flow. A yellow triangle labeled 'CF' has a thought bubble above it that says 'Hey, I got a message for one of you!'. A red arrow labeled '1' points from the CF to a blue grid labeled 'Hot Standby'. A second red arrow labeled '2' points from the CF to a stack of blue rectangles labeled 'IMS'. A black arrow points from the IMS stack back to the CF.

12



CFCC initially had a defect wherein it was not continuing on to provide subsequent notifications if the first one did not grab the work fast enough. Some customers that had a hot standby environment were significantly impacted if the hot standby was selected to receive the initial (and only) notification. Since the CF in effect lost initiative to provide notifications and the hot standby did not process the work, things hung. The CFCC has now fixed the defect (be sure you have the latest MCL). But analysis of this problem situation suggests that this new form of sublist notification may not be as suitable for all environments as first thought.

Sublist Notification Delay function was delivered in z/OS V1R10 (HBB7750), and requires CFCC Level16. The original support set the sublist notification delay time value to 5 microseconds. The sublist monitoring delay function was automatically applied to any CF List Structure allocated with Primary Keys with monitoring connectors.

With OA30994, enhancements are provided to allow the installation to specify the sublist notification delay time. A new CFRM policy parameter (SUBNOTIFYDELAY) on the existing STRUCTURE definition statement will allow the sublist notification delay time to be specified in microseconds. The delay time will take effect immediately when the policy is activated. The default sublist notification delay time will remain at 5 microseconds if the new parameter is not specified. SUBNOTIFYDELAY can be a number in the range of 0 to 1000000 (1 million) microseconds. For SUBNOTIFYDELAY to work predictably, SUBNOTIFYDELAY support needs to be present on all systems.

CFLEVEL 16 – Non-Disruptive Dumping



- Non-disruptive dumping available
 - As of this past Monday (Feb 28, 2011)
- Prerequisites
 - Appropriate CFCC service level (CF16 SL 4.01)
 - z/OS V1.9 and later with PTFs for OA33723

13

SHARE
in Anaheim
2011

Similar to the support provided with CFLEVEL 17 (which runs on a z196), the ability to take non-disruptive dumps for a CF running with CFLEVEL 16 (which runs on a z10) is also available with CFCC service level CF16 SL 4.01 or later. Non-disruptive dumps are not always possible, but in cases where they apply, the CF can continue operating. This new function can help improve Parallel Sysplex availability when it is necessary to capture CF data.

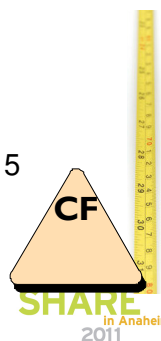
Nondisruptive CF dumping with CFLEVEL 16 requires support that is available on z/OS V1R9, V1R10, V1R11, and V1R12 with the PTFs for OA33723 installed.

CFCC Level 16 Migration



- z10 DR76 and DR79 contains CFCC Level 16 support
 - In general, get to most current LIC levels (**04.01**)
 - **See APAR OA31960 regarding service level 02.12 to 02.22**
- Use CF Sizer website to check/update structure sizes:
 - Many CF structure sizes will increase when migrating to CFCC Level 16 from earlier levels due to:
 - Increase in storage increment size to 1MB
 - Additional CFCC controls
 - Info APAR II14431
 - Recommend using CF Sizer
 - IBM's testers saw 5-10% growth from CFLEVEL=15

www.ibm.com/systems/support/z/cfsizer/index.html



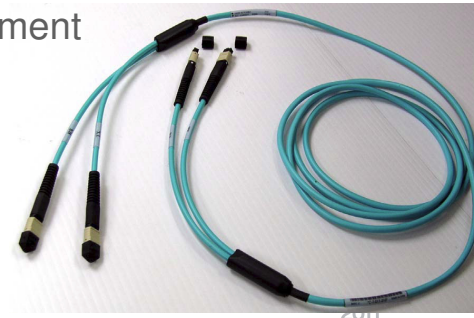
14

The problem documented in APAR OA31960 was introduced by CF micro-code and is fixed by CF micro-code. The problem is documented in an XES software APAR because the symptoms associated with the problem commonly surface in the CF Activity Report. If you are experiencing the problem documented in this APAR please contact your IBM CE for further assistance, reference RETAIN TIP H195228. Customers running with CFs at CFCC RELEASE 16.00, SERVICE LEVEL 02.12 or higher may experience application or batch job slow downs due to nearly all of the requests going to a structure being changed to async. The slowdowns have been seen particularly during intensive activity caused either by DB2 image copy jobs or write-intensive DB2 reorg jobs, or other similar workloads. Symptoms seen by several customers, though not a huge number. Typically, online performance is OK, but batch workload performance suffers significantly. **The problem is resolved as of CFCC Release 16 Service Level 2.24**

In general, one should always use the CF Sizer to resize structures when a new CF level is installed. In general, the size of a structure allocated at CF Level 16 will be larger than the size of the structure allocated at any lower CF level. The change in size for any particular structure is difficult to predict in general because the size changes are very dependent on the way the exploiting application actually uses the structure (the additional control space needed by the CF will depend on what's in the structure). Info APAR II14431 basically says: sizes will change, use the CFSIZER. Our test results on average saw 5-10% growth over a properly sized structure allocated in a CFLEVEL=15 CF.

Agenda

- Hardware Updates
 - CFCC Level 17
 - CFCC Level 16
 - **Parallel Sysplex InfiniBand Links**
- z/OS Updates
 - Sysplex Failure Management
 - z/OS V1R12
 - z/OS V1R13 preview
- Summary



Infiniband Coupling (PSIFB)



- New CF link type (CIB) for all IB coupling links
 - System z9, z10, and z196
 - 7 subchannels per CHPID applies to CIB links
- Simplifies physical connectivity
 - Multiple CIB CHPIDs per physical IB Coupling link
 - A single physical link can “share” CHPIDs across multiple CF images, within same sysplex, or across different ones
 - Additional subchannels, without additional physical links
- Additional flexibility for physical configuration
 - 150 meters (vs ICB4 limit of 10 meters)

PSIFB = PSIB = IFB = IB Coupling = CIB



16

Infiniband links are sometimes called IFB, PSIB, PSIFB, IB Coupling, or CIB (coupling over infiniband). They are all the same thing.

There are several ways of classifying IFB links. It can be Double Data Rate (DDR) or Single Data Rate (SDR). The IFB links can also be characterized by the amount of lanes of communication within each physical link. For example, 12x links provide 12 lanes of communication within each physical link. 1x links provide a single lane of communication within each physical link.

7 subchannels per CHPID is same as for other links. What's unique to infiniband is ability for the physical link to share CHPIDs.

Infiniband Coupling (PSIFB)

- Up to 16 CHPIDs can use same physical link
 - More subchannels / physical link
 - NOT** more subchannels / CHPID
- Can connect to multiple CF LPARs
- CF Receiver CHPIDs can share link

7 SubChannels / path
 Up to 16 channel paths / link

 Up to 112 subchannels / link

- MIF uses same address, 7 subchannels / CHPID

17

Note that CF receiver CHPIDs can now share the same physical infiniband link. Prior to this, there was a one-to-one correspondence between receiver CHPID and CF Link. In effect there will be a one-to-one correspondence between a sender CHPID and a receiver CHPID, but the physical link can be shared by many such pairs.

Allowing the same physical link to be shared by multiple sysplexes can be quite useful if connecting to a server containing multiple CFs at the “receiver” end. For example, if there are two CHPIDs defined at the “sender”, one can be directed to connect to one CF, the other CHPID to the other CF. Depending on the configuration, this may provide for a reduction in the number of coupling links required.

IBM expects that most clients will not see reasons nor benefits in configuring more than 8 CHPID across the two ports of an InfiniBand coupling HCA. However, this is not a restriction and configuring up to 16 CHPIDs across the two ports of the InfiniBand coupling HCA is indeed supported. However, there is a diminishing benefit to adding these CHPIDs and configuring more than necessary may actually impact performance instead of help.

Thus configuring up to 16 CHPIDs might be reasonable in cases where the load is “light” but the need for connections is “high”. But in cases where the load is “heavy”, one should refrain from going above 8 CHPIDs as one reaches a point of diminishing returns.

What does it mean to you?



- Fewer physical links
 - Easier to configure since less cabling is needed
 - Potential cost savings if allows fewer books on the machine
- Can define more CHPIDs to a physical link to get more subchannels
 - Beneficial for long links as can get more bandwidth
 - Reduce subchannel busy conditions
- Can have more CF's on a box if links were limiting factor

NOTE: z196 doubles number of coupling CHPIDs to 128 and increases number of physical coupling links to 80



18

Fewer physical links could enable \$\$ savings by reducing need for books on the machine, if number of books was related to need for physical connectivity.

Enable more subchannels per physical link: Each peer-mode coupling link supports seven subchannels, or seven concurrent messages to a coupling facility. If the volume and duration of CF accesses is high enough to cause subchannel busy conditions such as when data sharing across distances, then the additional subchannels can improve communication performance without using more physical links.

The zEnterprise 196 provides supports 128 coupling CHPIDs per server: To support larger Parallel Sysplexes with ever-increasing amounts of data sharing traffic to the Coupling Facility, the throughput and capacity of more coupling CHPIDs is also required. With z196, the number of coupling CHPIDs per server has been increased from 64 to 128. Since IFB links allow for multiple (logical) CHPIDs over the same (physical) link, this can also allow for larger Parallel Sysplexes without requiring more coupling link hardware. Also, the number of physical coupling links of any type is increased to 80 from the previous limit of 64.

PSIFB Configuration Considerations



- Pure Capacity
 - One 12x PSIFB replaces one ICB4
 - One 12x PSIFB replaces four ISC3
- Eliminating Subchannel and Path Delays
 - Extra ICB4 links might be configured to get additional subchannels/paths to eliminate delays caused by busy conditions
 - One 12x PSIFB link with multiple CHPIDs could replace multiple ICB4s in this case
- Multiple sysplexes sharing hardware
 - Production, development, and test sysplexes can share hardware, but they each need their own ICB4 and ISC3 links
 - One PSIFB link with multiple CHPIDs could replace multiple links in this case

Be sure to maintain redundancy !

SHARE
in Anaheim
2011

19

From a pure capacity perspective, one 12x PSIFB link is equivalent to either one ICB4 link or four ISC3 links.

Many installations configuring ICB4 links for their z9 or z10 servers, find that two such links between the z/OS image and the CF are sufficient. Some installations need the added capacity of a 3rd or 4th ICB. In those cases, an installation looking to replace ICB4 links with PSIFB links would do so on a one for one basis. More typically, installations need to configure a 3rd or 4th ICB4 link not for capacity reasons, but to overcome delays caused by “path busy” and “subchannel busy” conditions that can occur due to “bursty” traffic. In those cases, two PSIFB links with a pair of CHPIDs assigned to each link could replace the three or four ICB4 links. That is, one PSIFB link could replace more than one ICB4 link.


Some installations have multiple CECs, with multiple sysplexes running on each CEC. So on CEC “B”, one might for example have coupling facility CF1B used by sysplex 1, and coupling facility CF2B used by sysplex 2. CEC “A” would need at least two ICB4 links (for example) to allow the z/OS images on CEC “A” to communicate with those coupling facilities. One link would connect CEC “A” to coupling facility CF1B, and the other would connect it to CF2B. A separate link would be needed for each CF because there is a one-to-one correspondence between a CF receiver CHPID and an ICB4 link. However, one PSIFB link could be used to connect the two CECs because the receiver CHPIDs for each CF on CEC “B” could share the PSIFB link. In these cases, where multiple ICB4 and ISC3 links are configured to establish necessary connectivity (as opposed to capacity), one PSIFB link with multiple CHPIDs could be used to replace multiple links.

Some real world examples where installations have converted to PSIFB links.

- Consolidating to System z10 model E64 which does not support ICB-4.
- Consolidation of 16 ISC-3 links to 4 IFB (maintaining redundancy across two HCA) within a datacenter. Infrastructure savings and improvement in coupling efficiency.
- Installation had 2 ICB for development and 4 ICB for production Sysplex sharing System z10. Converted to 4 PSIFB links with shared CHPIDs for development and production without a significant decrease in coupling efficiency.

After PSIFB links are configured, additional CHPIDs can be added to a configuration providing additional subchannels without requiring additional physical hardware infrastructure.

Coupling Link Choices - Overview



- **ISC (Inter-System Channel)**
 - ▶ Fiber optics
 - ▶ I/O Adapter card
 - ▶ 10km and longer distances with qualified WDM solutions
- **PSIFB (1x IFB)**
 - ▶ Fiber optics – uses same cabling as ISC
 - ▶ 10km and longer distances with qualified WDM solutions
 - ▶ Supports multiple CHPIDs per physical link
 - ▶ Multiple CF partitions can share physical link
- **PSIFB (12x IFB)**
 - ▶ 150 meter max distance optical cabling
 - ▶ Supports multiple CHPIDs per physical link
 - ▶ Multiple CF partitions can share physical link
- **ICB (Integrated Cluster Bus)**
 - ▶ Copper cable plugs close to memory bus
 - ▶ 10 meter max length
 - ▶ Not available on z196
- **IC (Internal Coupling Channel)**
 - ▶ Microcode - no external connection
 - ▶ Only between partitions on same processor

Relative Performance
Based on avg data xfer size

SHARE
in Anaheim
2011

20

The slide lists the links in order of increasing performance, top to bottom. The 1x IFB links tend to be somewhat faster than the ISC links. The 12x IFB links are slightly slower than ICB links, but allow much greater distances.

ISC-3 links provide the connectivity required for data sharing between the CF and the systems. ISC links support point-to-point connections (directly connecting CFs and systems), and require a unique channel definition at each end of the link.

InfiniBand coupling links (PSIFB) are high speed links on z196, z10 and z9 servers. PSIFB coupling links use a fiber optic cable that is connected to a Host Channel Adapter fanout in the server.

Integrated Cluster Bus links are members of the family of coupling link options available on System z10 and previous System z servers. They are faster than ISC links, attaching directly to a Self-Timed Interconnect (STI) bus of the server. The ICB features are highly integrated, with very few components, and provide better coupling efficiency (less server overhead associated with coupling systems) than ISC-3 links. They are an available method for coupling connectivity when connecting System z10 and previous System z servers over short distances (seven meters). For longer distances, PSIFB (up to 150 meter), PSIFB LR (up to 100 km), or ISC-3 links (up to 100 km) must be used.

IC links are Licensed Internal Code-defined links to connect a CF to a z/OS logical partition in the same server. These links are available on all System z servers. The IC link is a System z server coupling connectivity option that enables high-speed, efficient communication between a CF partition and one or more z/OS logical partitions running on the same server. The IC is a linkless connection (implemented in Licensed Internal Code) and so does not require any hardware or cabling.

Coupling Technology versus Host Processor Speed

Host effect with primary application involved in data sharing

Chart below is based on 9 CF ops/Mi - may be scaled linearly for other rates

Host CF	z890	z990	z9 BC	z9 EC	z10 BC	z10 EC	z196
z890 ISC	13%	15%	16%	17%	19%	21%	NA
z890 ICB	9%	10%	10%	11%	12%	13%	NA
z990 ISC	13%	14%	14%	15%	17%	19%	NA
z990 ICB	9%	9%	9%	10%	12%	13%	NA
z9 BC ISC	12%	13%	14%	15%	17%	19%	23%
z9 BC PSIFB 12X	NA	NA	NA	NA	13%	14%	16%
z9 BC ICB	8%	9%	9%	10%	11%	12%	NA
z9 EC ISC	12%	13%	13%	14%	16%	18%	22%
z9 EC PSIFB 12X	NA	NA	NA	NA	13%	14%	16%
z9 EC ICB	8%	8%	8%	9%	10%	11%	NA
z10 BC ISC	12%	13%	13%	14%	16%	18%	22%
z10 BC PSIFB 12X	NA	NA	11%	12%	13%	14%	15%
z10 BC ICB	8%	8%	8%	9%	10%	11%	NA
z10 EC ISC	11%	12%	12%	13%	15%	17%	22%
z10 EC PSIFB 12X	NA	NA	10%	11%	12%	13%	15%
z10 EC ICB	7%	7%	7%	8%	9%	10%	NA
z196 ISC	NA	NA	11%	12%	14%	16%	21%
z196 PSIFB 12X	NA	NA	9%	10%	11%	12%	14%

With z/OS 1.2 and above, synchron->asynch conversion caps values in table at about 18%

PSIFB 1X links would fall approximately halfway between PSIFB 12X and ISC links

IC links scale with speed of host technology and would provide an 8% effect in each case

21


2011

The coupling efficiency of a Parallel Sysplex cluster, particularly one that has heavy datasharing, is sensitive to the performance of the operations to the Coupling Facility. The chart estimates the "host effect" for a heavy data sharing production workload for various combinations of host processor and coupling technology. The values in the table represent the percentage of host capacity that is used to process operations to the coupling facility (which includes the time spent communicating with the CF and the time spent waiting for the CF to process the request). For example, a value of 10% would indicate that approximately 10% of the host capacity (or host MIPS) is consumed by the subsystem, operating system and hardware functions associated with coupling facility activity. The table is based on a "coupling intensity" of 9 CF operations per million instructions (MI), which is typical of high end data sharing work loads.

The values in the table can be adjusted to reflect the coupling intensity for any workload. One can calculate the coupling intensity by simply summing the total req/sec of the CFs and dividing by the used MIPS of the attached systems (MIPS rating times CPU busy). Then, the values in the table would be linearly scaled. For example, if the workload was processing 4.5 CF operations per million instructions (or 4.5 CF ops/second/MIPS), then all the values in the table would be cut in half.

For 9 CF requests/MI, host effect values in the table may be considered capped at approximately 18% due to z/OS Synchronous to Asynchronous CF Message Conversion. Configurations where entries are approaching 18% will see more messages converted to asynchronous. z/OS converts synchronous messages to asynchronous messages when the synchronous service time relative to the speed of the host processor exceeds a breakeven threshold at which it becomes cheaper to go asynchronous. When all CF operations are asynchronous, the overhead will be about 18%. By the time you have reached >=18% in the table, that corresponds to the time z/OS must have been converting almost every operation asynchronous. The 18% cap scales proportionally with the CF requests/MI activity rate. For example, at 4.5 CF requests/MI, the cap would 9%.

The hardware cost can be minimized by using the most efficient links with faster engine speeds for the CFs. This reduces the time that z/OS is waiting for the response while the message is on the CF link and while the CF is busy processing the request. With this in mind, it becomes obvious that the best coupling efficiency is generally seen when the CF is on the fastest processor and connected to the z/OS image via the fastest links. The chart bears this out. For example, holding the CF and host processor technology constant, the chart shows that coupling efficiency increases with faster links (z/OS spends less time waiting because the communication with the CF is faster). For a given host processor and link technology, coupling efficiency increases with faster CF technology (z/OS spends less time waiting because the CF processes the request faster). In most cases, upgrading to faster links has a more dramatic impact on coupling efficiency than upgrading to a faster CF.



Maximum CF Links

Server	IC	IFB	ICB-4	ICB-3	ICB	ISC-3	Max # Links
z800	32	-	-	5 6 (OCF)	-	24	26 + 32
z900-100 CF	32	-	-	16	16	32 42 w/RPQ	64
z900	32	-	-	16	8 16 w/RPQ	32	64
z890	32	-	8	16	-	48	64
z990	32	-	16	16	8	48	64
z9 EC	32	16	16	16	-	48 Peer	64
z9 BC	32	12	16	16	-	48 Peer	64
z10 EC	32	32	16	-	-	48 Peer	64 32 IFB + ICB-4
z10 BC	32	12	12	-	-	48 Peer	64 56 External 12 IFB + ICB-4
z196	32	32	-	-	-	48 Peer	80

SHARE
in Anaheim
2011

22

The zEnterprise 196 increases the number of external coupling links allowed from 64 to 80. This allows the full configuration of 32 PSIFB links and 48 ISC-3 links to be used. In addition, you can also configure up to 32 (internal) IC links for coupling between images defined on the same server. Having more coupling links is important to provide sufficient coupling connectivity for larger single Parallel Sysplexes, as well as for configurations where the same server hosts multiple Parallel Sysplexes and Coupling Facility images.

For z196 servers the maximum number of coupling links (CHPIDs) combined (PSIFB, active ISC-3 links, and IC) is 128 and up to 80 physical external links (PSIFB, active ISC-3). For z9 or z10 servers the maximum number of coupling links (CHPIDs) combined (ICB-3, ICB-4, PSIFB, active ISC-3 links, and IC) is 64.

The z10 BC only has room for six fanout cards. Configuring the max of 12 IFB links would use all six fanout cards. Configuring the max of 12 ICB-4 links would use all six fanout cards. So the total number of ICB-4 and IFB links cannot exceed 12. Configuring the max of 48 ISC links would require 2 fanout cards and 2 I/O drawers. Thus configuring one ISC link, reduces the number of high speed (IFB or ICB) links to 8.

The z10 EC supports a max of 32 IFB and ICB-4 links in any combination.

Certain models have smaller limits:

A maximum of 8 ICB-4s are supported with a z9 BC capacity setting A01.


A maximum of 16 PSIFB links are supported on z196 model M15.

A maximum of 16 PSIFB links are supported on z10 EC model E12.


A maximum of 12 PSIFB links are supported on z9 EC model S08.

The brackets on the sides of the chart are meant to suggest configurations that are supported within the same sysplex (to account for N-2 compatibility).

PSIFB Configurations Supported



CF	z/OS	z9	z10	z196
z9		No	Yes	Yes
z10		Yes	Yes	Yes
z196		Yes	Yes	Yes



23

The z890 and z990 do not support PSIFB links.

On the z196, z10 and z9 servers InfiniBand coupling links (PSIFB) are available. PSIFB supports peer mode connectivity between:

- Any z196 or z10 to any z9 (Link is 12x IB-DDR but operates at SDR)
- Any z196 or z10 to any z10 or z196 (12x IB-DDR and 1x IB-SDR or DDR)
- z9 to z9 PSIFB is not supported

DDR = double data rate, SDR = single data rate

Distance Considerations



Distance	IC	ICB-4	12x IFB	ISC-3 1x IFB
Within server	Yes	n/a	n/a	n/a
<10 m		Yes	Yes	Yes
10 m – 150 m			Yes	Yes
150 m – 100+ km				Yes

SHARE
in Anaheim
2011

24

The ISC and PSIFB 1x provide longer links, but slower. The PSIFB is a nice replacement for the ISC links. Both types of links require some sort of repeater (DWDM dense wave division multiplexer) to go beyond 10 km. The PSIFB 12x and ICB links are shorter range, but faster. PSIFB can extend distance.

1x IFB, like a 1 lane highway, good for up to 10 kilometers

12x IFB, like a 12 lane highway, get more bandwidth / parallelism / striping.

Statement of Direction



- **The System z10 will be the last server to support ICB-4 links**
 - Feb 26, 2008 z10 GA announcement
 - Restated in April 28, 2009 announcement
- **Implications are now reality:**
 - To connect a z9 or a z10 to a z196 with high-speed coupling links, one must configure some PSIFB links

25

SHARE
in Anaheim
2011

February 26, 2008: ICB-4 links to be phased out: IBM intends to not offer Integrated Cluster Bus-4 (ICB-4) links on future servers. IBM intends for System z10 to be the last server to support ICB-4 links.

April 28, 2009: The System z10 will be the last server to support ICB-4 links. IBM intends to not offer Integrated Cluster Bus-4 (ICB-4) links on future servers. IBM intends for System z10 to be the last server to support ICB-4 links as originally stated in Hardware Announcement 108-154, dated February 26, 2008.

The z196 is one such “future server”. The z196 does not support ICB links. One must configure infiniband links to connect a z196 to either a z9 or a z10 with high speed links.

Statement of Direction



- **Z196 announcement, July 22, 2010**
- **The z196 will be the last high-end server to:**
 - Offer ordering of ESCON channels
 - Offer ordering of ISC-3
 - Support dial-up modem
- Implications
 - If using CTC devices for XCF signalling paths, need to migrate to FICON from ESCON
 - Migrate from ISC-3 coupling links to infiniband
 - Migrate to alternatives for dial-up time services

SHARE
in Anaheim
2011

26

The ordering of ESCON channels applies to channel path identifier (CHPID) types CNC, CTC, CVC, and CBY. Enterprises should begin migrating from ESCON to FICON. Alternate solutions are available for connectivity to ESCON devices. IBM Global Technology Services, through IBM Facilities Cabling Services, offers ESCON to FICON Migration (Offering ID #6948-97D), to help facilitate migration to FICON to simplify and manage a single physical and operational environment while maximizing green-related savings.

The IBM z196 CPC will be the last high-end server to offer ordering of ISC-3. Enterprises should begin migrating from ISC-3 features (#0217, #0218, #0219), to 12x InfiniBand (#0163 - HCA2-O fanout) or 1x InfiniBand (#0168 - HCA2-O LR fanout) coupling links.

The IBM z196 CPC is planned to be the last high-end server to support dial-up modems for use with the Remote Support Facility (RSF), and the External Time Source (ETS) option of Server Time Protocol (STP). The currently available Network Time Protocol (NTP) server option for ETS as well as Internet time services available using broadband connections can be used to provide the same degree of accuracy as dial-up time services. Enterprises should begin migrating from dial-up modems to Broadband for RSF connections.

For more information



- “IBM System z Connectivity Handbook” (SG24-5444)
- “Getting Started with Infiniband on System z10 and System z9” (SG24-7539)
 - Available at www.redbooks.ibm.com
- www.ibm.com/systems/z/advantages/pso/whitepaper.html
 - CF Configuration Options White Paper
- Session 8863: Coupling Technology Overview and Planning - What’s the Right Stuff for Me?



27

www.redbooks.ibm.com/redpieces/pdfs/sg245444.pdf This IBM® Redbooks® publication discusses the connectivity options available for use within and beyond the data center for the IBM System z® family of mainframes. The book highlights the hardware and software components, functions, typical uses, coexistence, and relative merits of these connectivity features. It will assist readers in understanding the connectivity alternatives that are available when planning and designing their data center infrastructure.

www.redbooks.ibm.com/redbooks/pdfs/sg247539.pdf This IBM® Redbooks® publication provides introductory information about the InfiniBand standard and how that standard is implemented and used to support system connectivity for System z10™ and System z9® servers. The book will help you plan and implement InfiniBand support in your System z10 and System z9 environment. It also provides step-by-step information about configuring InfiniBand connections.

www.ibm.com/systems/z/advantages/pso/whitepaper.html has a link to the white paper on “Coupling Facility Configuration Options”. This paper examines the various *Coupling Facility* technology alternatives from several perspectives. The characteristics of each CF option are compared in terms of function, inherent availability, performance and value. It also looks at CF structure placement requirements based on an understanding of CF exploitation by z/OS® components and subsystems.

Agenda


- Hardware Updates
 - CFCC Level 17
 - CFCC Level 16
 - Parallel Sysplex InfiniBand Links
- z/OS Updates
 - **Sysplex Failure Management**
 - z/OS V1R12
 - z/OS V1R13 preview
- Summary




Sysplex Failure Management (SFM)

- MEMSTALLTIME
- SSUMLIMIT
- SFM and AutoIPL
- **SFM with BCPii**
- System Default Action
- XCF FDI Consistency
- **Critical Members**
- **CFSTRHANGTIME**

- z/OS 1.8
- z/OS 1.9
- z/OS 1.10
- z/OS 1.11 ←
- z/OS 1.11
- z/OS 1.11
- z/OS 1.12
- z/OS 1.12



SFM is the subcomponent within XCF that deals with the detection and resolution of sympathy sickness conditions that can arise when a system or sysplex application is unresponsive

29


SFM deals with the detection and resolution of sympathy sickness conditions that can arise when a system or sysplex application is unresponsive. This slide summarizes the most recent SFM related technologies. Due to time restrictions, only those topics in boldface will be discussed during the presentation. However, Session 9028 Parallel Sysplex Resiliency this past Tuesday at 1:30 pm discussed most of these items in detail. You are encouraged to investigate and exploit them as appropriate. I claim that exploitation of “SFM with BCPii” may well be the most significant thing you can do for your installation with regard to availability in the sysplex.


Single-system “sick but not dead” issues can and do escalate to cause sysplex-wide problems. A sick system typically holds resources needed by other systems and/or is unable to participate in sysplex wide processes. Thus other systems become impacted. But the root cause of the sickness is a single system problem (contention, dispatching delays, spin loops, overlays, queue/data corruption, etc). Routing work away from the troubled system does not necessarily guarantee that other systems will not be impacted.

System/sysplex cleanup when subsystems or systems actually *terminate* is not the problem. Indeed, removal of the sick system from the sysplex generally remedies the problems. Allowing non-terminating problems, where something simply becomes unresponsive, to persist typically compounds the problem. By the time manual intervention is attempted, it is often very difficult to identify the appropriate corrective action. Appropriate SFM specifications enable systems in the sysplex to take corrective action automatically. In general, each parameter arises out of real world situation that led to some sort of (usually quite ugly) outage.


Sysplex Failure Management – z/OS 1.11

SFM with BCPii

- Expedient removal of unresponsive or failed systems is essential to high availability in sysplex
- XCF exploits new BCPii services to:
 - Detect failed systems
 - Reset systems
- Benefits:
 - Improved availability by reducing duration of sympathy sickness
 - Eliminate manual intervention in more cases
 - Potentially prevent human error that can cause data corruption



30

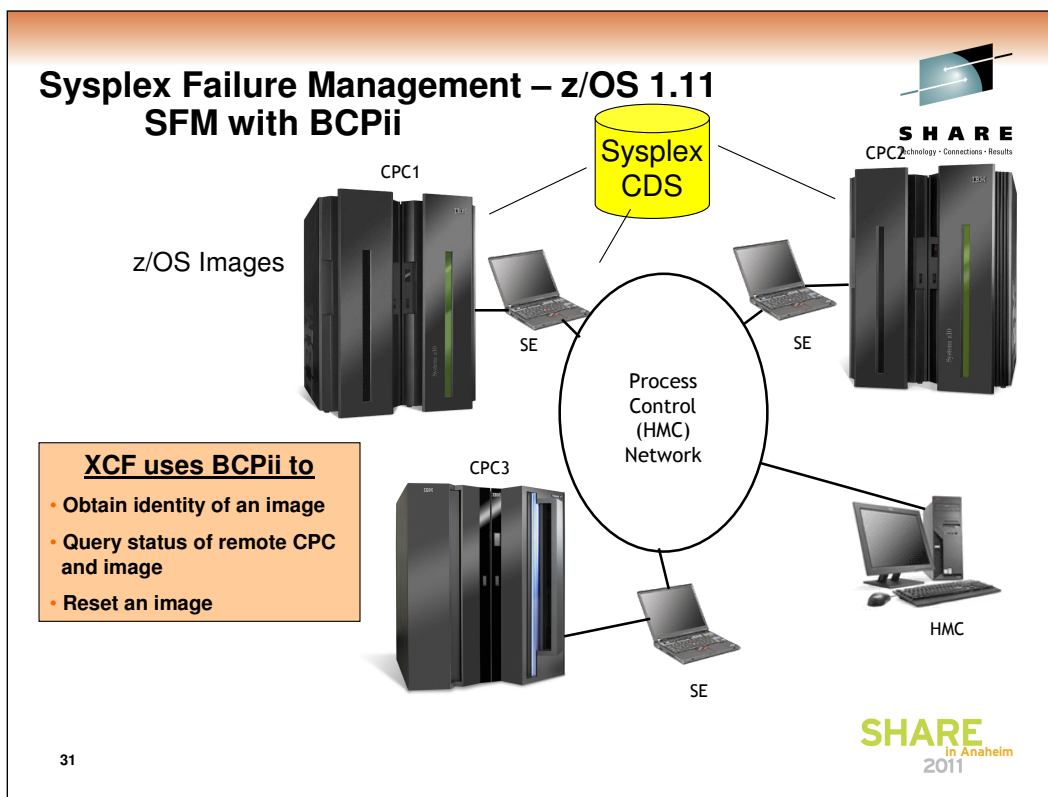


The sysplex failure management (SFM) component of XCF, which is used to manage failures and reconfiguration in the sysplex, has been enhanced in z/OS V1.11. It is now designed to use new Base Control Program internal interface (BCPii) services to determine whether an unresponsive system has failed, expedite sysplex recovery by bypassing delay intervals when possible, and automatically reset failed systems without manual intervention. This function allows SFM to avoid waiting for a period of time before assuming that systems have failed, improves the responsiveness of failure management, avoids operator intervention, and helps limit or avoid sysplex-wide slowdowns that can result from single-system failures.

The Base Control Program Internal Interface (BCPii) component of z/OS provides a set of programming interfaces to allow authorized programs to perform Hardware Management Console (HMC) functions for System z servers within an attached HMC network. These operations include obtaining information about servers and images (LPARs), issuing commands for certain hardware and software-related functions, and listening for certain hardware and software events. BCPii communication to HMCs and Support Elements (SEs) uses internal communication protocols and does not require communication on an IP network. Therefore, it is isolated from other network traffic.

Through the use of BCPii, XCF can detect that a system has entered a non-restartable wait-state, or that it has been re-IPLed. XCF can also perform a system reset on other systems in the sysplex. Thus XCF now has the ability to ascertain with certainty that a system is no longer operational. With this certain knowledge, XCF can ensure that the system is safely isolated from shared resources and remove the failed system from the sysplex – all without operator involvement. Furthermore, since XCF need not wait for the system failure detection interval to expire (to conclude that the system has no signs of life), the isolation of the failed system can occur sooner, which in turn reduces the amount of time that other systems in the sysplex will experience sympathy sickness.

BCPii is available on z/OS V1.10 with PTF UA47493, and requires a System z9 or z10 server with a microcode update. (For more information about required hardware levels, see the appropriate PSP bucket.) In z/OS V1.11, BCPii is designed to allow authorized programs to change or set data for certain HMC-managed objects associated with a CPC, Image, or Capacity Record. In addition, support for interactions with activation profile attributes is planned to be made available with the PTF for APAR OA29638 in first quarter of 2010.



CPC – Central Processor Complex containing images (LPARs)
SE – Support Element
HMC – Hardware Management Console


The Base Control Program internal interface (**BCPii**) allows authorized z/OS applications to have HMC-like control over systems in the process control HMC network. Note that there is complete communication isolation of existing networks (internet/intranet) from the process control (HMC) network, and communication with the System z support element is completely within base z/OS. BCPii provides a set of authorized APIs to enable communications between z/OS applications and the local support element, as well as between other support elements connected to other CPCs routed by the HMC. The BCPii query services can provide information about the operational state of any CPC connected to the HMC network, as well as the operational state of any image on the CPC.


As each z/OS image IPLs into the sysplex, XCF sets an IPL token in the hardware to uniquely identify the image. The IPL token is also published in the sysplex couple data set so that each system in the sysplex can ascertain the IPL token for every other system. If a system appears to be unresponsive, XCF uses BCPii query services to inquire as to the state of the subject system. If the system is down, it will be removed from the sysplex. As needed, XCF will use BCPii services to reset the system. For example, a system reset might be needed to ensure that the system has been successfully isolated from the sysplex. The IPL token is used when doing such resets, as it ensures that the reset is applied to the intended instance of the system image

Sysplex Failure Management – z/OS 1.11


SFM with BCPii

- With BCPii, XCF can know that system is dead, and:
 - Bypass the Failure Detection Interval (FDI)
 - Bypass the Indeterminate Status Interval (ISI)
 - Bypass the cleanup interval
 - Reset the system even if fencing fails
 - Avoid IXC102A, IXC402D and IXC409D manual intervention
 - Validate “down” to help avoid corruption of shared data





32



Unresponsive systems must be partitioned from the sysplex in a timely manner to avoid sympathy sickness. But, without BCPii, XCF does not really *know* a system's operational state. At best, systems can monitor each other for signs of activity (updates to the sysplex couple data set, signals being exchanged). When the monitored activity stops, XCF waits the Failure Detection Interval (FDI) to try to avoid falsely removing an operational system. One would not want remove a system that was suffering a temporary problem. But the penalty for this caution is that the sysplex may suffer workload processing interruptions/delays if the system has truly failed.


A partitioned system must be isolated from the rest of the sysplex to avoid corruption of shared data. In a parallel sysplex, XCF relies on CF Isolate command to fence a system from the channel subsystem. If the installation does not have a CF, or if the isolation fails, manual system reset is required. Manual intervention elongates the partitioning process, which elongates sympathy sickness.

But with BCPii services, XCF can now detect that a system has failed, and if so, whether it has been reset or otherwise isolated from shared resources. With this certain knowledge, XCF can safely remove a failed system from the sysplex without waiting for the failure detection interval to expire. If the failed system has not been isolated, XCF need not wait for the indeterminate isolation interval (ISI) to expire before it attempts to fence the failed system. If the fencing fails (or cannot be performed due to a lack of a CF), XCF can use BCPii services to appropriately reset the failed system. In cases where the failed system has been appropriately reset, XCF need not perform any isolation actions.

Thus the certain knowledge that a system has failed enables XCF to immediately partition failed systems from the sysplex, all without operator intervention.

ISI – can behave like ISOLATETIME(0) regardless of ISOLATETIME value specified in policy


Sysplex Failure Management – z/OS 1.11 SFM with BCPii



- SFM will automatically exploit BCPii and as soon as the required configuration is established:
 - Pairs of systems running z/OS 1.11 or later
 - BCPii configured, installed, and available
 - XCF has security authorization to access BCPii defined FACILITY class resources
 - z10 GA2 with appropriate MCL's, or z196
 - New version of sysplex CDS is primary in sysplex
 - Toleration APAR OA26037 for z/OS 1.9 and 1.10
 - Does NOT allow systems to use new SSD function or protocols

Enabling SFM to use BCPii will have a big impact on availability. Make it happen !

33



•See topic "Assigning the RACF TRUSTED attribute" in *MVS Initialization and Tuning Reference* for information on using RACF to assign the TRUSTED attribute to the XCF address space.

•Refer to the "BCPii Setup and Installation" topic in *MVS Programming: Callable Services for High Level Languages* for information on installation and configuration steps and SAF authorization requirements to enable BCPii to invoke z/Series Hardware APIs.

•A system running on z/OS V1R11 and down-level hardware is only eligible to target other systems that are enabled to exploit the full functionality of the SSD partitioning protocol. A system not running on the requisite hardware can not be the target of SSD partitioning protocol functions.

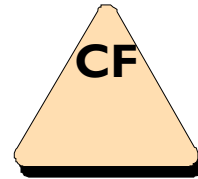
•Install toleration PTFs for OA26037 on V1R10 and V1R9 systems in the sysplex to use the newly formatted sysplex couple data set required by the protocol.

•By default, the SYSSTATDETECT function is enabled in V1R11. The current setting of the SYSSTATDETECT function can be determined by issuing a DISPLAY XCF,COUPLE command.


For more information on enabling or disabling the SYSSTATDETECT function in V1R11, see *MVS Initialization and Tuning Reference* for information on specifying SYSSTATDETECT in the COUPLExx parmlib member and *MVS System Commands* for information on enabling and disabling the SYSSTATDETECT function via the SETXCF FUNCTIONS command

Agenda

- Hardware Updates
 - CFCC Level 17
 - CFCC Level 16
 - Parallel Sysplex InfiniBand Links
- z/OS Updates
 - Sysplex Failure Management
 - **z/OS V1R12**
 - z/OS V1R13 preview
- Summary




34



z/OS 1.12

- **Critical Members**
- **CFSTRHANGTIME**
- **REALLOCATE**
- Support for CFLEVEL 17
- Health Checks
- Auto Reply
- Run Time Diagnostics
- XCF Programming Interfaces

35



z/OS 1.12 functionality that may be of interest from a sysplex perspective.

Due to time restrictions, we may only be able to discuss the topics in bold. However slides for the remaining topics are included for your consideration. Feel free to email the speaker if you have additional questions or comments.

z/OS 1.12 - Critical Members



- A system may appear to be healthy with respect to XCF system status monitoring, namely:
 - Updating status in the sysplex CDS
 - Sending signals
- But is the system actually performing useful work?
- There may be critical functions that are non-operational
- Which in effect makes the system unusable, and perhaps induces sympathy sickness elsewhere in the sysplex
- Action should be taken to restore the system to normal operation OR it should be removed to avoid sympathy sickness



36

z/OS 1.12 extends XCF System Status monitoring to incorporate status information about critical components (such as GRS). Currently, a system is deemed unresponsive if it stops sending XCF signals and stops updating its status in the sysplex Couple Data Set (CDS). However, these indications of activity do not necessarily imply that a system is able to accomplish useful work. Indeed, an apparently active system could in effect be causing sympathy sickness because critical components are unable to accomplish their intended function. The goal of the “critical member” support is to resolve the sympathy sickness by expeditiously partitioning a sick system out of the sysplex whenever any critical XCF member on that system is deemed unresponsive. Though still not a perfect indicator of whether a system is performing useful work, the discovery of unresponsive critical components should provide an incremental improvement that helps the sysplex better identify (and remove) unresponsive systems.

z/OS 1.12 - Critical Members ...



- A Critical Member is a member of an XCF group that identifies itself as “critical” when joining its group
- If a critical member is “impaired” for long enough, XCF will eventually terminate the member
 - Per the member’s specification: task, space, or system
 - SFM parameter MEMSTALLTIME determines “long enough”
- GRS is a “system critical member”
 - XCF will remove a system from the sysplex if GRS on that system becomes “impaired”



37

z/OS 1.12 extends XCF Member Status monitoring to take some form of action when a critical XCF group member appears to be non-operational. XCF externalizes via messages that the member is “impaired”. Furthermore, XCF will terminate the critical member if the impaired state persists long enough. Surfacing the condition should make it easier to identify situations where an application may not be operating normally. Termination of the critical member should relieve the sympathy sickness condition and allow the application to resume normal operation. Alternatively, such termination may also make it possible for more timely restart of the application (or other appropriate recovery action) that can then lead to full recovery. There is some danger that such termination could negatively impact the application, however, that is something for the application writer to assess and exploit as appropriate. The application determines whether it is “critical” and if so, the means by which it should be terminated. Said termination could entail termination of the member’s task, address space, or system. If the system is to be terminated, the member is presumed to be “system critical”. GRS is “system critical”.

If a critical member remains continuously impaired for as long as the system failure detection interval (FDI), XCF inspects the Sysplex Failure Manager (SFM) specification for the MEMSTALLTIME parameter. For MEMSTALLTIME(NO), XCF delays termination of the member for FDI seconds, or two minutes, whichever is longer. If MEMSTALLTIME(nnn) is specified, XCF delays termination for the indicated number of seconds).

This function is intended to help reduce the incidence of sysplex-wide problems that can result from unresponsive critical components. GRS exploits these XCF critical member functions in both ring and star modes. GRS monitors its key tasks and notifies XCF if it detects that GRS is impaired.

z/OS 1.12 - Critical Members ...



- New Messages
 - IXC633I “member is impaired”
 - IXC634I “member no longer impaired”
 - **IXC635E “system has impaired members”**
 - IXC636I “impaired member impacting function”
- Changed Messages
 - IXC431I “member stalled” (includes status exit)
 - IXC640E “going to take action”
 - IXC615I “terminating to relieve impairment”
 - IXC333I “display member details”
 - IXC101I, IXC105I, IXC220W “system partitioned”



38

This slide summarizes the essence of the new and changed messages related to impaired member processing.

Installations may choose to develop automation and/or operational procedures to deal with impaired member conditions. The Sysplex Failure Management (SFM) policy MEMSTALLTIME specification should be specified accordingly. For example, if an installation wants operators to investigate and resolve such problems, one will likely specify a longer MEMSTALLTIME value to allow time for such actions to occur. MEMSTALLTIME in effect serves as a back stop to allow the system to take automatic action to alleviate the problem if the operations personnel are unable to resolve the problem in a timely manner. Of course one should recognize that the higher the MEMSTALLTIME value the longer the potential sympathy sickness impact on other systems in the sysplex will persist.

Message IXC635E is likely the key message that would be used to trigger the relevant automation and/or operational procedures.

The sysplex partitioning messages (IXC101I “system being removed”, IXC105I “system has been removed”, and IXC220W “XCF wait-stated system”) have new inserts to indicate that the system was removed as the result of terminating an impaired member. The XCF wait-state code 0A2 has a new reason code (x194) to indicate this condition as well.

z/OS 1.12 - Critical Members ...



- **Coexistence considerations**

- Toleration APAR OA31619 for systems running z/OS V1R10 and z/OS V1R11 should be installed before IPLing z/OS V1R12
- The APAR allows the down level systems to understand the new sysplex partitioning reason that is used when z/OS V1R12 system removes itself from the sysplex because a system critical component was impaired
- If the APAR is not installed, the content of the IXC101I and IXC105I messages will be incorrect



39

The z/OS V1R12 system has a new sysplex partitioning reason code and a new 0A2 wait-state reason when a system is removed from the sysplex. We now have so many reasons for killing systems that we have exhausted the internal data structures that were used to share this information among systems in the sysplex. To add the new “impaired member” reason, we had to make changes that will cause down-level systems to issue partitioning messages with completely misleading inserts. For example, a down level system would issue the following messages if a z/OS V1R12 system was removed from the sysplex to terminate an impaired critical member:

```
IXC101I SYSPLEX PARTITIONING IN PROGRESS FOR SY4 REQUESTED BY
XCFAS. REASON: LOSS OF COUPLE DATA SET
```

```
IXC105I SYSPLEX PARTITIONING HAS COMPLETED FOR SY4
```

```
- PRIMARY REASON: LOSS OF COUPLE DATA SET
```

```
- REASON FLAGS: 800015
```

The toleration APAR allows the down level systems to issue messages with the correct text. For example:

```
IXC101I SYSPLEX PARTITIONING IN PROGRESS FOR SY4 REQUESTED BY
XCFAS. REASON: SYSTEM HAS AN IMPAIRED CRITICAL MEMBER
```

We believe the only detrimental impact of running without the toleration APAR is the misleading messages. Still, we recommend that it be installed before a z/OS V1R12 system is IPL'ed into the sysplex.

z/OS 1.12 - Critical Members ...



- **Potential migration action**
 - Evaluate, perhaps change MEMSTALLTIME parameter



40

If you do not currently have an active SFM policy, or your SFM policy does not specify MEMSTALLTIME, MEMSTALLTIME(NO) is the default. With MEMSTALLTIME(NO), SFM will terminate an impaired critical member after the system failure detection interval (FDI) or two minutes, whichever is greater. The default FDI on a z/OS V1 R12 system is likely 165 seconds. You can issue the DISPLAY XCF,COUPLE command to see the FDI (aka INTERVAL) value that is being used by your system.

If your active SFM policy specifies MEMSTALLTIME(n) where “n” is some integral number of seconds, that value “n” will determine the number of seconds that SFM waits before it terminates an impaired critical member that is deemed to be impacting its function (and thus the system, and thus the sysplex). This specification is likely suitable for z/OS V1R12 as well. The rationale used to pick a MEMSTALLTIME value for dealing with signalling sympathy sickness conditions is likely valid for critical member support as well. Namely, a situation has occurred in which the system has recognized that there might be a sympathy sickness impact. MEMSTALLTIME indicates how long XCF should delay before taking action to alleviate the condition (i.e. terminate the member). However much time was needed for automation and/or operational procedures to run their course for resolving a signalling sympathy sickness problem is very likely the same as would be needed to resolve an impaired member problem. Installations that want to preserve past behavior to the greatest extent possible, which is to say, they want to prevent the system from terminating impaired critical members, will need to create and activate an SFM policy with a MEMSTALLTIME specification.

It is NOT possible to completely disable the termination of impaired critical members. However, by specifying a large MEMSTALLTIME value, one can in effect delay the action for so long that it is unlikely to be taken. One would expect one of two things to have happened before the MEMSTALLTIME value expires, either (1) the system resumes normal behavior, or (2) the system is re-IPLed because the “sick but not dead” issues effectively rendered the system unusable.

If you want/need to set an SFM policy MEMSTALLTIME specification, then depending on what you already have set up, you might need to:

- Run the IXCL1DSU utility to create a couple data set that will be used to hold SFM policies
- Run the IXCMIAPU utility to create SFM policies with the desired MEMSTALLTIME value
- Make the SFM CDS available to the sysplex (COUPLExx or SETXCF COUPLE)
- Start the desired SFM policy (SETXCF START,POLICY,TYPE=SFM,POLNAME=xxx)

z/OS 1.12 - CFSTRHANGTIME



- Connectors to CF structures need to participate in various processes and respond to relevant events
- XES monitors the connectors to ensure that they are responding in a timely fashion
- If not, XES issues messages (IXL040E, IXL041E) to report the unresponsive connector
- Users of the structure may hang until the offending connector responds or is terminated

41

SHARE
in Anaheim
2011

The XES hang detect function was introduced in OS/390 V1R8 (HBB6608) to report cases when an expected response to a structure-related event is not received in a timely manner. After 2 minutes without a response, XES issues IXL040E or IXL041E to identify the unresponsive connector, the associated structure, the event, and the affected process. Installations often fail to react to these messages, or worse, react by terminating the wrong connector.

IXL040E CONNECTOR NAME: conname, JOBNAME: jobname, ASID: asid
HAS NOT responsetext
process
FOR STRUCTURE strname CANNOT CONTINUE.
MONITORING FOR RESPONSE STARTED: mondate montime
DIAG: x x x

IXL041E CONNECTOR NAME: conname, JOBNAME: jobname, ASID: asid
HAS NOT RESPONDED TO THE event FOR
SUBJECT CONNECTION: subjectconname.
process
FOR STRUCTURE strname CANNOT CONTINUE.
MONITORING FOR RESPONSE STARTED: mondate montime
DIAG x x x

z/OS 1.12 – CFSTRHANGTIME ...



- CFSTRHANGTIME
 - A new SFM Policy specification
 - Indicates how long the system should allow a structure hang condition to persist before taking corrective action(s) to remedy the situation
- Corrective actions may include:
 - Stopping rebuild
 - Forcing the user to disconnect
 - Terminating the connector task, address space, or system



42

The existing XCF/XES CF structure hang detect support is extended by providing a new CFSTRHANGTIME SFM Policy option that allows you to specify how long CF structure connectors may have outstanding responses. When the time is exceeded, SFM is designed to drive corrective actions to try to resolve the hang condition. This helps you avoid sysplex-wide problems that can result from a CF structure that is waiting for timely responses from CF structure connectors.

The IXCMIAPU policy utility provides a new keyword CFSTRHANGTIME for the SFM policy:

```
DEFINE POLICY NAME(TEST) CONNFALL(YES) REPLACE(YES)
SYSTEM NAME(*)
ISOLATETIME(0)
WEIGHT(10)
CFSTRHANGTIME(900)
```

The interval specified with the CFSTRHANGTIME keyword begins after a hang is recognized, approximately 2 minutes after the delivery of the event that requires a response. CFSTRHANGTIME(0) means that the system is to take action immediately upon recognizing that the response is overdue. The initial suggestion, documented by way of the new XCF_SFM_CFSTRHANGTIME health check, is to set CFSTRHANGTIME at 5 minutes, but was increased by APAR OA34439 increased the value to 15 minutes. This allows time for the installation to evaluate the situation and decide whether to take manual action, and possibly allow the hang to clear spontaneously, while not permitting the hang to persist long enough to cause sysplex-wide problems.


Initial action is taken at expiration of CFSTRHANGTIME interval. If hang persists, escalates to more aggressive actions. The escalation hierarchy begins with the least disruptive actions and progresses to more disruptive actions. Actions include: stop rebuild, stop signaling path (XCF signaling only), force disconnect (XCF signaling only), terminate connector task, terminate connector address space, partition connector system. Each system acts against its own connectors (no system will take action against any other system). No attempt to evaluate causal relationships between multiple events is made.


z/OS 1.12 – CFSTRHANGTIME ...

New Messages

IXL049E HANG RESOLUTION ACTION FOR CONNECTOR NAME: conname
TO STRUCTURE strname, JOBNAME: jobname, ASID: asid:
actiontext

IXL050I CONNECTOR NAME: conname TO STRUCTURE strname,
JOBNAME: jobname, ASID: asid
HAS NOT PROVIDED A REQUIRED RESPONSE AFTER noresponsetime SECONDS.
TERMINATING termtarget TO RELIEVE THE HANG.





43

If no SFM policy is active or the policy specifies CFSTRHANGTIME(NO), no hang relief action is taken. If a policy is started, stopped, or changed, the monitor will re-evaluate the required response and possibly reissue IXL049E.

IXL049E may indicate that:

- (1) The system will not take action because (a) there is no SFM policy, (b) the SFM policy requires manual intervention, or (c) the system has tried everything it knows how to do, or
- (2) The system is taking action now (either because the policy specified CFSTRHANGTIME(0) or it was changed in a way that makes the action past due), or
- (3) The system will take action at the time specified in the message.

For IXL049E, *actiontext* is one of:

SFM POLICY NOT ACTIVE, MANUAL INTERVENTION REQUIRED.
SFM POLICY REQUIRES MANUAL INTERVENTION.
SYSTEM IS TAKING ACTION.
SYSTEM WILL TAKE ACTION AT termdate termtime
SYSTEM ACTION UNSUCCESSFUL, MANUAL INTERVENTION REQUIRED

For IXL050I *termtarget* is one of:

REBUILD
SIGNAL PATHS (ATTEMPT 1)
SIGNAL PATHS (ATTEMPT 2)
SIGNAL PATHS (ATTEMPT 3)
CONNECTION
CONNECTOR TASK
CONNECTOR SPACE (WITH RECOVERY)
CONNECTOR SPACE (NO RECOVERY)
CONNECTOR SYSTEM

z/OS 1.12 – CFSTRHANGTIME ...



- Initiates diagnostic dump as appropriate
- APAR OA34440 (z/OS 1.12 only) will take the dump sooner rather than later



44

XES will take only one dump per set of monitored events. If monitoring multiple events, any one or any combination of them may have caused or contributed to the hang, but since we can't analyze those possible relationships there is no point in taking repeated dumps after the hang is first recognized. Once a dump is taken, no further dumps will be taken until all events being monitored have been processed.

If the hang action analysis concludes that automatic action will not be taken to relieve the hang, a dump is taken on the same analysis cycle. Except for the case of policy changes, that would be on the same cycle in which the hang is recognized, which implies that the dump is taken after issuing the IXL040E / IXL041E and IXL049E messages.

If the next anticipated hang relief action is termination of the connector's task or "soft" termination of the connector's address space (i.e., task recovery will be allowed to run), the connector's recovery will presumably take a dump. We expect that an application dump would be more valuable in debugging root cause than a XES dump, so we avoid taking a dump that might inadvertently prevent capture of the application's dump. Only if we believe that there is sufficient time to capture a XES dump before initiating action will we do so.

If the next anticipated action is not expected to produce a connector dump, XES will take one.

For down-level releases, OA28298 introduced dumping of connector address and data spaces at hang recognition time.

Dump title is tailored to indicate a connector issue:

ABEND=026, REASON=08118001, CONNECTOR HANG: CONNAME=conname,
JOBNAME=jobname

APAR OA34440 changed the dump timing so that if the system intends to initiate a corrective action in the future, a dump is taken as soon as the hang is recognized (unless it would interfere with an expected connector dump). The original behavior was to wait until close to the time of automatic action before dumping. Taking the dump closer to the point of the hang provides for better first failure data capture.

Background - REALLOCATE



- SETXCF START,REALLOCATE
 - Well-received, widely exploited for CF structure management
 - For example, to apply “pure” CF maintenance:
 - SETXCF START,MAINTMODE,CFNAME=cfname
 - SETXCF START,REALLOCATE to move structures out of CF
 - Perform CF maintenance
 - SETXCF STOP,MAINTMODE,CFNAME=cfname
 - SETXCF START,REALLOCATE to restore structures to CF

SHARE
in Anaheim
2011

45

The SETXCF REALLOCATE command is an existing system command used for CF structure management. The command causes the Coupling Facility Resource Manager (CFRM) component of XCF/XES to analyze existing coupling facility structures with respect their placement in various CFs. The command then effects the necessary changes to correct any structure-related problems that it finds that are within its “scope of expertise.” As appropriate, the command may initiate actions (such as stop duplexing, rebuild, and/or start duplexing) in an orderly fashion to correct any problems it finds.

When a coupling facility is in maintenance mode, no new structures will be allocated in the CF and REALLOCATE will move allocated structures to alternate CF per the preference list in the CFRM policy.

The SETXCF START,MAINTMODE command, which is available as of z/OS V1R9, in conjunction with the SETXCF START,REALLOCATE command can be used in combination to greatly simplify planned reconfiguration and maintenance actions. In particular, one need not update the CFRM policy in order to prevent structures from being allocated in the CF. In the example illustrated in the slide, we have a case where (disruptive) CF maintenance is to be applied. The subject CF is placed in “maintenance mode” so that CFRM will refrain from allocating new structures in the CF. The REALLOCATE command takes maintenance mode into account, and concludes that structures need to be moved out of the subject CF. REALLOCATE will take the necessary actions to get the structures moved to an alternate CF (per the preference list).

Background - REALLOCATE



But...

- Difficult to tell what it did
 - Long-running process
 - Messages scattered all over syslog
 - Difficult to find and deal with any issues that arose
- And people want to know in advance what it will do

46

SHARE
in Anaheim
2011

The existing SETXCF REALLOCATE command causes CFRM to analyze all allocated structures in the sysplex with respect to their placement in various CFs, and initiate the necessary changes to correct any structure-related problems that it finds. Each structure is successively evaluated and processed (as needed) one at a time so as to minimize disruption to the sysplex. As each structure is processed, REALLOCATE issues messages to describe its decisions and actions. In some cases, REALLOCATE is unable to successfully resolve a structure, and issues messages to so indicate. It is often the case that some form of manual intervention is needed in order to accomplish the desired reallocate of those structures. Prior to z/OS V1R12, the installation would have to search logs to find the messages for structures that had issues – a rather tedious task.

Furthermore, installations have also wanted to know in advance what actions reallocate was going to take. Depending on the structures to be manipulated, it might for example be desirable to delay the reallocate to a “slow” time of day so as to minimize the disruption to the exploiting application.

z/OS 1.12 - REALLOCATE



- DISPLAY XCF,REALLOCATE,option
- TEST option
 - Provides detailed information regarding what REALLOCATE would do if it were to be issued
 - Explains why an action, if any, would be taken
- REPORT option
 - Provides detailed information about what the most recent REALLOCATE command actually did do
 - Explains what happened, but not why

47

SHARE
in Anaheim
2011


In z/OS V1.12 a new DISPLAY XCF,REALLOCATE,TEST command simulates the reallocation process and provides information about the changes that REALLOCATE (were it to be initiated by the SETXCF START,REALLOCATE command) would attempt to make, as well as any errors that it might encounter. This capability will provide information you can use to decide when to invoke the actual REALLOCATE process, and also whether you might need to make coupling facility configuration changes before issuing the actual REALLOCATE command.

For TEST, the CFRM policy is read into local storage (and it is not written back). The same processing that a real REALLOCATE would do is then applied against the in-store copy of the policy.

A new DISPLAY XCF,REALLOCATE,REPORT command will provide detailed information on the results experienced by a previously executed REALLOCATE command. This capability is intended to help you find such information without searching through the system log for REALLOCATE related processing and exception messages.

An actual REALLOCATE process stores “history” in the CFRM CDS for each structure defined in the policy. D XCF,REALLOC,REPORT reads the data and builds message text to reflect those results.

REALLOCATE TEST Example (part 1)



```

D XCF,REALLOC,TEST
IXC347I 10.31.05 DISPLAY XCF

```

COUPLING FACILITY STRUCTURE ANALYSIS PERFORMED FOR REALLOCATE TEST.


STRUCTURE(S) WITH AN ERROR/EXCEPTION CONDITION

NONE

STRUCTURE(S) WITH A WARNING CONDITION

NONE

results from a simulated REALLOCATE


48


The TEST output has four sections. The first section lists any structures for which there would be issues, which makes it easy to find the ones that might require intervention. The report will also indicate what manual actions (if any) might be needed to deal with these exceptions.

What sort of issues might there be?

- Structures that would not be reallocated to reside in their “most preferred” CF(s) for some reason
- Structures that would not be re-duplexed as they were supposed to be
- Pending policy changes that would not be cleared by the reallocate process and therefore would remain pending even after the completion of reallocate processing

REALLOCATE TEST Example (part 2)



```

STRUCTURE(S) REALLOCATED SUCCESSFULLY


STRNAME: BIGONE                                INDEX: 38
SIMPLEX STRUCTURE ALLOCATED IN CF(S) NAMED: LF02
CFNAME      STATUS/FAILURE REASON
-----
LF01        PREFERRED CF 1
                                INFO110: 00000003 AC007800 00010011
LF02        PREFERRED CF ALREADY SELECTED
                                INFO110: 00000003 AC007800 00020011

1 REALLOCATE STEP(S): REBUILD
-----
STRUCTURE(S) ALREADY ALLOCATED IN PREFERRED CF(S)

...

```

49



In the second section of the report, we find information about structures that TEST would expect REALLOCATE to successfully process. First it lists each structure for which REALLOCATE would be expected to take action. For these structures, the output explains what actions would be taken against the structure. It also lists each CF that was considered and explains why the various CF's would or would not have been selected. Then it lists the structures for which REALLOCATE would not be expected to take action. Again similar information about the relevant CF's would be presented.

The INFO110 is diagnostic information similar to what is provided with message IXC574I. Intended for use by IBM service personnel as needed.

REALLOCATE TEST Example (part 3)



COUPLING FACILITY STRUCTURE ANALYSIS OUTPUT FOR REALLOCATE TEST

```
CFNAME: LF01
COUPLING FACILITY      :  SIMDEV.IBM.EN.ND0100000000
                        :  PARTITION: 00  CPCID: 00

CONNECTED SYSTEM(S):
SY1      SY2      SY3

ACTIVE STRUCTURE(S):
BIGONE           CACHE01 (OLD)           CACHE02 (OLD)
CACHE12          CACHE128                CACHE16
CACHE256         CACHE32                 CACHE64
```

...

Output is similar to message IXC362I from DISPLAY XCF,CF,CFNAME=ALL and shows approximately what that message would look like AFTER performing the REALLOCATE.

50

SHARE
in Anaheim
2011

In the third section, the TEST output summarizes on a CF by CF basis, the collection of structures that would be expected to reside in each CF after the proposed REALLOCATE was finished.

REALLOCATE TEST Example (part 4)



```
REALLOCATE TEST RESULTED IN THE FOLLOWING:
  1 STRUCTURE(S) REALLOCATED - SIMPLEX
  0 STRUCTURE(S) REALLOCATED - DUPLEXED
  0 STRUCTURE(S) POLICY CHANGE MADE - SIMPLEX
  0 STRUCTURE(S) POLICY CHANGE MADE - DUPLEXED
  6 STRUCTURE(S) ALREADY ALLOCATED IN PREFERRED CF - SIMPLEX
  2 STRUCTURE(S) ALREADY ALLOCATED IN PREFERRED CF - DUPLEXED
  0 STRUCTURE(S) NOT PROCESSED
102 STRUCTURE(S) NOT ALLOCATED
 17 STRUCTURE(S) NOT DEFINED
-----
128 TOTAL

  0 STRUCTURE(S) WITH AN ERROR/EXCEPTION CONDITION
```

51

SHARE
in Anaheim
2011

In the final section, the TEST output summarizes what it expects would have happened if the proposed REALLOCATE were initiated.

z/OS 1.12 – REALLOCATE ...



Caveats for REALLOCATE TEST option

- Actual REALLOCATE could have different results
 - Environment could change
 - For structures processed via user-managed rebuild, the user could make “unexpected” changes
 - Capabilities of systems where REALLOCATE runs differ from the system where TEST ran
 - For example, connectivity to coupling facilities
- TEST cannot be done:
 - While a real REALLOCATE (or POPCF) is in progress
 - If there are no active allocated structures in the sysplex




52

The major things that could cause the TEST results to be different from the actual REALLOCATE:

- The environment could change between the TEST and the actual REALLOCATE. For example, the CFRM policy might be changed, or the set of connected coupling facilities might change.
- User-managed rebuild could do something "unexpected". When REALLOCATE initiates a user-managed rebuild, it has no real control over what the exploiter will actually do during the rebuild. For example, certain structure attributes and/or parameters could be changed by the user as part of the rebuild. Since TEST cannot know what the exploiter will really do during its processing, it makes a reasonable guess. If the guess is sufficiently divergent from what the exploiter actually does during the rebuild, the TEST results could differ from what the real REALLOCATE ultimately achieves.
- TEST and REALLOCATE might run on systems with different capabilities. TEST runs on exactly one system in the sysplex. REALLOCATE processing can run on different systems as it makes progress. Thus it could be that the system that processed the TEST has connectivity to a different set of CFs than one or more of the systems that participate in the actual REALLOCATE. Those differences could cause the REALLOCATE to put a structure in a different CF than the TEST expected.

TEST recognizes when it would not be worthwhile for it to run. If a REALLOCATE is currently in progress, the state of the structures is in flux and TEST cannot really make reliable predictions.

REALLOCATE REPORT Example (part 1)



```

D XCF,REALLOC,REPORT
IXC347I 10.37.45 DISPLAY XCF

THE REALLOCATE PROCESS STARTED ON 08/07/2009 AT 10:31:23.98.
THE REALLOCATE PROCESS ENDED ON 08/07/2009 AT 10:36:09.81.
-----
STRUCTURE(S) WITH AN ERROR/EXCEPTION CONDITION


NONE
-----
STRUCTURE(S) WITH A WARNING CONDITION

NONE
-----

```

results from a real REALLOCATE


53



The REPORT has three sections. The first section lists any structures for which reallocate had issues, thus making it easier to find the ones that require intervention. The report will also indicated what manual actions (if any) are needed to deal with these exceptions.

What sort of issues might there be?

- Structures that could not be reallocated to reside in their “most preferred” CF(s) for some reason
- Structures that could not be re-duplexed as they were supposed to be
- Pending policy changes that could not be cleared by the reallocate process and therefore remained pending even after the completion of reallocate processing



REALLOCATE REPORT Example (part 2)

```

STRUCTURE(S) REALLOCATED SUCCESSFULLY


STRNAME: CACHE01                                INDEX: 2
  3 REALLOCATE STEP(S): KEEP=OLD, REBUILD, DUPLEX
  COMPLETED ON SYSTEM SY1 ON 08/07/2009 AT 10:31:40.01.

STRNAME: CACHE02                                INDEX: 6
  3 REALLOCATE STEP(S): KEEP=OLD, REBUILD, DUPLEX
  COMPLETED ON SYSTEM SY1 ON 08/07/2009 AT 10:31:53.03.
-----
STRUCTURE(S) ALREADY ALLOCATED IN PREFERRED CF(S)

STRNAME: IXC2                                    INDEX: 22
EVALUATED ON SYSTEM SY1 ON 08/07/2009 AT 10:32:32.60.
-----

```

REALLOCATE processes structures in INDEX order




54

The second section of the report shows what actions, if any, were taken by reallocate. There are two subsections, one showing structures that were manipulated by reallocate, and another showing the structures for which no action was needed. In this example, we see that reallocate stopped the duplexed rebuild for CACHE01 and CACHE02, rebuilt them, and then re-established duplexing. No action was taken against structure IXC2.


The CFRM policy in effect maintains a table of structures, with one entry per structure. Reallocate processes the structures in index order. Note that reallocate only processes structures that are physically allocated in some coupling facility.

REALLOCATE REPORT Example (part 3)



```
REALLOCATE PROCESSING RESULTED IN THE FOLLOWING:  
91 STRUCTURE(S) REALLOCATED - SIMPLEX  
8 STRUCTURE(S) REALLOCATED - DUPLEXED  
0 STRUCTURE(S) POLICY CHANGE MADE - SIMPLEX  
0 STRUCTURE(S) POLICY CHANGE MADE - DUPLEXED  
1 STRUCTURE(S) ALREADY ALLOCATED IN PREFERRED CF - SIMPLEX  
0 STRUCTURE(S) ALREADY ALLOCATED IN PREFERRED CF - DUPLEXED  
0 STRUCTURE(S) NOT PROCESSED  
12 STRUCTURE(S) NOT ALLOCATED  
16 STRUCTURE(S) NOT DEFINED  
  
-----  
128 TOTAL  
  
0 STRUCTURE(S) WITH AN ERROR/EXCEPTION CONDITION  
  
0 STRUCTURE(S) MISSING PREVIOUS REALLOCATE DATA
```

55



The final section of the report summarizes everything that reallocate did or discovered.

There may be structures missing reallocate data if structures were removed from the CFRM policy since the last REALLOCATE started. It may also be missing if the last REALLOCATE that was done was performed in a sysplex that did not have all z/OS V1R12 and/or the necessary coexistence PTFs installed on systems running lower releases.

z/OS 1.12 – REALLOCATE ...



Caveats for REPORT option

- Can be done during or after a real REALLOCATE (but not before a real REALLOCATE is started)
- A REPORT is internally initiated by XCF if a REALLOCATE completes with exceptions



56

Since the REPORT option gathers the data from the CFRM policy that describes the results of the most recent REALLOCATE, if the installation never performed a REALLOCATE, there would be nothing to report. Changes to the policy that result in structures being deleted will also delete the related history records. If a REPORT is requested while a real REALLOCATE is in progress, structures that have not yet been processed would appear in the “warning” section of the report.

If REALLOCATE completes with exceptions, XCF will internally initiate a DISPLAY XCF,REALLOCATE,REPORT request so that the results can be neatly summarized in the log. A comment on the DISPLAY command that is issued will indicate “ISSUED TO LOG REALLOCATE ERROR(S)/EXCEPTION(S)”.

z/OS 1.12 – REALLOCATE ...



- Software Dependencies
 - Available with z/OS V1.12
 - Coexistence apar OA29236 is required on V1R10 and V1R11

57

SHARE
in Anaheim
2011

z/OS 1.12 – Support for CFLEVEL 17



- Large CF Structures
 - Increased CF structure size supported by z/OS to 1TB
 - Usability enhancements for structure size specifications
 - CFRM policy sizes
 - Display output
- More CF Structures can be defined
 - New z/OS limit is 2048 (CF limit is 2047)
- More Structure Connectors (CF limit is 255)
 - Lock structure – new limit is 247
 - Serialized list – new limit is 127
 - Unserialized list – new limit is 255

SHARE
in Anaheim
2011

58

z/OS V1.12 on z/Enterprise 196 servers with Coupling Facility Control Code (CFCC) Level 17 supports up to 2047 structures per Coupling Facility (CF) image, up from the prior limit of 1023. This allows you to define a larger number of data sharing groups, which can help when a large number of structures must be defined, such as to support SAP configurations or to enable large Parallel Sysplex configurations to be merged. This function requires the PTF for APAR OA32807; PTFs are also available for z/OS v1.10 and z/OS V1.11. CFRM supports up to 2048 structures, whereas CFLEVEL 17 supports at most 2047. So in actual use, at most 2047 structures could be allocated at one time in any one CF.

z/OS V1.12 on z/Enterprise 196 servers with CFCC Level 17 also supports more connectors to list and lock structures. XES and CFCC already support 255 connectors to cache structures. With this new support XES also supports up to 247 connectors to a lock structure, 127 connectors to a serialized list structure, and 255 connectors to an unserialized list structure. This support requires the PTF for APAR OA32807; PTFs are also available for z/OS V1.10 and z/OS V1.11. CF Level 17 supports 255 connectors to all types of structures. The nature of the z/OS exploitation of those structures requires that the software impose smaller limits than what the CF supports. For lock structures, 8 bits of the 32 byte lock table entry is consumed for “global ownership”, leaving only 247 bits to represent shared interest. For a serialized list, one bit of the one byte lock table entry is used for lock management, so $x'7F' = 127$ is the largest number of connectors that can be represented in the remaining bits of the byte.

z/OS V1.12 supports larger Coupling Facility (CF) structures. The maximum size you can specify for a CF structure is increased from slightly below 100 GB (99,999,999 KB) to 1 TB. The CFRM policy utility (IXCMIAPU) is updated to allow you to specify structure sizes in units of KB, MB, GB, and TB. These changes improve both Parallel Sysplex CF structure scalability and ease of use.

z/OS 1.12 – Support for CFLEVEL 17 ...



- A new version of the CFRM CDS is needed to define more than 1024 structures in a CFRM policy
- May need to roll updated software around the sysplex for any exploiter that wants to request more than 32 connectors to list and lock structures
 - Not aware of any at this point (so really just positioning for future growth)

SHARE
in Anaheim
2011

59

The Couple Data Set (CDS) containing Coupling Facility Resource Manager (CFRM) policies for the sysplex needs to be reversioned in order to allow CFRM policies that define more than 1024 coupling facility (CF) structures.

Only systems running z/OS V1R10 (or later) with the functionality of APAR OA32807 installed can use a CFRM CDS that is formatted to support more than 1024 structure definitions. Once such a CDS is brought into use, down level systems that need to exploit CFRM will not be able to join the sysplex. Since a sysplex-wide outage is required to fall back to a CFRM CDS that does not support more than 1024 structures, it is advisable to delay using this support until you are satisfied that no system in the sysplex will need to run (fall back to) down level code. Versioning the CFRM CDS to support new functions and protocols is the same technique used in the past for message-based protocols, system-managed rebuild, and again for system-managed duplexing. The new version allows CFRM to ensure that all systems can deal with the greater number of structure definitions.

Use the Couple Data Set (CDS) format utility IXCL1DSU to format a CFRM CDS that supports more than 1024 structures by coding the NUMBER keyword with an appropriate value: (see “Setting Up A Sysplex” for complete details):

ITEM NAME(STR) NUMBER(nnnn)

Finally bring the new CFRM CDS into use. For an existing sysplex that is already using CFRM (most likely case), use the SETXCF COUPLE,TYPE=CFRM,ACOUPLE command to define the new CFRM CDS as an alternate, then use the SETXCF COUPLE,TYPE=CFRM,PSWITCH command to make it the primary CFRM CDS. Don't forget to fix the single point of failure by issuing another SETXCF COUPLE,TYPE=CFRM,ACOUPLE command to define another CFRM CDS that supports at least as many structure definitions as an alternate for redundancy. For an existing sysplex that is not using CFRM (not likely), use the SETXCF COUPLE,TYPE=CFRM,PCOUPLE command to define an appropriate CFRM CDS as the primary. If IPLing the first system into the sysplex, define the CFRM CDS (primary and alternate) in the COUPLExx parmlib member.

The Administrative Utility (IXCMIAPU) can then be used to define CFRM policies with as many structure definitions as the CDS supports.

Enabling support for more than 32 connectors to lock and list structures requires software upgrades. The new version of the CFRM CDS is not needed for this aspect of z/OS 1.12 support. All systems in the sysplex that need to connect to a structure that can have more than 32 connectors will need to have z/OS V1R10 (or later) with the PTFs for APAR OA32807 installed. In addition, the exploiters are required to code a new MAXCONN keyword on the invocation of the IXLCONN macro that is used to connect to the structure.

z/OS 1.12 – Support for CFLEVEL 17 ...



- z/OS requests non-disruptive CF dumps as appropriate
- Coherent Parallel-Sysplex Data Collection Protocol
 - Exploited for duplexed requests
 - Triggering event will result in non-disruptive dump from both CFs, dumps from all connected z/OS images, and capture of relevant link diagnostics within a short period
 - Prerequisites:
 - Installation must ENABLE the XCF function DUPLEXCFDIAG
 - z/OS 1.12
 - z/OS 1.10 or 1.11 with OA31392 (IOS) and OA31387 (XES)
 - Note that full functionality requires that:
 - z/OS image initiating the CF request reside on a z196
 - CF that “spreads the word” reside on a z196



60

The DUPLEXCFDIAG optional function is defined to allow an installation to control the use of the “Coherent Parallel-Sysplex Data Collection Protocol”, aka the “link diagnostics” protocol. By default, the protocol is disabled, and it is expected that an installation would only enable it if it experiences duplexing issues. If the DUPLEXCFDIAG function is disabled, z/OS will not request use of the protocol when issuing duplexed CF requests. DUPLEXCFDIAG is enabled and disabled using the same parmlib (COUPLExx) and command (SETXCF FUNCTIONS) interfaces as all other optional XES / XCF functions.

Even if the function is ENABLED, the protocol is requested only if both CFs are at or above CFLEVEL 17, since there’s no point collecting link data if we aren’t going to get the whole picture. The protocol relies on the CF to capture non-disruptive dumps and to propagate the request for data to the peer CF and connected z/OS images.

z/OS must set parameters in the request that is sent to the CF to indicate that this form of data collection is desired. Similarly, if a CF detects a triggering event, it must set parameters in the notifications that are sent to the peer CF and the connected z/OS images. These parameters are interpreted by the links. Only the z196 side of the links has the ability to process these parameters. Thus, full functionality of the coherent data collection would only be available if the z/OS image initiating the request resides on a z196, and if the CF that detects the triggering event and propagates the request for data collection also resides on a z196. With the appropriate releases and/or PTFs installed, the z/OS images that do not reside on a z196 can generate the appropriate dumps when asked, but they would not be able to initiate coherent data collection.

z/OS 1.12 Health Checks



- **XCF_CF_PROCESSORS**
 - Ensure CF CPU's configured for optimal performance
- **XCF_CF_MEMORY_UTILIZATION**
 - Ensure CF storage is below threshold value
- **XCF_CF_STR_POLICYSIZE**
 - Ensure structure SIZE and INITSIZE values are reasonable

61

SHARE
in Anaheim
2011

New health checks for the Parallel Sysplex components, XCF and XES, are included in z/OS 1.12. These checks can help you correct and prevent common sysplex management problems.

XCF_CF_PROCESSORS

Raises an exception if a CF is not configured with all CPs dedicated. The IBM supplied check raises an exception if any CF in use by the sysplex is configured with one or more shared CP's. To obtain maximum CF performance and throughput, a coupling facility should be configured completely with dedicated processors instead of shared processors. The installation can specify a check parameter to exclude designated CF's from being considered by the check. For example, one might elect to exclude a CF that is using shared CP's because it is part of a test configuration.

XCF_CF_MEMORY_UTILIZATION

Raises an exception if the CF storage usage exceeds the indicated threshold. The IBM supplied check raises an exception if the storage utilization exceeds 60%. The percentage of memory utilization in a coupling facility should not approach an amount so high as to prevent the allocation of new structures or prevent the expansion of existing structures. The installation can specify a check parameter to designate some other threshold.

XCF_CF_STR_POLICYSIZE

Raises an exception if the CFRM policy definition for a structure has size specifications (SIZE and INITSIZE) that can either (a) cause CF storage to be wasted, or (b) make the structure unusable, or (c) prevent the structure from being allocated. Specifying different INITSIZE and SIZE values provides flexibility to dynamically expand the size of a structure for workload changes, but too large a difference between INITSIZE and SIZE may waste coupling facility space or prevent structure allocation.

z/OS 1.12 Health Checks ...



- XCF_CDS_MAXSYSTEM
 - Ensure function CDS supports at least as many systems as the sysplex CDS
- XCF_CFRM_MSGBASED
 - Ensure CFRM is using desired protocols
- XCF_SFM_CFSTRHANGTIME
 - Ensure SFM policy using desired CFSTRHANGTIME specification

Initially complained if more than 300 (5 minutes). APAR OA34439 changed it to 900 (15 minutes) to allow more time for operator intervention and more time for all rebuilds to complete after losing connectivity to a CF

SHARE
in Anaheim
2011

62

XCF_CDS_MAXSYSTEM

Raises an exception when a function couple data set (CDS) is formatted with a MAXSYSTEM value that is less than the MAXSYSTEM value associated with the primary sysplex CDS. A “function CDS” is any CDS other than the sysplex CDS. For example, a function CDS might contain CFRM policies or SFM policies. If a function CDS does not support at least as many systems as the sysplex CDS, a new system might not be fully functional when it joins the sysplex (if it can join at all).

XCF_CFRM_MSGBASED

Raises an exception if CFRM is not configured to exploit the desired structure event management protocols. The IBM supplied check raises an exception if CFRM is not configured to exploit “message based” protocols. The CFRM “message based” protocols were introduced in z/OS V1R8. Use of this protocol reduces contention on the CFRM policy CDS, which can significantly reduce the recovery time for events that trigger CFRM activity against lots of structures (such as loss of a CF or a system). The installation can specify a check parameter to indicate which CFRM protocol is desired, “message based” or “policy based”.

XCF_SFM_CFSTRHANGTIME

Raises an exception if the active SFM policy is not consistent with the indicated CFSTRHANGTIME specification designated by the check. The IBM supplied check will raise an exception if the SFM policy specifies (or defaults to) CFSTRHANGTIME(NO), or if the SFM policy specifies a CFSTRHANGTIME value greater than 900 seconds (was 300 seconds prior to APAR OA34439). The installation can specify a check parameter to indicate the CFSTRHANGTIME value that the installation wants to use. The check will then raise an exception if the SFM policy is not in accordance with the check parameter.

z/OS 1.12 Auto-Reply



- Fast, accurate, knowledgeable responses can be critical
- Delays in responding to WTOR's can impact the sysplex
- Parmlib member defines a reply value and a time delay for a WTOR. The system issues the reply if the WTOR has been outstanding longer than the delay
- Very simple automation
- **Can be used during NIP !**



63

Many installations no longer have operators closely monitoring the system waiting to immediately reply to a WTOR. Operators typically do not have authority, experience, or system understanding to make their own decision on what to reply for uncommon WTORs. Customers have told us that it is reasonable to assume a WTOR may take at least 30-45 minutes to be answered. So it is no longer feasible to expect fast, accurate, knowledgeable answers from system operators. Some WTORs are so infrequent that they are not automated and operators may never have seen them before. Reply delays can affect all systems in sysplex.

In z/OS V1.12, a new Timed Auto Reply function enables the system to respond automatically to write to operator with reply (WTOR) messages. This new function is designed to help provide a timely response to WTORs and help prevent delayed responses from causing system problems.

The Timed Auto Reply Function allows you to specify message IDs, timeout values, and default responses in an auto-reply policy, and to be able to change, activate, and deactivate autoreply with operator commands. Also, when enabled, it starts very early in the IPL process, before conventional message-based automation is available, and continues unless deactivated. You can also replace or modify an IBM-supplied auto-reply policy in a new AUTOR00 parmlib member. This new function is expected to help provide a timely response to WTORs and help prevent delayed responses from causing system problems.

z/OS 1.12 Auto-Reply



- For example:

```
IXC289D REPLY U TO USE THE DATA SETS LAST USED FOR  
typename OR C TO USE THE COUPLE DATA SETS SPECIFIED  
IN COUPLExx
```

- The message occurs when the couple data sets specified in the COUPLExx parmlib member do not match the ones in use by the sysplex (as might happen when the couple data sets are changed dynamically via SETXCF commands to add a new alternate or switch to a new primary)
- Most likely always reply "U"



64

From a sysplex perspective, Auto-Reply will likely prove useful for messages to which the operator must reply while the system is IPLing. May prove quite useful for disaster recovery testing.

z/OS 1.12 – Runtime Diagnostics



- Allows installation to quickly analyze a system experiencing “sick but not dead” symptoms
- Looks for evidence of “soft failures”
- Reduces the skill level needed when examining z/OS for “unknown” problems where the system seems “sick”
- Provides timely, comprehensive analysis at a critical time period with suggestions on how to proceed

- Runs as a started task

SHARE
in Anaheim
2011

65

In z/OS V1.12, a new capability, z/OS Runtime Diagnostics, is designed to help when the need for quick decision-making is required. With Runtime Diagnostics, your z/OS system is designed to analyze key system indicators of a running system. The goal is to help you identify the root of problems that cause system degradation on systems that are still responsive to operator commands. Runtime Diagnostics is anticipated to run quickly to return results fast enough to aid you in making decisions about alternative corrective actions and facilitate high levels of system and application availability.

Run Time Diagnostics will identify critical messages, search for serialization contention, find address spaces consuming a high amount of processor time, and analyze for patterns common to looping address spaces.

z/OS 1.12 – Runtime Diagnostics ...



- Does what you might do manually today:
 - Review critical messages in the log
 - Analyze contention
 - Examine address spaces with high CPU usage
 - Look for an address space that might be in a loop
 - Evaluate local lock conditions
- Additional analysis based on what it finds
 - For example, if XES reports connector as unresponsive, RTD will investigate the appropriate address space

66

SHARE
in Anaheim
2011

Critical Message Analysis: Reads through the last hour of OPERLOG looking for critical messages. If any are found, lists the critical message as an error event. For a subset of critical messages, performs additional analysis based on the message identifier and the content of the message.

Contention Analysis: Provides a point in time check of ENQ contention equivalent to issuing the D GRS,AN,WAITER command. Compares the list of job names that are waiters with the list of system address spaces that are started at IPL to determine if any system address spaces are waiters. If ENQ contention is found, issues an error event message.

CPU analysis: Provides a point in time check of any address space that is using more than 95% of the capacity of a single CPU, which might indicate the address space is in a loop.

Loop Detection: Looks through all tasks in all address spaces to determine if a task appears to be looping. Examines various system information for indicators of consistent repetitive activity that typically appears when a task is in a loop. When both a HIGHCPU event and a LOOP event list the job name, there is a high probability that a task in the job is in a loop.

Local Lock Contention: Provides a point in time check of local lock suspension for any address space. Calculates the amount of time an address space is suspended waiting for the local lock. If an address is suspended more than 50% of the time waiting for a local lock, issues an event message.

z/OS 1.12 – Runtime Diagnostics ...



For more information:

- z/OS V1R12 Problem Management (G325-2564)

67

SHARE
in Anaheim
2011

Runtime Diagnostics may prove quite useful when XCF surfaces various hang conditions or situations for which sympathy sickness might arise. That is, those situations that are to be addressed automatically via SFM parameters such as MEMSTALLTIME and CFSTRHANGTIME.

XCF Programming Interfaces



- IXCMSGOX
 - 64 bit storage for sending messages
 - Duplicate message toleration
 - Message attributes: Recovery, Critical
- IXCMSGIX
 - 64 bit storage for receiving messages
- IXCJOIN
 - Recovery Manager
 - Critical Member
 - Termination level

SHARE
in Anaheim
2011

68

Two new services based on existing XCF signaling services are introduced to support the use of 64-bit addressable virtual storage message buffers and associated input and output parameters. The two new services, IXCMSGOX and IXCMSGIX, are the 64-bit counterparts of the existing IXCMSGO and IXCMSGI services, which are used by XCF group members to send and receive messages. These new services make it easier for exploiters to achieve virtual storage constraint relief by removing the need to copy message buffers and associated storage structures from 64-bit addressable virtual storage to 31-bit storage and back.

IXCMSGOX allows the sender to indicate that the target(s) can tolerate receiving duplicate copies of the message, which allows XCF to resend the message sooner if there should be a signal path failure while the message is being sent. Other new message attributes include “recovery” which is used to identify signals that are involved in recovery processes, and “critical” which indicates that the message is critical to the exploiter.

The IXCJOIN service, which is invoked to become a member of an XCF group, has some new keywords that enable the member to indicate that it is a “recovery manager” that performs a sysplex-wide recovery process, or that it is a “critical member” that is to be terminated by XCF if it appears to be unresponsive. The “termination level” allows the member to indicate the scope at which it wants to be terminated (task, address space, or system) when XCF decides to do so.

Agenda

- Hardware Updates
 - CFCC Level 17
 - CFCC Level 16
 - Parallel Sysplex InfiniBand Links
- z/OS Updates
 - Sysplex Failure Management
 - z/OS V1R12
 - **z/OS V1R13 preview**



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

69

SHARE
in Anaheim
2011

z/OS 1.13 Preview - zFS



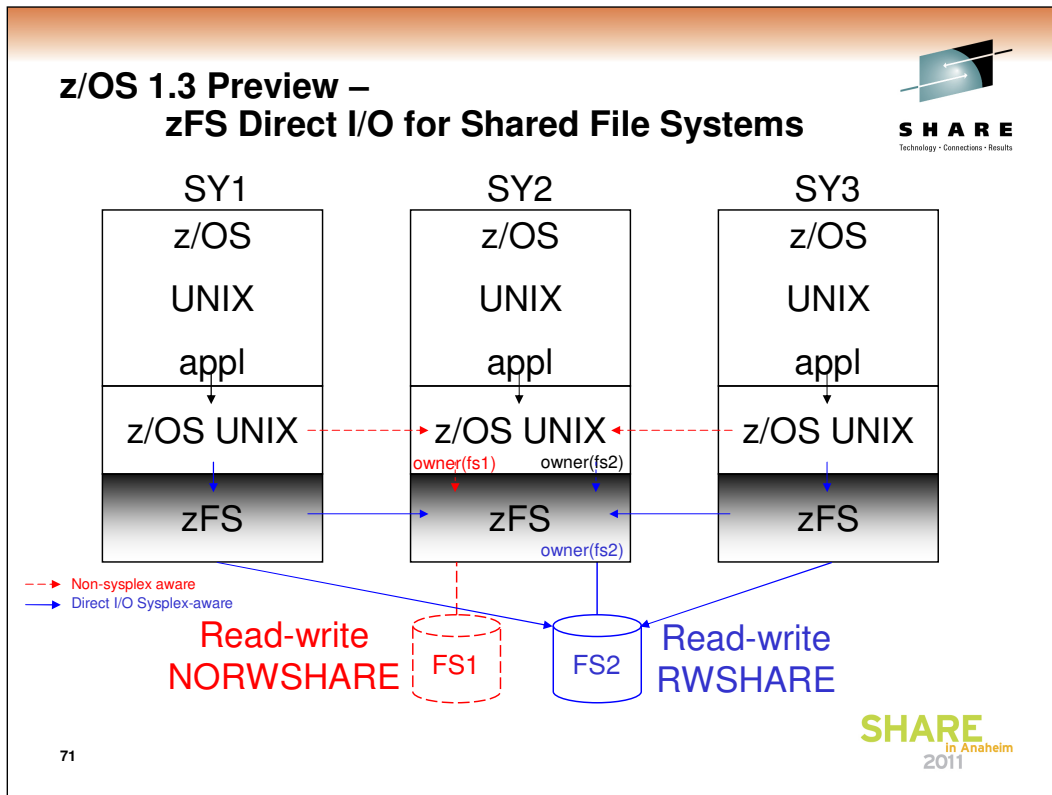
- Full read/write capability from anywhere in the sysplex for shared file systems
 - Better performance for systems that are not zFS owner
 - Reduced overhead on the owner system
- Expected to improve performance of applications that use zFS services
 - z/OS UNIX System Services
 - WebSphere® Application Server

70

SHARE
in Anaheim
2011

zFS processing has been redesigned to allow all members of a Parallel Sysplex to perform zFS file system read and write I/O operations for shared file systems. This is expected to yield substantial performance gains for systems that would not have been zFS owning systems in the prior design, without performance impacts to systems that would have been zFS owning systems.

Faster z/OS UNIX workloads in a Parallel Sysplex: for z/OS UNIX System Services, IBM plans to introduce fully shared zFS file systems across systems in a Parallel Sysplex with direct I/O and zFS internal restart. Applications that use zFS, such as z/OS UNIX System Services and WebSphere Application Server for z/OS, are expected to benefit.



In zFS R13, zFS supports direct I/O for sysplex-aware read-write file systems. This means that when all systems are running z/OS V1R13, zFS can directly read and write user data to a sysplex-aware (RWSHARE) read-write file system (for example, FS2). This generally improves the performance of client system access to the files (from SY1 and SY3) since the data does not need to pass through SY2 with XCF communications. Only metadata updates are sent to the zFS owning system.

z/OS 1.13 Preview – SDSF



- SDSF provides sysplex view of panels:
 - CK (health checks)
 - PS (processes)
 - ENC (enclaves)
 - RM (JES2 resources)
- Data gathered on each system using the SDSF server
- Consolidated on client for display so user can see data from all systems

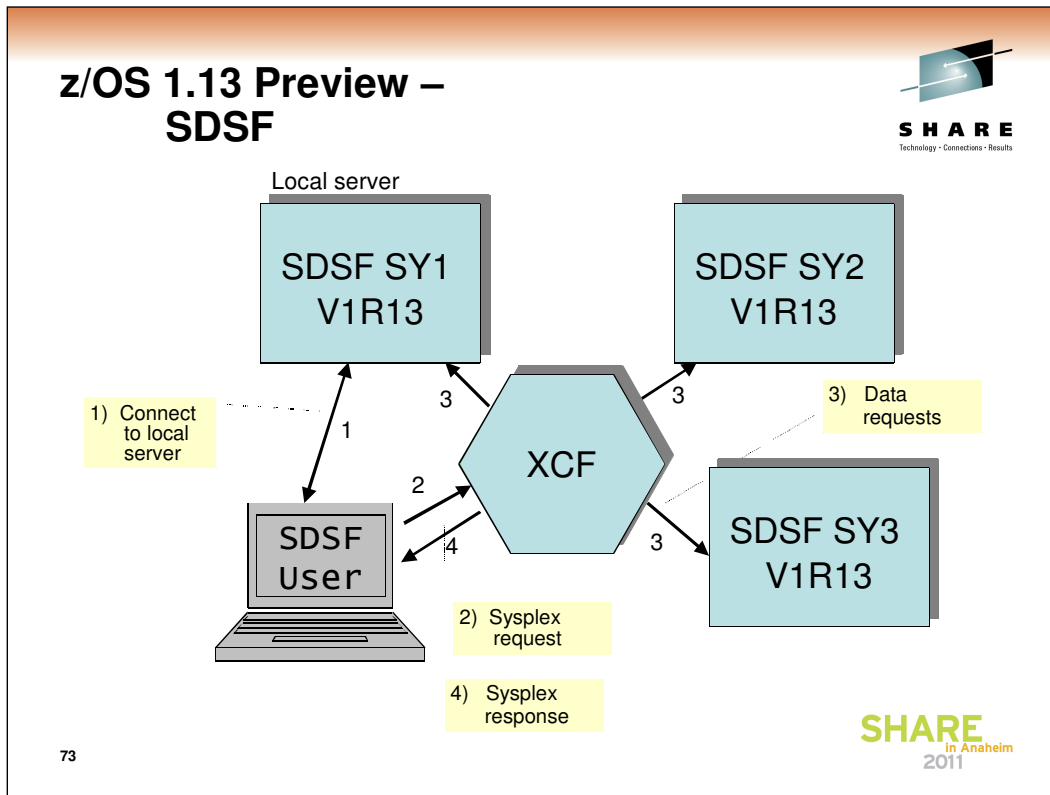
- In z/OS 1.13, SDSF can use XCF communication
- Simplifies configuration and eliminates dependency on MQSeries

72

SHARE
in Anaheim
2011

SDSF is planned to add new support and to remove the requirement for IBM WebSphere® MQ for z/OS (5655-L82) in JES2 environments once all systems in a MAS are running z/OS V1.13 JES2. In this release, SDSF is designed to implement for JES3 all applicable functions that are supported for JES2. For JES2, new planned support includes JES network server and network connections displays. Once all systems in a JES3 complex are using z/OS V1.13 JES3, the new planned support includes displays for initiators, output, held information, job 0, punches, readers, JES network server, and JES Network Connections. The corresponding SDSF Java classes are planned to be updated to support the new displays and actions. These changes are intended to provide systems management improvements.

If want a sysplex wide view in a mixed sysplex, need to continue to use MQSeries for all systems. If willing to limit “sysplex-wide” to just those systems running z/OS V1R13, one need not use MQSeries. Otherwise not until all systems are running with z/OS V1R13 can the requirement for MQSeries be eliminated.



This diagram illustrates the interaction of SDSF with XCF. (1) The SDSF user (client) connects to the SDSF server on the local system. When the user asks SDSF to display a panel with sysplex data, SDSF invokes (2) the IXSEND service to send a request to each of the SDSF servers in the sysplex. The request is delivered only to systems running z/OS V1R13. The SDSF servers gather the relevant data and (3) invoke IXSEND to send a reply back to the originating system. XCF gathers those responses and returns the data to SDSF when (4) the IXCRECV service is invoked. SDSF then presents the results to the SDSF user.

Conceptually, the processing is similar to the behavior on prior releases, except that XCF is now used as the communication vehicle instead of MQSeries.

z/OS V1R13 Preview - XCF Client/Server Interfaces



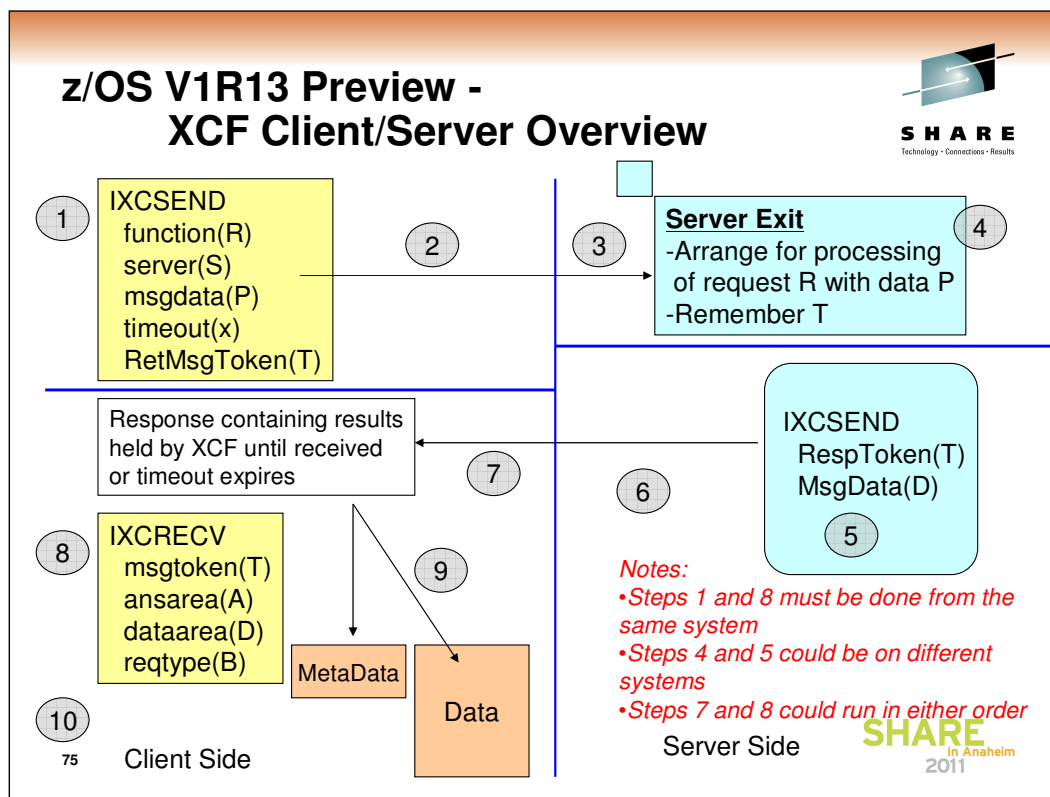
- IXCSRVR – define server(s) to receive requests
- IXCSEND – send a request to a server
- IXCRECV – receive server's response

- Exploiters do not need to join an XCF group
- Requests delivered to server task (not SRB)
- Under the covers, traditional XCF messaging is used

74

SHARE
in Anaheim
2011

XCF is planned to provide a simplified set of interfaces for passing messages within a Parallel Sysplex. New services will be designed to allow a server to be established to process messages and for messages to be sent across the sysplex without first joining an XCF group. This is intended to make it easier to exploit XCF services for applications that do not require the member management and monitoring provided by the XCF group services interfaces.



1. Client issues IXCSEND macro to send request R to server S, providing parameters P. The macro returns a token T that the client will later use to retrieve the results provided by the server. The parameters P can be at most 100 MB. The target server is identified by its name and the name of the system on which it resides. The same server name can be defined on any system in the sysplex.
2. XCF builds control blocks to manage the request and invokes IXCMMSGOX to send the request and its parameters to the system(s) on which the server (allegedly) resides.
3. XCF on the target system intercepts the signal. If the target server does not exist, the message is discarded and acknowledged with a "no receiver" response. If the server does exist, XCF invokes IXCMMSGC to save the message and queues a work item for the server. As needed, a server is selected and resumed. The server exit stub makes suitable preparations for processing the work item, including doing IXCMMSGIX to extract the client parameters from the saved message. XCF calls the server exit routine to present the request to the server.
4. The server exit routine inspects R to determine what type of request is to be processed. The server exit can process the request directly or it can arrange for it to be processed asynchronously. It needs to retain the token T that represents the client request for later use when sending the results of the request back to the originating client. Note that the token given to the client and the token given to the server represent the same logical request, but the tokens themselves will not have the same content.
5. After the server exit or its agent processes the request, the IXCSEND macro is invoked to send the results D back to the originating client. The token T identifies the client request to which the results belong. The results D can be at most 100 MB.
6. XCF decodes the token T to determine where the results should be sent. XCF invokes IXCMMSGOX to send the results to the originating system. If the message exceeds 60KB, XCF may suspend the responding thread until the IXCMMSGOX service finishes sending the message.
7. XCF on the client system intercepts the server response. XCF locates the control blocks used to manage the original client request. If not found, the client request timed out and the server response is discarded. Otherwise XCF binds the response message to the client request and holds it until the results are gathered by the client, or the client request times out. If the client is waiting for the results, XCF notifies (resumes) the client.
8. The client issues an IXCRECV request to gather the results of the request identified by token T (which was returned in step 1 when the request was initiated). The client provides an answer area A which will be filled with metadata that describes the response. The data area D will be filled with the response data sent by the server. It could be that the response has not yet been arrived when the client attempts to gather it. The reqtype B indicates how the client wants to deal with this. The IXCRECV service can suspend (reqtype=blocking) until the response arrives (or the request times out or is cancelled), or the service routine can return immediately if the response is not yet available (reqtype=nonblocking).
9. To process the IXCRECV request, XCF locates the control blocks being used to manage the request identified by token T. If not found, the request timed out (or was cancelled) and no results are available. If the request is found, XCF determines whether the response has arrived or not. If not, XCF either suspends client thread or returns per the reqtype B. If suspended, XCF will resume the thread when the results arrive, or when the request times out, or when the request is cancelled, whichever comes first. Assuming the results have arrived, XCF determines whether the answer area and data area provided by the caller are large enough. If not, XCF returns to the client indicating what size is needed to receive the results. If the output areas are large enough, XCF fills them with the appropriate data, discards the control blocks used to manage the request, and returns to the client.
10. Client inspects Metadata and/or Data, and processes the results of the request.

z/OS 1.13 Preview – Runtime Diagnostics



- Add monitoring of latch contention
 - GRS
 - z/OS Unix System Services file system

76

SHARE
in Anaheim
2011

Runtime Diagnostics, introduced with z/OS V1.12, enables your z/OS system to quickly and automatically scan system components, analyze metrics, and report on components (such as address spaces or tasks) it suspects as being the cause of potentially abnormal system behavior. Runtime Diagnostics is designed to operate on a still-running z/OS system, giving your system programmers accurate information to work from in real time. z/OS V1.13 Runtime Diagnostics is planned to add additional monitoring of GRS latch and z/OS UNIX System Services file system latch contention.


Agenda

- Hardware Updates
 - CFCC Level 17
 - CFCC Level 16
 - Parallel Sysplex InfiniBand Links
- z/OS Updates
 - Sysplex Failure Management
 - z/OS V1R12
 - z/OS V1R13 preview
- **Summary**




SHARE
in Anaheim
2011

77



Highlights

- **CFLEVEL 17 for z196**
 - More structures and nondisruptive dumping
- **Infiniband links for**
 - Bandwidth
 - Fewer physical links
 - High performance links at 150 meters
- **SFM with BCPii for better availability**
- **z/OS 1.12**
 - Sympathy sickness resolution for better availability
 - REALLOCATE test and report for CF structure management



78

Coupling Facility Control Code (CFCC) CFLEVEL 17 for the z196 supports up to 2047 structures. Nondisruptive dumping enables capture of diagnostic data without disruption.

Infiniband links can be used for increased bandwidth, can help simplify sysplex configurations by reducing the number of links needed to connect CECs, and can provide high performance links at greater distances than do current links.

SFM with BCPii is a critical technology for improving sysplex availability as it allows XCF to know with certainty that an apparently unresponsive system is in fact not operational. This knowledge enables XCF to remove systems from the sysplex without operator intervention.

z/OS 1.12 extends Sysplex Failure Manager (SFM) support to provide automatic resolution of additional sympathy sickness conditions which would otherwise impact the sysplex. It also provides some enhancements related to the Coupling Facility Resource Manager (CFRM) REALLOCATE function that customers have requested, namely the ability to determine what the function might do and what it most recently did.

Sysplex-related Redbooks of potential interest



- System z Parallel Sysplex Best Practices, SG24-7817
- Considerations for Multi-Site Sysplex Data Sharing, SG24-7263
- Exploiting the IBM Health Checker for z/OS Infrastructure, REDP-4590
- Available at www.redbooks.ibm.com

79

SHARE
in Anaheim
2011

Redbooks often provide clear, concise, comprehensive material.

Other Sources of Information




- *MVS Setting Up a Sysplex (SA22-7625)*
- *MVS Initialization and Tuning (SA22-7591)*
- *MVS Systems Commands (SA22-7627)*
- *MVS Diagnosis: Tools and Service Aids (GA22-7589)*
- *z/OS V1R12.0 Planning for Installation (GA22-7504)*
- *z/OS MVS Programming: Callable Services for High Level Languages (SA22-7613)*
 - Documents BCPii Setup and Installation and BCPii APIs

80


SHARE
in Anaheim
2011

These publications are available at
<http://www.ibm.com/systems/z/os/zos/bkserv/>

Parallel Sysplex Web Site



- www.ibm.com/systems/z/pso



Parallel Sysplex

IBM SERVER TIME PROTOCOL (STP)
Time Synchronization for the Next Generation
→ Learn more

About | **STP** | **Supporting products** | **Learn more** | **Services**

Overview | Detailed info | Benefits | What's new |
CF structures | CF levels | IFB

81

SHARE
in Anaheim
2011

The parallel sysplex web site is a good starting point for sysplex information. Under the “Learn More” tab one can find white papers and other documentation that is helpful with respect sysplex configuration and management.