Best Practices for IBM z/OS in a Parallel Sysplex
IBM Z WSC Health Check Guidelines

This document can be found on the web at
https://www.ibm.com/support/pages/node/6565411

Version 1: 5 April 2022

Washington Systems Center

Jack Billings
Z Software Technical Specialist

Trademarks

The following terms are registered trademarks of International Business Machines Corporation in the United States and/or other countries: AIX, AS/400, DB2, IBM, Micro Channel, MQSeries, Netfinity, NUMA-Q, OS/390, OS/400, Parallel Sysplex, PartnerLink, POWERparallel, RS/6000, S/390, Scalable POWERparallel Systems, Sequent, SP2, System/390, ThinkPad, WebSphere.

The following terms are trademarks of International Business Machines Corporation in the United States and/or other countries: DB2 Universal Database, DEEP BLUE, e-business (logo), ~, GigaProcessor, HACMP/6000, Intelligent Miner, iSeries, Network Station, NUMACenter, POWER2 Architecture, PowerPC 604,pSeries, Sequent (logo), SmoothStart, SP, xSeries, zSeries. A full list of U.S. trademarks owned by IBM may be found at http://iplswww.nas.ibm.com/wpts/trademarks/trademar.htm. NetView, Tivoli and TME are registered trademarks and TME Enterprise is a trademark of Tivoli Systems, Inc. in the United States and/or other countries.
Oracle, MetaLink are registered trademarks of Oracle Corporation in the USA and/or other countries.
Microsoft, Windows, Windows NT and the Windows logo are registered trademarks of Microsoft Corporation in the United States and/or other countries.
UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.
LINUX is a registered trademark of Linus Torvalds.
Intel and Pentium are registered trademarks and MMX, Pentium II Xeon and Pentium III Xeon are trademarks of Intel Corporation in the United States and/or other countries.
Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States and/or other countries.
Other company, product and service names may be trademarks or service marks of others.

Table of Contents

1.0 Abstract 1
1.1 Contact and Support 2
1.2 Feedback 2
2.0 Parallel Sysplex and z/OS 3
2.1 Separate Production and Non-Production Workloads 3
2.2 Software Maintenance and Cloning 4
2.3 Automation 5
2.4 HyperSwap 8
2.5 Coupling Facility 9
2.6 Sysplex Data Sets 13
2.7 XCF Signaling Paths 14
2.8 System Logger (LOGR) 15
2.9 Naming Conventions 16
3.0 CICS 18
3.1 High Availability/Topology 18
3.2 CICS Data Sharing 19
3.3 CICSPlex System Manager (CP/SM) 20
3.4 Monitoring 21
4.0 Communications Server 22
4.1 Sysplex Distributor 22
4.2 SNA 23
4.3 TCP/IP 23
4.4 Network 26
5.0 Db2 28
5.1 Db2 Data Sharing 28
5.2 Db2 Recovery Logs 30
5.3 Backup and Reorg Processes 32
6.0 IMS 35
6.1 IMS Connectivity 35
6.2 IMS Data Sharing 36
6.3 IMS Shared Queues 37
7.0 MQ 42
7.1 MQ Queue Sharing Group 42
7.2 Coupling Facility Resources for MQ 44
8.0 WLM 46
8.1 General 46
8.2 Service Classes 46
9.0 Other Sources of Information 51
9.1 Contacts 51
9.2 Additional Guides and Documentation 51

1.0 Abstract

This guide is intended to show z/OS best practices for any system in a Parallel Sysplex. It will provide IBM Z WSC health check guidelines that are applicable for all systems to keep them running properly.

The Primary Focus of the IBM Z WSC Health Check is to Improve Availability.

Availability – attribute of a component, or service, to perform its required function over a stated period of time

- High Availability (HA)
    - The attribute of a system to provide service during defined periods, at acceptable or agreed upon levels and mask unplanned outages from end-users.
        - Employs: fault tolerance; automated failure detection, recovery, bypass; testing, problem and change management.
- Continuous Operations (CO)
    - The attribute of a system to continuously operate and mask planned outages from end-users
        - Employs: non-disruptive hardware and software change, non-disruptive configuration, software coexistence.
- Continuous Availability (CA)
    - The attribute of a system to deliver non-disruptive service to the end-user, 24 hours a day, 7 days a week (No planned or unplanned outages)
- Disaster Recovery (DR)
    - Protecting “critical” business services/functions during wide-scale disruptions, minimizing “out-of-market” durations with limited loss of data.
        - DR cannot be executed without significant impact to the business
        - Recovery times and data loss may be regulated (government and/or business)
        - Data mirroring and automation are required to achieve business goals/objectives

1.1 Contact and Support

Work with Jack Billings (jack.billings@ibm.com) or Jim Thomas (jimbo@ibm.com) as first points of contact if you need help.

1.2 Feedback

All suggestions or errors in this document should be sent to Jack Billings using this email address jack.billings@ibm.com

You can also take the error code and enter it into the IBM Knowledge Center to see what the issue is as well. https://www.ibm.com/support/knowledgecenter

2.0 Parallel Sysplex and z/OS

High availability and continuous operations cannot be achieved without the use of Parallel Sysplex. Continuous or near-continuous application availability can only be achieved by properly designing, implementing, and managing the Parallel Sysplex systems environment. The overall objective of designing a Parallel Sysplex for high availability and continuous operations is to create an environment where the loss of any single component or a planned outage does not affect the availability of the application. This is achieved by designing a fault-tolerant systems environment where:

- Hardware and software failures are masked by redundant design.
- Hardware and software systems are configured symmetrically, thereby enabling workloads to run anywhere in the Parallel Sysplex.
- Workloads and logons are balanced among systems by cloning applications and utilizing high availability features such as Dynamic VIPA and data sharing.

2.1 Separate Production and Non-Production Workloads

Description/Benefit

- Separate production and non-production workloads into separate sysplex environments.
  - Prevents non-production workloads/activities from disrupting critical production business services.
  - Facilitates independent testing of software, automation, and recovery testing processes before introduction to the production environment.

Details

- Create a separate Parallel Sysplex environment and move all non-production workloads to this sysplex.
  - QA/Preproduction test environment configuration should mirror the production environment.
  - At least 2-LPARs, data sharing, CICS & CICSPlex configuration, software levels, automation.
  - Review and establish processes to move application and other software changes between environments.
  - If development requires access to production data to create test data bases to support testing activities, establish processes to pull data from production and FTP to the test environment.
  - Run data extraction and masking tools in production to extract the production data and to protect consumer personal and confidential data and then FTP to the development sysplex.

- Review and update test processes and procedures.
    - IBM Z Business Resilience Stress Test (zBuRST) can be used for load or stress tests at production scale or higher (2x).

2.2 Software Maintenance and Cloning

Description/Benefit

- Set up a “master” System Modification Program Extended (SMP/E) environment that is used to apply and propagate maintenance across the different sysplex environments. Exploit z/OS cloning techniques and share SYSRES, Master Catalogs, and System PARMLIBs and members across all LPARs in the sysplex to ensure symmetry across the sysplex. The benefit is when you apply maintenance to your master SMP/E environment it will only affect your master SMP/E environment not your production and non-production running environments.

Details

- Set up a master SMP/E environment for each software component in the SYSPROG test environment to support software installation and maintenance activities.
    - Each software product should have their own DLIB/TLIB SMP/E zones and libraries.
    - Consider sharing the Global SMPPTS spill data sets for all products to receive maintenance.
    - Follow IBM’s Recommended Service Upgrade (RSU) process to apply maintenance 2-4 times per year.
    - Use SMP/E cloning techniques to create a SYSRES volume that will be copied between sysplex environments.
- Exploit z/OS cloning techniques for sharing the SYSRES volume(s), Master Catalog, SYS1.PARMLIB and PARMLIB members and software across all LPARs within a sysplex environment.
    - Establish naming conventions and change processes.
    - Create a “sysplex” PARMLIB on a non-SYSRES volume and place all shared/modified members in that PARMLIB.
    - Exploit Define Alias with SYMBOLICRELATE for software data sets (e.g. Db2 LOADLIBs, etc.)

Example - SMP/E Maintenance Db2

1. All normal maintenance is applied against SMP/E master environment.
    - Apply Recommended Service Upgrade (RSU) maintenance to master target volume.

2. Copy target data sets to MVS1 using SMP/E cloning techniques.
3. Change system symbolic for DB2 version.
    - Datasets are cataloged using SYMBOLICRELATE.
4. After maintenance has been tested in TESTPLEXA and “Stabilized”.
    1. Copy Maintenance volume to new Production volume.
    2. IPL production systems using new maintenance (rolling IPLs) / or use dynamic update for symbolic and refresh linklst/DB2, etc.).
        - Db2 Members require their own LOADLIB.
        - Use SYMBOLICRELATE to point to the correct Db2 Maintenance.

2.3 Automation

Description/Benefit

- Automate IPL, shutdown, operator, and recovery actions.
- React faster to operational activities and recovery actions by eliminating the need for a human to respond.
- Automation product must have sysplex awareness and the ability to perform automated actions across the sysplex.

Details

- Identify changes to existing automation:
    - Subsystems start/shutdown commands (Db2, IMS, MQ, etc).
    - New subsystem messages.
- Enable AUTOIPL in a non-GDPS environment.
- Automate system IPL and Shutdown processes.
- All cross-system coupling facility (XCF) and sysplex services for data sharing (XES) Health checks have been enabled to run on an ongoing basis, and any resulting warnings/exceptions have been investigated and addressed as needed.
- Automate critical messages on things that indicate failures, abnormal conditions, or potential recovery actions that should be automated to ensure that critical system/sysplex messages which are important for availability reasons are noticed, investigated, and reacted to as swiftly as possible. The IXC Messages page has a list of all system/sysplex messages (make sure to choose the correct version of z/OS). The Mission: Available white paper also has some suggestions for messages to automate.

Automatic Restart Manager (ARM)

- Need to determine automation strategy for in-place and cross-system restart for critical subsystems – either use automation or ARM
    - Automation provides additional functions and must be sysplex compliant.
    - ARM reacts faster to ABENDs.
- Use ARM to restart Automation if it fails.
- Ensure that an active ARM Policy exists for the sysplex.
- For cross-system automatic restarts, ensure that those systems have access to the required databases, files, logs, program libraries, and so on that are required to execute the restarted workload.
- Define restart processes for all restartable elements and enable cross-system restarts in the sysplex.
- Automate recovery actions:
    - Define and activate an SFM policy based on best practices that allows sick (but not dead) systems to be partitioned out of the sysplex.
    - Enable System Status Detection and Partitioning Protocol.
    - Review and consider using z/OS ARM:
        - Establish in-place restart automation to quickly restart failed subsystems on the same LPAR (limit to 3 attempts).
        - For an LPAR failure, establish cross-system restart to start a failed Db2 and IMS member on an available LPAR to release retained locks.
        - Determine if CICS regions have in-doubt units of work and need to be restarted to release locks (cross-system).
    - Automate activation of spare capacity in the event of an LPAR or CEC failure to prevent workload slowdowns/disruptions.

Sysplex Failure Management (SFM) – See SFM section of Mission: Available white paper for more details.

- For systems running discretionary workloads, set up an active SFM to automatically partition sick but not dead systems out of the sysplex when:
    - You get a status update missing condition (Isolation).
    - You get a connectivity failure (CONNFAIL).
- Specify CONNFAIL(YES) to automate recovery from system-to-system connectivity failures.

- The failure detection interval (FDI) is the length of time a system can be in a "status update missing" condition – basically, not updating its sysplex couple data set "heartbeat" information that makes the system appear active to other systems – before other systems in the sysplex will consider a true failure to have been detected for that system. Failure Detection Interval should be set to meet the needs of the enterprise and have a correct relation to excessive spin parameters.

- Use the defaults for EXSPATxx and allow the FDI to default to the calculated SPIN FDI based on those excessive spin defaults.

- Assign an appropriate system weight value to prioritize recovery actions.

    - Specify REBUILDPERCENT(1) for structures in your coupling facility resource management (CFRM) policy to automatically initiate rebuild in case of any loss of connectivity to the structure. Use caution, as improper REBUILDPERCENT and system weight values may cause structure rebuild to be suppressed.

- Specify NOPROMPT – Do not prompt an operator to reply to a WTOR if a system enters a status update missing condition.

- ISOLATETIME=0 – The number of seconds to wait (after the system is in a status update missing condition) to isolate a system using the fencing services through the coupling facility for a system that is not sending XCF signals and has entered status update missing condition.

- SSUMLIMIT(900) - Number of seconds a system can be in a status update missing condition and yet still sending XCF signals before SFM removes the system from the sysplex.

- MEMSTALLTIME(900) – Alleviate signaling sympathy sickness caused by a stalled XCF group. Recommendation: 600 seconds to 900 seconds.

- CFSTRHANGTIME(900) – The time that a coupling facility structure connector can remain unresponsive before the system takes action to relieve a hang in a structure-related process.

System Status Detection Partitioning Protocol (SSDPP)

- Enable SSDPP to partition a failed system out of the sysplex without waiting for the FDI to expire. SSDP utilizes BCPii functions to determine if a system whose status update is missing is actually in a known state where it cannot possibly resume processing without a re-IPL (perhaps in a nonrestartable disabled wait state, or on an image that has been reset or deactivated).

    - Configure the BCPii interface.

    - Format the XCF couple data set (CDS).

        - ITEM NAME(SYSSTATDET) NUMBER(1)

2.4 HyperSwap

Description/Benefit

- Enhance Parallel Sysplex availability by implementing HyperSwap to protect against unplanned or planned storage subsystem outages to provide near continuous availability.
    - Without HyperSwap, if a storage subsystem fails, the Parallel Sysplex will fail because it cannot access the data resulting in a full business outage. Furthermore, there is no remote disk mirroring so a storage subsystem failure could result in a prolonged business outage that last hours or days if data must be recovered from tape.
    - HyperSwap provides the ability to non-disruptively swap to a secondary disk system to keep the business running.

Details

- Implement Metro Mirror HyperSwap
    - Basic z/OS HyperSwap is included with z/OS and uses Copy Services Manager to manage the disk replication and HyperSwap. Only HyperSwap for z/OS is supported and there are no system automation capabilities.
    - GDPS Metro Mirror HyperSwap with cross-platform disaster recovery (xDR) provides full automation, monitoring, and HyperSwap support for z/OS, z/VM, and LINUX LPARs running on IBM Z.
- Requirements:
    - IBM Z NetView
    - IBM Z System Automation
    - System Automation Multi-Platform is required for z/VM and LINUX on Z. A second disk subsystem with metro mirror support.
    - One of:
        - z/OS Basic HyperSwap
        - Copy Services Manager
        - GDPS Metro HyperSwap Manager
        - GDPS Metro

2.5 Coupling Facility

Description/Benefit

- Coupling Facilities (CFs) are required in a Parallel Sysplex environment to support data sharing. This section provides best practices for the CFs themselves, and guidance for the structures that reside in the CFs.

Details

- Use multiple coupling facility resource management (CFRM) policies.
    - Use a different name from the current policy when you need to make a change. The simple reason for this is that it allows you to easily back out the new policy if you discover that it contains an error. If you replace the policy with an updated one with the same name, the only way to back out the change is to update the policy source again to undo your change—a process that takes longer and requires more expertise than simply activating the previous policy again.
    - CFRM policy using all CFs. Used when doing CFCC upgrades or planned outages of servers.
- Establish and document operational and recovery procedures:
    - Document procedures for:
        - Moving structures from one CF to another and restoring them back to their original CF. The REALLOCATE option of the SETXCF START command and POPULATECF command is very useful for this task.
        - Changing CFRM policies and handling “policy change pending” situations.
        - Shutting down and removing a CF.
        - Adding a CF to the sysplex.
    - Plan for structure recovery in the event of a CF failure.
    - When possible, automate operational and recovery procedures to prevent outages due to human or procedural errors.
- Initialize the CFRM data sets to support message-based processing.
    - Enabling MSGBASED processing ensures that systems will use XCF signals to communicate the stages of rebuild processing and recovery, as well as for coordinating other types of structure event processes. To use MSGBASED processing and minimize CFRM-related delays during processing of structure events, including rebuild times, the CFRM CDS must be formatted with:
        - ITEM NAME(MSGBASED) NUMBER(1)
    - SMDUPLEX allows for such structures to more quickly and more transparently recover from a structure failure, CF failure, or loss of CF connectivity which affects

  such structures. This should be enabled, especially if using integrated catalog facilities (ICFs). To use SMDUPLEX, the CFRM CDS must be formatted with:

    - ITEM NAME(SMDUPLEX) NUMBER(1)

  - Enablement for SMREBLD is strongly recommended to broaden the scope of CF structure rebuild processing for planned reconfiguration and REALLOCATE purposes. To use SMREBLD, the CFRM CDS must be formatted with:

    - ITEM NAME(SMREBLD) NUMBER(1)

  - ASYNCDUPLEX (although if just ASYNCDUPLEX then don’t need all the others, but good to specify anyway).

- Configure at least two CFs to provide redundancy in the event one of the CFs fails. This ensures that structures can be rebuilt in the alternate CF and the sysplex can continue to run.

- There are multiple ways how to distribute z/OS system and CF partitions across servers. For large production sysplexes, it is recommended to run stand-alone CF (SACF) servers.

- Need failure isolation – isolate lock structure so if CEC fails then you don’t simultaneously lose a Db2 instance and lock or SCA structures.

  - BEST – 2-SACF servers are preferred to internal CFs.

  - GOOD – Configure one external CF, one internal CF.

    - Failure isolate Db2 Lock1, Shared Communication Area (SCA) by placing them in the SACF.

    - Place lock structure in the SACF.

  - OK – Configure two internal CFs.

    - Duplex lock structures.

      - System managed duplexing adds 4-5x overhead with duplexed lock structures.

      - CF Asynchronous Lock structure duplexing significantly reduces the overhead for Lock structure duplexing.

    - Duplex Db2 SCA structure.

    - Duplex critical structures as needed.

- Refer to the CF Configuration Options White Paper for more details.

- Install the necessary z/OS program temporary fixes (PTFs) to keep the firmware for a CF up to date.

- To avoid a Power-On Reset (POR) for any I/O configuration change, Hardware Configuration Definition (HCD) and firmware has the capability to perform hardware-only dynamic activate for remote stand-alone coupling facilities (SACFs). The activation can be driven by a z/OS system by invoking a hardware activation service on the remote SACF. It is best practice to use the hardware-only dynamic activate for SACF processors. The z/OS HCD Planning document describes this process in more detail.

- Define dedicated Internal Coupling Facilities (ICFs) CPs for production CFs doing application data sharing. The use of shared engines can impact performance and introduces the risk that abnormal activity in the sharing sysplex could impact the production sysplex.

- For production CFs in a resource sharing environment the access rates are much lower (other than global resource serialization (GRS) structure) and one can easily manage with shared CPs without performance impact, especially when using CF Thin Interrupts.

- If you're using System Managed Duplexing the best practice is to define at least two CPs for each production CF for availability.

- Have at least two paths from each operating system image to the CF. Additional paths may be required with heavy workloads.

- CF-to-CF links should be available and being used for duplexing and possibly for Server Time Protocol (STP). Do not need CF-to-CF links if not duplexing and STP configuration doesn't require it.

- Monitor CF Link utilization as workload utilization increases to ensure there are no performance issues.

  - Should Path Busy of Subchannel delays approach or exceed 10% of total requests, increase the number of CF Links and subchannel.

- Change CFs to NONVOLATILE to avoid rebuild issues with structures that are sensitive to CF volatility.

  - The option is set from the CF's hardware management console (HMC) using the CF MODE= command.

- To ensure the availability of mission critical data, be sure to configure structure duplexing (either system-managed or user-managed) to match the recommendation of the exploiting application. Note that system-managed duplexing may have significant performance implications which must be considered prior to implementation for any given CF structure.

- z/OS Management Facility (z/OSMF) version 2.5 introduced the CFRM Administrative Policies. You can use this task to:

  - Edit a CFRM policy by using a graphical user interface (GUI). You can add, delete, and modify the control statements in a policy without having to know or understand JCL. As you work, the editor checks your changes for correct syntax.

  - Tailor an existing CFRM policy for your installation.

  - Create a CFRM policy by using a series of dialogs and prompts. On completion, the policy is ready for use in your sysplex.

There are two IBM tools available to assist with the proper sizing of CF structures.

- First, the CFSizer website can be used to obtain accurate sizings for CF structures which are being created for the first time, or when an application workload or sysplex infrastructure parameter (for example, number of systems in the sysplex) is being changed in a way that may affect proper CF structure sizing. The CF Sizer output can then be used to update the CFRM policy appropriately with new CF structure sizes, and these size changes can be implemented using the REALLOCATE function.

- Second, the Sizer batch utility can be used when upgrading from one CFCC level (CFLEVEL) to the next level.

    - In order for this batch utility to carry out its sizing function, the up level CF must be available for use in your sysplex, but must have not yet been populated with CF structures.

    - When the up level CF is first made available in your sysplex, run the SIZER batch job to obtain new sizings for all of the active structures that are currently still allocated in the downlevel CF. Next, update the CFRM policy appropriately with the new structure sizes determined by the SIZER utility. Finally, make the up level CF available for use and allow structures to rebuild (as appropriate) into the new CF using the REALLOCATE function.

    - Note that the SIZER batch utility is only useful if the structure sizes on the downlevel CF are currently believed to be appropriate and correct for the applications using the structures. If, in fact, the allocated structures on the downlevel CF are inadequate, then the sizes determined by SIZER will also be similarly inadequate on the up level CF. The SIZER batch utility is trying to determine "equivalence" between the structures allocated in a downlevel and up level CF; it is not trying to determine correctness or sufficiency of the structure sizes.

- Recommendations:

    - Do not make your CF structures too small. Initial allocations may fail when the connectors reject the attributes of a too-small structure, or in some cases it may not be possible to allocate an under-sized structure in the CF at all. If a very small structure is allocated, even if it is accepted for use by the connector, it is likely to encounter performance or availability problems. Generally speaking, erroring on the side of "oversizing" CF structures is far less likely to lead to problems than "undersizing" them.

    - Enable AutoAlter processing, ALLOWAUTOALT(YES), for structures that support it. This allows XCF/XES to dynamically expand, contract, and reapportion a coupling facility structure. Use INITSIZE and SIZE parameters to adjust the size of structures.

    - The ratio between the initial allocation size of the structure (INITSIZE) and the maximum possible size of the structure (SIZE) should not exceed 1:2. The greater the discrepancy between the structure's INITSIZE and maximum SIZE, the greater the amount of "fixed overhead" space required to manage the potential future growth in the structure size – which takes away usable space from the structure given its initial allocation size. While it is desirable to allocate CF structures in such a way that they allow room for future expansion via structure alter, this should not be over-done.

2.6 Sysplex Data Sets

Description/Benefit

- The sysplex data sets are classified as follows:
  - Sysplex couple data set - This is the data set that is used by XCF to manage the sysplex.
  - Functional couple data sets - These data sets are associated with a particular function such as CFRM, Workload Manager (WLM), and so on.
- The sysplex couple data set and the CFRM couple data set are critical to the sysplex. Any system that loses access to these data sets is placed into a non-restartable WAIT state.

Details

- Define primary, alternate, and spare couple data sets (CDS) with the same attributes.
- Allocate the primary, alternate, and spare CDSes on high availability storage.
- Allocate data sets on storage subsystems with caching and fast write enabled.
- The primary, alternate, and spare CDSes should be placed on different storage subsystems. If not possible, ensure they are on different Logical Control Units (LCUs). Attempt to distribute the CDSes across as many physical storage subsystems as possible or reasonable. If you have a multi-site sysplex, give special consideration to the placement of the CDSes across the two sites. It is critical that the CDSes are placed on volumes that will not have any RESERVEs issued against them. The XCF_CDS_SEPARATION health check validates the placement of CDSes.
- CDSes must be allocated on isolated volumes to avoid I/O delays and reserve activity – DO NOT allocate other data sets on the CDS volumes.
- CDS volumes should not be managed by Storage Management Subsystem (SMS) or HSM.
- Do not allocate the primary sysplex couple data set on the same volume as the primary CFRM.
- Allocate LOGR couple data sets on mirrored DASD volumes.
- Establish redundant I/O paths through different FICON directors, ensuring that there are no single points of failure in any of the primary, alternate, or spare data set configurations.
- Do not over-specify parameters when formatting CDSes. If you specify values that are far larger than necessary, this can create unnecessary overhead.
- You can dynamically increase the values by formatting new couple data sets then switching to them by using the SETXCF command. If you want to decrease values, it requires a sysplex-wide outage.

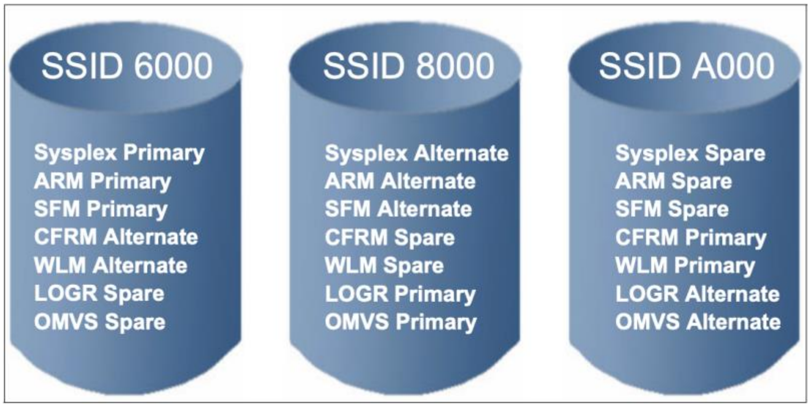
- Enable automation to add the spare couple data set. Having defined a spare CDS, you want to be sure that if XCF ends up running with just a single CDS, the spare is added as the new alternate as quickly as possible.

- Do NOT mirror the CDSes using Metro Mirror. Instead, have the alternate CDSes be on the secondary disk.

- Put automation in place to issue the SETXCF COUPLE command to add the spare couple data set in the event the original primary or alternate CDS becomes unavailable (and notify the system programmer of the recovery event). After activating the spare data set, recover the failed CDS quickly so it can become a spare. Remember to update the COUPLExx member to reflect the new data set names for the primary and alternate couple data sets. If using GDPS, let GDPS handle this automation.

- Set the CLEANUP statement in all COUPLxx members to the default 15 seconds.

  Example in a single site configuration:



2.7 XCF Signaling Paths

Description/Benefit

- Every system in the sysplex must be able to communicate with every other system in the sysplex. A loss of intersystem signaling connectivity may have an impact on the entire sysplex. Need to ensure that there is redundancy built into the XCF signaling configuration.

- One of the other significant functions of XCF is to assist in communication between programs in a sysplex. The communicating programs might be on the same system, or they might be on different systems. One of the advantages of using XCF is that the exploiters do not need to be aware of the location of their peer. They only need to know the XCF member name of the peer.

Details

- Each CF should have at least two hardware paths to each z/OS, using CF links.
  - In the early days, channel-to-channels (CTCs) were faster than CF links, but today CF links are faster and using CF structures makes the environment less complicated.

- Ensure there are no single points of failure related to the hardware paths:
    - To ensure the sysplex can survive/recover in the event of an ICF failure, ensure there are multiple XCF Signaling Structures, appropriately sized, and allocated across multiple CFs that each have redundant coupling link connectivity to all systems in the sysplex. XCF will still be able to place all the signaling structures in the same CF if that is the only CF available, so this does not cause any exposure.
- Define no more than 3 or 4 transport classes.
    - Monitor XCF traffic over an extended period to validate TCLASS sizes.
    - Set MAXMSG to 4000 on inbound paths and 2000 on outbound paths as a starting point and monitor for NO Buffer Conditions.
- Ensure there are 2 structures defined for each transport class and placed in alternate CFs.
    - Run CFSIZER Tool to size XCF structures based on number of systems in the sysplex.
- Enable the XTCSIZE function to allow XCF to manage message size segregation. This eliminates the need to tune message size segregation for groups that are not assigned to a specific transport class.

2.8 System Logger (LOGR)

Description/Benefit

- The z/OS System Logger is used by numerous z/OS features (APPC/MVS, OPERLOG, LOGREC, RRS) and by other products such as CICS and IMS. The performance and availability of the System Logger (LOGR) therefore has a critical impact on the availability of applications using these features or products.

Details

- In GDPS environment, use LOGRY and LOGRZ on the K-Systems.
- If possible, isolate the Primary LOGR CDS from all other CDSes. If this is not possible, then at least isolate it from the Primary Sysplex CDS. This is done because of the high I/O rate to this data set when many log streams are in use.
- Don't oversize the LOGR structures. The SMF Type 88 records provide information to help evaluate the utilization of the LOGR structures.
- Place the offload data sets on the fastest available DASD.
- The offload and staging data sets must be allocated with SHAREOPTIONS(3 3).
- Ensure the LOGR CDS is formatted at the current level.

- Try to have at least two active log streams per CF structure, connected to more than one system, to allow peer recovery in case of failure.

- When another log stream is connected to a LOGR structure, the existing log streams will be resized. This could potentially cause short-term log stream-full conditions and impact exploiters. Therefore, try to start all connectors to a given LOGR structure at the same time.

- Code WARNPRIMARY(YES) for all critical log streams to enable monitoring warning messages to be issued for the following conditions:

    - When the log stream primary (interim) storage consumptions is 2/3 between the HIGHOFFLOAD value and 100% full. (Log Stream Imminent Alert Threshold)

    - For a CF based log stream, when a 90% entry full condition is encountered.

    - When an interim (primary) storage full condition is encountered.

- Establish automation and alerting to notify systems and operations of potential LOGR issues.

- Create an IXGCNF00 member for the sysplex and activate. At a minimum review and set up monitoring intervals for warning and actions messages.

- Increase the efficiency of the SMF offload process and prevent loss of SMF data when writing large amounts of SMF data by splitting SMF record types into multiple log streams to increase the number of tasks writing SMF data (1 task per log stream).

2.9 Naming Conventions

To facilitate the implementation of automation, and to reduce the incidence of human error, you should develop meaningful naming and numbering conventions for:

- Sysplexes
- Processors
- LPARs
- Systems
- Subsystems
- Catalogs
- Data sets
- Parmlib and proclib member suffixes
- Consoles
- CF structures

- SMP zones
- Job and proc names
- I/O paths and devices
- Define names such that technical staff and management can easily identify the sysplex, system, device, and so on.
- If possible, configure CHPIDs and device addresses on multiple systems using the same numbering scheme.

When deriving naming conventions for these entities, bear in mind the use of system symbols. These can make the implementation of naming conventions easier, simplify system operations, and reduce the cost of managing a multisystem sysplex.

3.0 CICS

This chapter looks at IBM best practices for maximizing CICS availability in a Parallel Sysplex environment.

3.1 High Availability/Topology

Description/Benefits

- To maximize CICS availability, if there are multiple regions to deliver a CICS function, it should be possible to stop any of those regions and still deliver the service to the user, provided at least one of those regions is still available. If a CICS region experiences a planned or unplanned outage, all the work that the region would otherwise process should automatically get routed to one of the other regions. In the case of an unplanned outage, the failed CICS region should be automatically restarted.

Details

- Implement cloning of CICS regions:
    - Manage affinities in CICS applications.
        - Use CICS Interdependency Analyzer (IA) and Application Discovery and Delivery Intelligence (ADDI) to discover and analyze affinities.
        - Depending on the type and duration, you can manage affinities using CICSPlex System Manager (CP/SM) and CICS IA without any application changes needed.
    - Implement Socket-Owning Regions (SORs) and Terminal-Owning Regions (TORs) on multiple LPARs.
        - Balance end-user logons using VTAM General Resource (GR), Sysplex Distributor, port sharing, Dynamic VIPA, and so on.
        - For LU6.2 sessions, implement VTAM Persistent Sessions or VTAM Multinode Persistent Sessions (MNPS) for faster logon reconnect times.
        - Ensure that SORs and TORs have access to all or most Application-Owning Regions (AORs).
    - Implement CICS regions on multiple LPARs.
- Use ARM and/or your favorite automation tool to quickly restart failed CICS regions in the event of a region or system failure.

3.2 CICS Data Sharing

Descriptions/Benefits

- To provide a high-availability environment, CICS needs to be set up to maximize the availability of the connections between it and the other subsystems. Data needs to be shared through these connections with multiple CICS regions, including regions that are running on different LPARs.

Details

- Implement data sharing for cloned AORs, using:
    - IMS/DB
        - See IMS data sharing section
    - Db2
        - See Db2 data sharing section
    - MQ
        - MQ is dependent on Db2's data sharing group for implementation of an MQ queue sharing group.
            - See Db2 data sharing section
            - See MQ queue sharing group section
    - VSAM/RLS
        - Allow VSAM data to be shared, with full update integrity, among many applications running in one or more z/OS images in a Parallel Sysplex, without the need for a File-Owning Region (FOR) in CICS.
        - Split the SMSVSAM "Cache Set" between two different CFs for availability (and performance).
        - Implement Transactional VSAM (TVS) for sharing VSAM files between CICS and batch with full update integrity.
    - Implement CICS Coupling Facility servers
        - Temporary Storage Queues
        - Coupling Facility Data Tables
        - Global ENQ/DEQ
        - Named Counter Server
- Share the DFHCSD data set among all clones and among the region "types" which comprise an application set of regions (TOR/AOR/FOR).

- Use a single JCL procedure, started with the S procname,JOBNAME=stcname command to start the cloned regions. Use symbolic overrides to create unique instances.

3.3 CICSPlex System Manager (CP/SM)

Description/Benefits

- In a Parallel Sysplex, there is a need to manage multiple CICS regions. The CICSPlex System Manager (CPSM) is a system management tool that enables you to manage multiple CICS systems from a single point of control. It simplifies systems management by providing reliability, availability, and serviceability features.

Details

- Implement dynamic routing for transactions, distributed program links (DPLs), and STARTS using CPSM optimized routing.
- Use CPSM single system image and single point of control to:
  - Provide operational interface for CICSplex (complex or cluster of CICS regions).
  - Enable grouping of regions.
  - Reduce complexity and burden on operations.
  - Ease problem diagnosis for system programmers and operators.
- Isolate the Maintenance Point CICSPlex SM address space (CMAS). This will allow recovery of the CMAS on any available LPAR, as well as insulating the CICS regions from software level changes during maintenance and upgrade cycles.
  - Maintenance Point CMAS should not manage any CICS regions.
- Configure CMASs in an any-to-any configuration.
- Start up CMASs and WUI during z/OS IPLs.
  - CMASs must be fully available before any CICS regions start up.

3.4 Monitoring

Descriptions/Benefits

- To optimize CICS performance, monitoring data can be processed to provide information to help you analyze the performance of your system and detect problems early.

Details

- Review all defaults in the System Initialization Table (SIT).
- Use CICS policies to enable early problem detection.
    - Monitor critical resources and raise alerts.
    - Set thresholds to detect problems early.
    - Use automation facilities such as CICS events to respond to threshold boundaries.
- Utilize storage protection and transaction isolation.
    - If it is not viable to run with storage protection and transaction isolation in production, consider running with these protection mechanisms turned on during testing to discover and correct coding errors.
- Application programs should be AMODE (31).
- Application programs should be reentrant (RENT).
- Consider using terminal autoinstall without cataloging to improve CICS restart times.
- Implement threadsafe processing for application programs.
    - Utilize CICS Performance Analyzer (PA) and CICS Interdependency Analyzer (IA) for analysis of application suitability for threadsafe.
- Collect interval statistics at a minimum of one hour.
    - Recommendation is to set statistics interval to the same as the RMF interval.
    - Use CICS PA to report on collected statistics.
- Configure a large internal trace table.
    - Default size may be too small.

4.0 Communications Server

z/OS Communications Server provides a set of communications protocols that support peer-to-peer connectivity functions for both local and wide-area networks, including the most popular wide-area network, the Internet. z/OS Communications Server also provides performance enhancements that can benefit a variety of TCP/IP applications. z/OS Communications Server provides both SNA and TCP/IP protocols for z/OS.

4.1 Sysplex Distributor

Description/Benefits

- Sysplex distributor extends the notion of dynamic VIPA and automatic VIPA takeover to allow for load distribution among target servers within the sysplex. It extends the capabilities of dynamic VIPAs to enable distribution of incoming TCP connections to ensure high availability of a particular service within the sysplex.

Details

- Need to define DynamicXCF for XCF sysplex support.
- Enable QDIO Accelerator.
    - Defining QDIO Accelerator reduces the overhead of routing traffic through the TCP/IP stack when the packets are not destined for applications on that particular LPAR. This is beneficial for Sysplex Distributor traffic as well as any other traffic that passes through the stack.
- Define Sysplex Routing (IPCONFIG SYSPLEXROUTING) and Server WLM (IPCONFIG SERVERWLM) for WLM input to Sysplex Distributor to enable WLM priorities to be used for Sysplex Distributor.
- Use Sysplex Distributor with VIPA Route which avoids traffic on the XCF links.
- Sysplex Distributor should monitor parameters Recovery and Auto Rejoin to allow automatic removal when sysplex problems are detected, and automatic rejoin when the problem is resolved.
- Use OSPF stub area to provide dynamic routing which is essential to proper Sysplex Distributor function.

4.2 SNA

Description/Benefits

- SNA products recognize and recover from loss of data during transmission, use flow control procedures to prevent data overrun and avoid network congestion, identify failures quickly, and recover from many errors with minimal involvement of network users.

Details

- Implement VTAM Generic Resources to enable generic logons:
  - Ensure proper Coupling Facility structure size and placement.
  - Enable subsystem exploitation of generic resources.
- Exploit APPN and HPR.
- Define two APPN network nodes.
- Use ARM to quickly restart VTAM in the event of a VTAM failure.
- Use XCF for data transportation between systems.
- Exploit VTAM and application cloning via system symbols for easier systems management and dynamic application definitions.
- Configure all systems within a Parallel Sysplex with the same NETID.
- VTAM Generic Resources requires that all systems in the sysplex be part of the same network.
- Provide multiple paths from each host to each network controller, to ensure connectivity can be maintained across the failure of any connection.
- VTAM traffic between hosts should be sent over EE links

4.3 TCP/IP

Description/Benefit

- TCP/IP takes advantage of the Coupling Facility and Workload Manager to optimize availability and load balancing in a sysplex.

Details

- Implement Dynamic XCF to ensure automatic IP connectivity among all the TCP/IPs in the sysplex.

- Exploit VIPA for critical IP applications and enable physical interfaces to be backed up in case of a failure.

- Use dynamic VIPA takeover in case of a failure.

- Use ARM for application server restart, as well as in-place restart for TCP/IP itself.

- Implement Dynamic VIPAs on an application basis to ensure fast server availability after a failure, as seen by clients:

    - Use Automated Takeover for multiple homogeneous servers (such as TSO or Web Server), each of which can satisfy the same client requests.

    - Use Application-Defined Dynamic VIPAs for unique restartable or movable server applications.

- Use Dynamic IP to balance TCP/IP clients over multiple stacks.

- Implement a WLM-supported Domain Name Server (DNS) for workload balancing of traditional business applications.

- Use multiple Telnet (TN3270E) servers with WLM/DNS.

- For Web Server connections, implement a WLM-guided Network Dispatcher to enable load balancing.

- Use application cloning for the CS for z/OS TN3270E server so that there can be freedom of movement without corresponding VTAM definition coordination.

- Use system symbolics to simplify the task of TCP/IP configuration file maintenance.

- As mentioned in the SNA section, ensure that each host has multiple network interfaces. This, along, with Routing Daemons such as OMPROUTE, will help in situations where one of the interfaces had a failure.

- GATEWAY statement is an outdated way to add static routes to the route table. The BEGINROUTES/ENDROUTES statement is the recommended statement for defining static routes because it has more capabilities. Migrate to the BEGINROUTES statement.

- In the output from the D TCPIP,,NETSTAT,ROUTE command, the total number of IPv4 routes should be less than 2000.

- In the output from the D TCPIP,,NETSTAT,ROUTE command, the total number of IPv6 routes should be less than 2000.

- The TN3270E server supports multiple ports. You can define a “well known” backup port so that if the main telnet port (normally port 23) fails, the backup can be used.

- Use DHCP for dynamic assignment of client IP addresses. This makes system administration easier.

- The use of SNMPV3 for network management should be deployed.

- Set TCP/IP storages limits. SOMAXCONN specifies the maximum number of connection requests queued for any listening socket. It is recommended to set SOMAXCONN to 1024 or higher.

- TCP/IP buffers should be at least 64K which enables Dynamic Right Sizing support for high latency links.

- If an FTP Server is running, then TCP Max Receive Buffer Size should be at least 184320 (180K).

- Use INET rather than CINET since the overhead of CINET (multiple TCP/IP stacks per z/OS) is no longer required.

    - CINET is still used by some customers for security reasons. When using CINET it is important not to connect multiple TCP/IP stacks to the same network due to routing issues. The D OMVS,CINET=ALL command should be used to confirm routing is as desired.

- Change the TCP/IP message format from the default "short" to "long".

- Migrate from token ring devices to Ethernet OSA QDIO devices defined with the Interface statement.

- Manually defined MPCPTP IUTSAMEH and XCF between the TCP/IP stack and VTAM for EE, as well as XCF to other members of the same sysplex are not recommended. Migrate from manual MPCPTP definitions to dynamically defined devices using the DynamicXCF parameter on the IPCONFIG and IPCONFIG6 statements.

- If DYNAMICXCF is defined on IPCONFIG or IPCONFIG6 then GLOBALCONFIG SYSPLEXMONITOR RECOVERY should be defined.

- In the output from the D A,L command, if an FTP Server is running, then TCPCONFIG TCPMAXRCVBUFRSIZE should be at least 180K.

- Outdated method of specifying whether the stack should offload TCP segmentation for IPv4 packets to OSA. This parameter will be removed in a later release.

4.4 Network

Description/Benefits

- In just about every installation, the vast majority of the application users are remote to the computer center. Therefore the availability of the resources used to connect the users to the systems is critical. This section contains a checklist of actions for both SNA and TCP networks, to ensure they are configured for maximum availability

Details

- Configure multiple gateways, each having sufficient extra capacity to take over the other's workload in the event of an outage. Each gateway should be processing real work and have automated failover.
- Use a Communications Management Configuration (CMC) configuration:
  - Originally, CMCs were used to own the NCPs and attached devices. However, in an APPN environment, the CMC can be extended to being the DLU Server, the Central Directory Server (CDServer), and also possibly the APPN Border Nodes.
  - Isolate the primary CMC on its own CPC or LPAR image to prevent application-caused outages from affecting the network.
  - Configure a backup CMC on a failure-independent CPC image.
  - Use XCF for communications between the CMC and z/OS images.
  - Configure CMCs as APPN Network Nodes or Interchange Nodes. Interchange Nodes should be used if the CMCs still attach to NCPs and/or other VTAMs using subarea connections.
  - Configure remaining nodes as End Nodes or Migration Data Hosts.
  - Note: We strongly recommend placing the CMC(s) inside the sysplex. However, if the CMC must be outside the Parallel Sysplex, at least one system (and preferably two) within the sysplex must be configured as a Network Node.
- Implement High Performance Routing (HPR) to provide for nondisruptive path switching between nodes:
  - HPR/IP is the preferred method for SNA applications (non TN3270/E) to communicate over an IP backbone.
  - At a minimum, implement APPN/HPR between the gateway and the Parallel Sysplex.
  - If possible, move APPN/HPR out to the end user sessions to establish full end-to-end, nondisruptive recovery from failures.
  - Define backup routes to enable switchover in the event of a network or system failure.

    - If carrying SNA traffic over an IP backbone network, use the APPN Enterprise Extender to maintain high availability network characteristics, including the use of APPN/HPR end-to-end support.

    - If using TN3270E to carry SNA applications over an IP backbone, place the TN3270E servers on the host. The host will generally have better availability than a router, and you have the entire TCP/IP WAN between the client and the server, which means that full TCP/IP re-route will occur on any outage.

- Define interface statements rather than Device/Link statements.

- HiperSockets Multi-Write support should be offloaded to zIIP and IPsec offloaded to zIIP.

- Enable Segmentation Offload. Segmentation Offload is recommended to offload the packet segmentation to the OSA cards.

- Do not use SNMP default password.

- SMF type 119 records should be turned on rather than SMF type 118 records. Type 119 records provide IPv6 support and all the latest enhancements.

- Total direct routes should be less than the recommended 2000 maximum.

- Define AT-TLS and IPsec policies using z/OSMF.

5.0 Db2

This section provides recommendations for Db2 to deliver optimum performance, scalability, and availability in a Parallel Sysplex, data-sharing, workload-balancing, environment.

5.1 Db2 Data Sharing

Description/Benefits

- In Db2 data sharing, applications that reside in multiple Db2 subsystems in a Parallel Sysplex can read from and write to the same Db2 for z/OS data concurrently, with integrity, performance, scalability, and dynamic workload balancing.

Details

- Establish and document naming conventions early on to avoid confusion, eliminate operator error, simplify problem determination, and facilitate changes to the data sharing group.
- Review CPU capacity requirements to ensure there is sufficient CPU capacity available to handle the additional resource overhead associated with Data Sharing: Data Sharing overhead can vary greatly from one system to another, depending on such factors as the degree of inter-Db2 read/write interest, configuration of coupling facilities, CF links, structure sizes, etc.
- Review all ISV software with vendors and verify that products are at the proper release and maintenance levels to support any new Db2 features that you plan to exploit.
- Monitor group buffer pools (GBPs) for synchronous service times and to avoid the three key indicators: directory reclaims, cross-invalidations (XIs) due to directory reclaims, and write failures due to no storage.
- Aim for zero cross-invalidations due to directory reclaims.
- If possible, at least one, and preferably both, CFs should be failure-isolated from the Db2 members that allocate the lock and shared communication area (SCA) structures.
- If the CF containing the Db2 lock and SCA structures is not failure-isolated from the connected Db2s, duplex those structures using system managed duplexing.
- Starting with Db2 12 there's a choice to use synchronous or asynchronous system managed duplexing for the lock structure:
    - ▪ With synchronous system managed duplexing, each synchronous lock request may incur a CPU cost 3 to 4 times greater than were the lock structure in simplex mode.
    - ▪ With Asynchronous CF duplexing updates to the secondary lock structure are performed asynchronously, this makes duplexing CF lock structures more

practical, even at extended distances and provides performance advantages for duplexing the lock structures.

- Ensure that the XCF_CF_STR_AVAILABILITY, XCF_CF_CONNECTIVITY, and XCF_CF_STR_PREFLIST health checks are enabled and that any exceptions that they raise are immediately investigated and addressed.

- Enable Auto Alter for all Db2 CF structures assuming sufficient storage in the CFs.

  - Structures enabled for auto alter could potentially be decreased in size if other structures demand more CF storage. Code MINSIZE to prevent structure from becoming too small.

  - XES can dynamically alter the ratio of directory entries to data elements if auto alter is enabled for GBPs. XES can also increase the GBP allocation up to the SIZE specification.

  - XES can dynamically increase the allocation for the SCA up to the SIZE specification.

  - The Db2 lock structure consists of a modify lock list and a lock table. XES can increase the allocation of the Db2 lock structure up to the total of the SIZE specification, but the modify lock list will consume all the additional storage. If the lock structure shows too high a level of false contention (greater than 1%), and you choose to address it, you then need to rebuild the lock structure to increase the size of the lock table. Auto alter cannot accomplish this on its own.

- Ensure that CFs have sufficient resources (CPU cycles, storage, and link capacity) and that CF Structures are properly sized: refer to CF Sizer in the Coupling Facility section

- Ensure that AUTOREC(YES), the default, is specified for each GBP. This enables automatic GBP recovery in case of a CF failure.

- Implement GBP user managed duplexing. The use of GBP duplexing significantly speeds up recovery if a GBP is lost.

- For initial startup, allocate structures generously and monitor for a period of time, then adjust sizes downward or upward as necessary.

  - Auto Alter may be specified on the Structure statement of the CFRM policy to assist you to tune your GBPs. XES can increase or decrease the size (gradually). It can increase the directory to data ratio. Auto Alter is designed to handle gradual changes in activity, not sudden fluctuations.

  - For GBPs specify the INITSIZE (initial allocation) and SIZE (maximum allocation) parameters, allowing some room for growth, to avoid the disruption of having to modify, install, and activate a new CFRM policy and issue a rebuild to increase the size. Make sure SIZE is no more than 2x INITSIZE; SIZE should be within the range of 120% to 200% of INITSIZE.

  - From the Resource Measurement Facility (RMF) Activity Report Summary, investigate any directory reclaims and increase the number of directory entries when there are "Dir Rec XI".

        - ▪ An excellent Db2 command to use is -DIS GBPOOL(*) TYPE(GCONN) GDETAIL(*). Issue it from any member connected to all GBPs and it will show only those GBPs that are connected (thus reducing output greatly).

- The SCA structure can impact availability if underallocated. Allocate at least 32MB to the SCA initially for small to medium data sharing groups. It contains exception states for objects such as those that are in a Read-Only status, as well as for Copy / Recovery / Reorg / Check Pending.

- Start with a 64MB Lock Structure for a small production environment, monitor false contention; if high, increase the Lock table to the next power of 2.

    - High is > 2% contention and 1% false contention as indicated in the RMF Coupling Facility Activity Report for the Lock Structure (Req. Total, Req. Deferred, False Cont) or the Tivoli OMEGAMON XE for Db2 Performance Expert (PE) Statistics Long report

- Keep the default REBUILDPERCENT of 1% for all structures so that the structure gets rebuilt if there is a loss of connectivity.

- Spread structures across more than one coupling facility in a manner that balances CF CPU and Link utilization as evenly as possible.

5.2 Db2 Recovery Logs

Description/Benefits

- Database logging is an important part of your highly available database solution design because database logs make it possible to recover from a failure, and they make it possible to synchronize primary and secondary databases. All databases have logs associated with them. These logs keep records of database changes. If a database needs to be restored to a point beyond the last full, offline backup, logs are required to roll the data forward to the point of failure.

Details

- Implement Db2 Dual logging support. This is similar to JES2's use of dual checkpoints, where one backs up the other.

- Each set of logs (active and archive) should reside on different volumes/control units/paths. Copy 1 should be on a separate “box” from Copy 2, if possible, or failure isolated in some other way.

- Exclude the Db2 Active Logs and BSDS’s from HSM migration activities.

- Define a sufficient number of Db2 active logs. Optimize the following:

    - Input and output log buffer sizes.

    - Write thresholds.

    - Archive log frequency.
    - Log update rate.

- The size and/or number of active log data sets should ensure a minimum of 6 hours of coverage of peak activity.

- Ensure the two copies of each Db2's active logs are failure isolated from each other.

- Consider using the Archive log option, TSTAMP=YES for ease in identifying archive logs if a problem occurs.

- Optimize Db2 Recovery and Restart times: Evaluate media used for Db2 archive logs and ensure that log access time is optimized. Consider archiving the primary Db2 log files to DASD to facilitate a quick recovery. The second archive copy can be written to tape and used for disaster recovery.
    - Ensure 24 hours of Db2 recovery log records are available on disk, either by archiving to disk, and later migrating to tape, or increasing the active log data sets to 24 hours of disk coverage.
    - Establish automated procedures (use HSM if available) to monitor space within the pool to ensure that sufficient space exists at all times to create new archive logs. In the event that the pool becomes full and Db2 is unable to dump the active logs, Db2 will stop until the situation is corrected and an active log is available again to write to.
    - Automate and monitor the following messages:
        - DSNJ110E - LAST COPYn ACTIVE LOG DATA SET IS nnn PERCENT FULL
        - DSNJ111E - OUT OF SPACE IN ACTIVE LOG DATA SETS
        - Anything in the range of DSNJ100 to DSNJ109
            - For example: DSNJ105I - csect name LOG WRITE ERROR DSNAME=..., LOGRBA=..., ERROR STATUS=cccc

- There should be enough tape drives to be able to support each Db2 in the data sharing group if they each have to mount tapes to go back to Archive logs or to HRECALL migrated DASD archives. This value, specified as DSNZPARM MAXRTU, should be no less than the number of Db2 members in the data sharing group. Alternatively, the number of tape drives can be changed by an operator -SET LOG command. A related value is specified as DSNZPARM DEALLCT with zero, so that a tape drive will be released as soon as it is not used, and its data set becomes available to other Db2 members that may request it.

5.3 Backup and Reorg Processes

Description/Benefits

- In a non-data sharing Db2 environment, when a Db2 subsystem is stopped (or fails), all the data under Db2's control is unavailable until the Db2 subsystem is restarted. Conversely, one of the design points of a Db2 data sharing group is that the Db2 data remains available as long as at least one member of the data sharing group is available. If one of the members of the data sharing group fails, it is almost certain that the failed Db2 is holding locks on Db2 objects when it fails. Until that Db2 is restarted and releases those locks, the related objects are not available to the other members of the sysplex.

Details

- In the event of a Db2 or system failure, automatically restart any failed members (non-quiesced members) to minimize restart time and free retained locks so that its data can be accessed by other Db2 members of the group.
    - Consider using the Automatic Restart Manager (ARM), or another automation tool, to restart the Db2 member in place if a Db2 member fails.
    - Consider using the ARM, or another automation tool, to restart the Db2 member on another z/OS image in the event of a z/OS and/or system failure.
        - Consider using "Restart Light" to quickly resolve retained locks with minimal disruption to other systems. Restart Light applies only for z/OS image failures. Refer to Db2 documentation for the syntax of the -START LIGHT command.
- Ensure that the DSNZPARM value for the RETLWAIT parameter is set correctly for your installation. It is expressed as a multiplier of the resource timeout value (IRLMRWT) that Db2 will wait before returning a "Resource Not Available" condition to the application. Values of 1 or 2 may be appropriate when you have automation perform Db2 restarts in place. Otherwise, specify 0.
- For heavy data sharing with large GBPs, try to dribble castout write activity between GBP checkpoints. CLASST (1-5%) and GBPOOLT (5-25%) are quite common for large GBPs. Monitor aggressively for GBP write failures (target should be 0) through the -DIS GBPOOL command.
- A long-running unit of work may elongate Db2 recovery and restart time. Therefore, incidents of "long-running units of work" should be minimized.
    - Db2 can issue the warning message (DSNR035I) for a long running unit of recovery (UR) based upon the number of checkpoint cycles to complete before the unit of work ends. Db2 will issue this message for a long running UR if DSNZPARM URCHKTH is non zero. But the number of checkpoints depends on several factors which may not include the long running job.
    - Another warning mechanism is based on the number of log records written by an uncommitted unit of recovery, specified by DSNZPARM URLGWTH. Message

DSNJ031I is issued when the threshold is reached. The number of updates is cumulative for a UR and the message is repeated every time the threshold is reached.

- Ensure that policies for application checkpoint, commit frequencies and restart guidelines are established and being followed.
  - Even read only transactions must commit, or they can cause utilities such as an online REORG to fail. REORG must drain even readers during the switch phase, which is extremely short but requires exclusive access.
  - Commits reduce the total number of locks held by the unit of work, reduce “false contention”, and avoid lock escalation.
- Consider putting automation and operational procedures in place to identify and cancel long-running units of work.
- Evaluate system checkpoint frequency for impact on Db2 restart times.
- Create partitioned table spaces to ensure granular utility domains. A restrictive state on one partition is unlikely to affect the availability of the others (limit is up to 4096 partitions).
- Minimize the disruption of ongoing work by utilizing the IBM Storage FlashCopy features to obtain point-in-time copies of data for use in Disaster Recovery.
- Optimize the use of Db2 COPY and REORG to reduce planned and unplanned outage time. Consider SHRLEVEL CHANGE options for both. Updaters are allowed while COPY executes, and updaters are allowed almost all the time during online REORG.
- Utilize Db2’s Online REORG capability. REORG performance can be optimized in a variety of ways, depending on the underlying table spaces or index spaces and concurrently executing processes. Carefully consider REORG control statements to balance the speed and cost of REORG with availability requirements of concurrent operations and business recovery objectives.
  - Some of the keywords to consider include, but are not limited to: DRAIN_WAIT, RETRY, RETRY_DELAY, FASTSWITCH, and SWITCHTIME.
  - Consider gathering inline statistics to avoid separate step of running the RUNSTATS utility to gather statistics
- Optimize Db2 shutdowns:
  - At Db2 shutdown, z/OS performs processing for all Db2 data sets opened since Db2 startup. You can reduce shutdown time by setting the z/OS parameter DDCONS(NO). Setting DDCONS(NO) can reduce Db2 shutdown time by reducing related SMF processing.
  - Stopping DDF earlier in the process can help purge distributed threads, particularly inactive ones.
  - You can use -STOP DB2 CASTOUT(NO) to cause Db2 to shutdown quickly. It causes the GBPs to remain allocated with changed pages not yet written out to DASD. It is intended to be used in situations where you plan to immediately restart Db2; for example, if you are applying service.

- Minimize Db2 Lock Contention
    - Tune to keep Db2 Lock Contention to less than 2%, and keep False Contention to less than half the total of the total data sharing lock contention (1%):
        - Leave the IRLMPROC parameter MAXUSRS at the default 7, unless you will have more than 7 members in the data sharing group.
        - If false contention is high, increase size of the lock structure to the next power of 2. This requires a rebuild of the lock structure.
        - Monitor with RMF Monitor III post processor report.
- Ensure that IRLM is running at a higher priority than the IMS, CICS, and Db2 address spaces, but not higher than the XCFAS address space. IRLM SRB time should always be less than Db2 SRB time. IRLM should run in the SYSSTC service class with the production Db2 address spaces in an importance 1 service class with a velocity goal.
- Evaluate lock escalation parameters to minimize lock escalation. Lock escalation occurs on a table basis; after a certain number of lower level locks (page or row) have been granted the tablespace itself becomes locked which reduces concurrency. The option (LOCKSIZE=ANY) is not recommended as it can greatly affect concurrency. Long running jobs should commit frequently in a data sharing environment.
- Set the IRLM startup parameter DEADLOK to (1,1)
- Turn on SMF 79 for IRLM reports when lock request has waited longer than specified LOCKTIME value (IRLM TIMEOUT) interval.
- IRLM for Db2 should be a higher dispatching priority than the Db2 control region.
- Review DSNZPARM CHKFREQ, PCLOSET and CLOSE=YES parameters
    - CHKFREQ should be set between 1 and 5 minutes unless you use LOGRECS or BOTH for the CHECKPOINT TYPE on installation panel DSNTIPL1. The default for CHKFREQ is 3 minutes.
    - PCLOSET is the number of minutes following the last update before a page set is converted to read only via pseudo close logic. The default is 45 minutes.
    - CLOSE=YES behavior indicates that after a PCLOSET interval a pseudo-closed data set will be physically closed. While this option is beneficial from the standpoint of eliminating GBP-dependency (and overhead of writing to the GBP in the case of the last updating member), when GBP-dependency is re-established (by the first update to that data set), it causes physical open of the data set. This option is appropriate for most page sets as it eliminates data sharing overhead as soon as possible. It also reduces recovery time in a DR situation.
    - CLOSE=NO on the Create or Alter Tablespace DDL statement indicates that the data set is not to be physically closed following pseudo close of the data set. While it does not decrease the GBP overhead, it also does not involve physical close and subsequent physical open of the data set on the first access following the close. This option is appropriate for page sets that are almost always active, as it avoids a lot of open/close activity.

6.0 IMS

This section discusses the sysplex best practices for Information Management System (IMS) to provide the most available, scalable, and high-performing environment.

6.1 IMS Connectivity

Description/Benefits

- There are many ways to connect to IMS. You can connect to the IMS transaction manager from VTAM SNA devices, through MVS APPC, MQ, and TCP/IP to IMS Connect. In addition, you can connect to the IMS database manager directly with CICS, Open Database Access (ODBA), or other means through the architected interface.

Details

- For IMS Connect, add an entry to the program properties table for HWSHWS00 and set it as non-swappable. If this is not done, response times can vary widely, especially at peak demand times.
- Specify a datastore for each IMS and provide either an exit routine or use IMS Connect Extensions to allow routing to multiple IMSs for workload balancing or to address the unavailability of a given IMS subsystem.
- Set ECB=Y and IPV6=Y, if you're using IPV6, in the HWSCFGxx member for best performance.
- Specify NODELAY=Y in the HWSCFGxx member to avoid delays when IMS Connect sends data.
- TCP/IP settings of TCPNODELAY=ENABLE and NODELAYACK minimize delays.
- Use VTAM generic resources with IMS to balance sessions across the available IMS subsystems and minimize disruption to users by routing logons to any member of the group.
- Use the IMS Resource Manager with Sysplex Terminal Management to allow conversations and other terminal status to be continued on another IMS in the event of a failure.
- For Database Control (DBCTL), use care when defining the Database Resource Adapter (DRA) resource (DFSPZPxx). These resources have an impact on various IMS specifications such as partition specification tables (PSTs), scheduling pools, and fast path buffer usage. This can be especially critical if adding new CICS AORs because the resource demand can easily be multiplied.
- For Open Database Manager (ODBM) and ODBA the same considerations apply as with DBCTL.

6.2 IMS Data Sharing

Description/Benefits

- IMS supports both high availability and continuous availability. To ensure the highest availability IMS applications may be configured in multiple instances to support both HA and CA. The applications in IMS should be cloned to run in several IMS subsystems concurrently, and data sharing implemented across LPARs to allow each subsystem direct read/write access to all user data.

Details

- Convert batch programs to batch message processing (BMPs). BMPs have better availability characteristics, such as dynamic backouts after all abends.
- Ensure that all batch and BMP update programs take checkpoints, and that they do so at regular intervals.
- Implement Fast Database Recovery (FDBR) to release locks quickly after IMS, MVS, and system failures.
- Use ARM for IMS Control Regions, IRLM, and common queue server (CQS) address spaces. Do not use ARM for FDBR address spaces since its use with FDBR disables the use of ARM for the IMS Control Region FDBR tracks.
- Use the LOCKMAX option to catch runaway BMPs and batch programs and to identify insufficient checkpoint frequencies.
- Minimize CI/CA splits and data set extensions by:
  - Ensuring sufficient data set space allocation.
  - Performing regular database reorgs.
- Carefully select Program Specification Block (PSB) processing options (PROCOPT) for high volume transactions and large volume batch jobs.
- Carefully select the IRLM startup parameters (PC, DEADLOK, MAXCSA, and so on).
- Ensure that IRLM has a higher dispatching priority than the IMS, CICS, and Db2 address spaces, but not higher than the XCFAS address space.
- IRLM for IMS should be a higher dispatching priority than the IMS control region.
- Consider using duplexed structures for VSO data sharing.
- Consider using System Managed duplexing for the shared message queue (SMQ) and expedited message handler queue (EMHQ) structures.
- Use System Managed simplexing for the IRLM structure. If not, then ensure that the IRLM lock structure is in a failure-isolated CF.

- Verify with system programmers that overflow structures for shared queues and shared expedited message handler (EMH) queues have been specified.

- For IMS, both control regions and BMPs will use the IMSGROUP parameter to simplify the execution of BMPs across multiple control regions.

- IMS IRLMID may use the same value for IRLMs in the IMSplex. Both the task name and IRLMNM must be different for all IMS IRLM address spaces.

- Specify SCOPE=NODISCON for the IRLMs. This is especially important in systems where IMS batch jobs participate in data sharing.

- Every IMS address space, excluding dependent regions, should be a z/OS started task.

- Update operational procedures as needed.

- Implement Common Service Layer (CSL) – Operations Manager. The use of type 2 commands can reach multiple IMSs from a single point.

- Turn on SMF 79.15 for IRLM reports when lock request has waited longer than specified LOCKTIME value (IRLM TIMEOUT) interval.

- Limit number of locks a unit of work can acquire via LOCKMAX.

6.3 IMS Shared Queues

Description/Benefits

- IMS shared queues provide a way for any IMS subsystem in the same shared queues group to process a transaction, no matter on which system it arrived.

Details

- The shared queues function of IMS uses the common queue server (CQS) for managing the queues, and CQS in turn interfaces with XES for Coupling Facility structure access and with the z/OS System Logger to provide recovery of the queues. The following recommendations might help improve CQS performance:

  - Automate the taking of structure checkpoints. Set the frequency and time of structure checkpoints to minimize disruption to your IMSplex.

  - Recovery time is mostly a function of the response time to the shared queues structures. The log read time is typically very fast, and while it adds slightly to the recovery time, it is not a major factor. There are approximately 1.5 structure accesses for each log record processed. The number of log records processed per period of time can be extracted from the SMF Type 88 records created by System Logger. Using that number and the average response time for the shared queues structure multiplied by 1.5, you can estimate the recovery time for that number of log records and determine whether that time is acceptable in the event of a failure. Remember, all log records from all CQSs must be processed to recover the structure. System Logger should have sufficient DASD backing it.

    - Use MSGBASED processing for the CFRM couple data sets. This minimizes the time to quiesce and resume structure activity because the IXLUSYNC function is used by CQS to coordinate structure checkpoints across multiple CQS images.
    - Implement transaction balancing for fast path messages using shared EMH.
    - If using both EMH and full-function shared queues, stagger the structure checkpoints.
    - Verify with system programmers that the two system restart data sets (SRDS) are on separate DASD volumes and separate controllers, if possible, to minimize the risk of losing both data sets.

- The structures used by shared queues include the primary and overflow message queue structures. Only the primary message queue structure is required. However, we also recommend defining the overflow structures. In addition, consider the following:
    - Specify both a SIZE and an INITSIZE for the structures. This allows dynamic altering of the structure size in case a structure starts to become full.
    - Allocate the overflow structures slightly larger than the primary structure maximum size. In the situation where a queue must go into the overflow structure, then all of those messages must be moved or none of them can be moved. If only one or two queue names are causing the primary structure to fill, the overflow structure must be large enough to hold the entire queue of messages.
    - Specify the correct entry-to-element ratio (controlled by the OBJAVGSZ value in CQSSGxxx) to facilitate CQS overflow processing. CQS monitors only the number of elements in use and does not detect whether the entries become full. The average message size is for both input and output messages combined. If in doubt, specify 512 to get a 1:1 ratio.
    - Enable AUTOALTER for the CQS structures to allow XES to dynamically alter the internal characteristics of the structures including the ratio mentioned above. However, if you enable AUTOALTER for the CQS structures, it is advisable to specify MINSIZE as being equal to INITSIZE to prevent the CF from reducing the size of the shared queues structures.
- Use Shared Queues Local First Optimization if you can.
- CQS uses the z/OS System Logger to log message activity. The following list details the best practices for the z/OS Logger as related to CQS log streams:
    - Use failure-isolated CFs to avoid the necessity of using staging data sets or duplexed structures when possible, except IRLM. For GDPS environments or if you use DASD mirroring for disaster recovery, this might not be possible, and the use of staging data sets might be necessary.
    - If using staging data sets, be sure to allocate enough space (much more than the structure) so that the full threshold is triggered by the structure and not by the staging data sets filling.

  - Set the size of the Logger structure and the CFRM FULLTHRESHOLD value to provide ample warning time to take action in the event of an offload failure.

  - Make the offload data set size (LS_SIZE in the Logger policy) large to avoid frequent new data set allocations. New allocations are more costly and time consuming than offloading to an existing data set.

  - Set the log stream LOWOFFLOAD value to 0 so that all data is moved to DASD after the offload process begins. For this type of log stream, sometimes called a funnel log stream, there is no benefit in keeping some of the log blocks in the structure.

  - Set the log stream HIGHOFFLOAD value to between 50 and 70% to avoid filling the structure before offload can complete. Together with the structure size, you can also provide early warning if the offload fails for any reason.

  - Set MAXBUFSIZE to 65276 (the largest possible buffer size) to use a 265-byte element size. This setting optimizes the use of the storage in the CF.

  - Specify ALLOWAUTOALT(NO) for the Logger structures. Logger monitors and alters the structure characteristics itself if necessary.

  - If you are using both full function and fast path shared queues, use separate structures for the full function and fast path log streams. They are used differently, and therefore management is easier when separate.

- The CFRM for database buffer pools should include space of all IMSs accessing the IMS shared queues.

  - For example, IMS A has 100 buffers and IMS B has 200 buffers, the sizing must include both at 300 buffers.

- Prevent ISRT (‘A7’) when the number of queue buffers in use by application exceeds a defined threshold value.

- OLDS buffers default to 5. Define a buffer count high enough to avoid write waits.

- Allocate OLDS as a multiple of 4K - 24576 is good – 2 blocks per track to allow Hyperwrite and zHiperFICON use.

- Allocate OLDS as SMS and Extend Format dataset to allow OLDS buffers to be 64 bit.

  - DFSDFxx Logger Section BUFSTOR=64

- Size OLDS data set for effective use.

  - An OLDS should be able to log longer than the length of time the Archive Job uses to copy it.

- Archive job problems / unexpected log volume

- Define additional DFSMDA Members for extra OLDS data set pairs

- Pre-Allocate at least one (large) extra OLDS data set pair without it being started in IMS.

- Automate on IMS message: DFS3258A LAST ONLINE LOG DATA SET IS BEING USED
- /STA the pre-allocated extra OLDS.
- Monitor virtual storage utilization of IMS Subsystem Address Spaces:
    - CTL,DLS,DBRC,IRLM,CQS,SCI
    - Track PVT/LSQA, EPVT/ELSQA
    - ECSA/CSA if not done by z/OS team
- Configure RMF to collect data for IMS address spaces.
    - ERBRMF00 in SYS1.PARMLIB
        - VSTOR(IMSPCTL)
        - INTERVAL(15M)
    - Run ERBRMFPP
        - REPORTS(VSTOR)
- Periodically use the DC Monitor.
    - Establish expectation and Base lines.
- Use IMS Performance Analyzer if available.
    - Uses current SLDSs.
    - All DC Monitor data.
    - IMS Connect Extensions data.
    - Granular and Expanded reporting.
- z/OS Region Size should not have an arbitrary limit. It should be 0M, or value close to maximum available.
- IMS Pool Specifications
    - Limit pools allocated in CSA/ECSA as appropriate to protect the z/OS system
    - Code no limit or largest possible value for pools allocated in EPVT
    - Some pools use LRU algorithms and may in some cases work better at smaller sizes, due to reduced overhead in pool management
- For synchronous (CM1) messages use Open Transaction Manager Access (OTMA) Message Flood Protection.
    - Default is 5K per TMEMBER. Resize as needed if appropriate.

    - In periods of IMS problems, consider turning off CM1 input rather than attempting to process it.

- For asynchronous (CM0) messages limit the number of TPIPEs.

- Have spare WADS data sets available.

- Running resources

    - Dynamic Allocation for ACBLIB

    - IMS Manages Directories as needed in a managed ACBs environment.

- OTMA ACK Timeout default is 120 seconds

    - Consider lowering it for your environment

7.0 MQ

MQ enables applications to participate in message-driven processing across the same or different platforms. The MQ flavor that meets the most stringent requirements for both the response time for messages and also the availability of the service (minimizing planned, unplanned outages) is the use of MQ shared queues in a Parallel Sysplex.

7.1 MQ Queue Sharing Group

Description/Benefits

- In a Parallel Sysplex, you can configure multiple MQ queue managers as a queue sharing group (QSG). Queue managers in a QSG support two types of local queue: a shared queue and a private queue. The queue managers in a QSG cooperate to maintain and access shared queues in one or more z/OS coupling facilities. In this way, all the queue managers own a shared queue. Using shared queues with your applications, you are able to exploit advantages of the Parallel Sysplex, such as high availability, workload balancing and reduction of administrative and operational work.

Details

- Review "WebSphere MQ in a z/OS Parallel Sysplex Environment, SG24-6864-00" from ibm.com/redbooks which discusses setting up and using MQ in a sysplex environment to improve throughput and availability of applications.
  - Some of this information is dated but it is generally a good starting point. It is missing some of the new features and functions.
- Review the MQ Capacity Planning SupportPac, MP16 for information about likely performance and Coupling Facility resource requirements.
- Implement Dual Logging and Dual BSDS support for each subsystem.
- You should have sufficient active logs to ensure that your system is not impacted in the event of an archive being delayed. In practice, the minimum should be four active log data sets but many systems have many more in their active log string, up to 310. The last output of your CFSTRUCT BACKUP command should ideally be on an active log.
- Your active logs should be a minimum of 3GB and can be as high as 4GB. If your logs are switching more than once per minute, either the size needs to be adjusted or you should think about breaking up your workload.
- SupportPac MP16 contains the estimated required log space.
- The standard recommendation for 4 bufferpools for a production system is:
  - BP0 –50,000 pages – reserved for the queue manager
  - BP1 –20,000 pages

    - BP2 –50,000 pages
    - BP3 –20,000 pages

- The restart mechanism can be manual, use ARM, or use system automation if you ensure the following:
    - All page sets, logs, bootstrap data sets, code libraries, and queue manager configuration data sets must be defined on shared volumes.
    - The subsystem definition must have sysplex scope and a unique name within the sysplex.
    - The level of early code installed on every z/OS image at IPL time must be at the same level.
    - TCP virtual IP addresses (VIPA) must be available on each TCP stack in the sysplex, and you must configure MQ TCP listeners and inbound connections to use VIPAs rather than default host names.
- You can use the Automatic Restart Manager (ARM) to restart all the systems involved in the failure (CICS, DB2, and MQ for example), and to ensure that they are all restarted on the same new processor. This means they can resynchronize, and enables rapid recovery of in-doubt units of work.
- Try to configure your queue sharing group so that every structure has at least two connectors, and, if possible, those connectors should be located on two CPCs, maximizing the chance of there always being at least one active queue manager still connected to the MQ structures.
- Examine the applications getting messages from the queue to determine the match option used on the get so the proper index can be set prior to migration to shared queues.
- Affinities: Removing the affinities between a queue manager and a particular z/OS image allows a queue manager to be restarted on a different z/OS image in the event of an image failure:
    - All page sets, logs, bootstrap data sets, code libraries, and queue manager configuration data sets must be defined on shared volumes.
    - The subsystem definition must have sysplex scope, a unique name within the sysplex, and be defined on each LPAR in the SYSPLEX.
    - The level of “early code” installed on every z/OS image at IPL time must be at the same level.
    - TCP virtual IP addresses (VIPA) must be available on each TCP stack in the sysplex, and you should configure MQ TCP listeners and inbound connections to use VIPAs rather than default host names. Each queue manager should have two listening ports, one is a shared port and one is a private port so there is no port clashing.

7.2 Coupling Facility Resources for MQ

Description/Benefit

- A queue sharing group uses Coupling Facility list structures to hold data. These structures are known in MQ as an administrative structure or an application structure. Application structures are defined to MQSeries as CFSTRUCT objects. An administrative structure is used for communication between queue managers in the Queue Sharing Group, and for management of units of work which involve shared queues.

Details

- A minimum of three CF structures are required. One is the CSQ_ADMIN structure used by IBM MQ. One is the CSQSYSAPPL structure for group units of recovery by client applications in CICS. All other structures are application structures used for storing shared queue messages.
- Up to 512 shared queues can be defined in each application structure. We have seen no significant performance effect when using a large number of queues in a single application structure. There can be up to 64 application structures in a queue sharing group.
- Use the IBM CF Sizer to size the administration structure.
- One or more application structures hold the messages which are resident on shared queues. The size of structure required for these will be application dependent. To estimate an application structure size use the IBM CF Sizer for MQ tool.
- MQ manages the backup and recovery of persistent messages held on an application structures through the use of simple commands. Automate the backup of CFSTRUCTs.
- The administration structure size depends on the number of queue managers in the QSG, and the CFCC level. It is recommended that the size of the administration structure is at least 20 MB. See MP16 for a chart on admin structure size.
- CFRM policy definition for IBM MQ CF structures
  - Consider making SIZE double INITSIZE.
  - It is recommended to define SIZE to be not more than double INITSIZE. The value of SIZE is used by the system to pre-allocate certain control storage in case that size is ever attained. A high SIZE to INITSIZE ratio could effectively waste a significant amount of CF storage.
  - Consider making MINSIZE equal to INITSIZE, particularly if ALLOWAUTOALT(YES) is specified.
- Consider adding additional queue managers to the queue sharing group to provide additional logging 'bandwidth', if required, for CFSTRUCT backups.
- Backup of all your CF structures a minimum of once per hour, to minimize the time it takes to restore a CF structure. The critical factor in media recovery scenarios is the amount and

location of log data which must be reapplied to fuzzy pageset and structure backups to bring them 'up to date'. MP16 contains an analysis of restart times and tuning information.

  - You could perform all your CF structure backups on a single queue manager, which has the advantage of limiting the increase in log use to a single queue manager.

  - Alternatively, you could spread the backup processes across all the queue managers in the queue-sharing group, which has the advantage of spreading the workload across the queue-sharing group

8.0 WLM

Workload Manager (WLM) monitors a sysplex and determines how to allocate the resources for all the work in a Sysplex to meet the goals that you have defined for it. It also reports data about the work.

8.1 General

- Establish meaningful naming conventions for the different constructs used in the WLM policy. This will allow for easier identification of elements such as Service Classes, Report Classes, Classification Groups, or workloads.
- Establish a WLM policy management process for backing up, modifying, activating, and restoring your WLM service definition.
  - Before modifying your WLM service definition, always save a copy of the current version as backup. Keep a backup on a separate DASD subsytem to safeguard against failure.
- Limit use of Resource Groups to where absolutely necessary.
- Use batch initiator management (WLM managed initiators) to spread batch work across the sysplex.
- Any VTAM application should use the VTAM Generic Resource function to distribute sessions across a sysplex.

8.2 Service Classes

Description/Benefit

- A service class is a named group of work within a workload with similar performance goals, resource requirements, and business importance. WLM manages each group of work according to the performance goal assigned to the service class, and the business importance assigned to that performance goal.

Details

- Define a WLM service definition and workload goals to meet your business objectives.
  - Identify business/application requirements and establish service level objectives.
  - Make your “loved ones” the most important work in the system (importance 1 or 2).
  - Establish goals that reflect your true requirements.
- You must define at least one service class.

- Each service class must contain at least one performance period. You can specify up to eight performance periods in a single service class but the recommended value is no more than two periods for most work. You use performance periods to assign service goals and importance levels to a service class for a specific duration.

- Keep the number of active service class periods to a range of 25-35. This helps ensure WLM responsiveness in managing all service class periods to their goals during periods of high CPU contention. There are a couple reasons why it helps to limit the number of service classes:

    - WLM Responsiveness - at each 10 second policy interval, WLM selects the service class period in the most need of help and takes one policy action on its behalf. The more service class periods that need assistance, the more 10 seconds cycles needed for WLM to take the needed policy actions on behalf of these service class periods.

    - Better WLM Decisions - as part of its decision-making, WLM makes projections for how work will behave when resource changes are made. The accuracy of these projections depends on the quality of history data gathered for a service class period. The more similar work that is grouped together in a single service class period, the more statistically valid the history data, and the better WLM decisions will be. If the service class period is too small, consider reducing the number of periods. If the service class only has one period consider combining with another service class.

- Four types of goals can be defined for any service class

    - Average Response Time

        - Good for stable and predictable workloads where long running transactions are abnormal

    - Percentile Response Time Goal

        - Certain number of transactions must run below specified response time, average does not matter

        - Best for workloads where periodic transactions will run long

    - Velocity Goal

        - X% of the time this service class wants a resource it will get it

        - For a sysplex with mixed machine types, Velocity Goals for the same work can achieve different velocities on processors of different speeds and/or different numbers of CPs. Some helpful hints for selecting Velocity Goals in this environment are:

            - Higher Velocity Goals are more sensitive to processor differences, so select a goal that is attainable on the smaller processor(s).

            - Do not try to be too precise in selecting a number, small differences in velocity goals have little noticeable effect. For example, velocities that represent “slow”, “medium”, and “Fast”, might be 10%, 40%, and 70% respectively.

        - Define a high velocity goal for CICS and IMS regions to ensure fast initialization, with the transactions defined with Response Time Goals.

        - Reevaluate Velocity Goals when you turn on or off I/O priority queuing or batch initiator management. These two functions change the velocity calculation. The projected velocity values ("migration velocities") are reported in RMF to help you decide on new goals before you migrate to these functions.

        - Consider having classification by system name for Velocity Goals.

    - Discretionary Goal

        - No business objective, never considered missing its goal

        - Only service class goal with mean time to wait

- Response Time Goals have several advantages over Velocity Goals, such as:

    - Velocity Goals must be reevaluated whenever hardware is upgraded to reflect the difference in velocities due to differences in processor speed or number of CPs. Whereas Response Time Goals, based on End-User requirements, would not necessarily change across an upgrade.

    - CICS workloads, using Velocity Goals, will get storage protection only after a page delay problem is detected by WLM. CICS workloads, defined with Response Time Goals, is assumed to be interactive and WLM proactively protects its working set, even before page delay issues are detected.

- Always review any of your goals after a processor change, OS upgrade, software/middleware upgrades, etc.

- Consider using low importance levels and/or resource groups for work with the potential to dominate system resources.

- Avoid setting to many aggressive goals that cannot be met, otherwise WLM will spend a lot of time managing and not getting other work done (e.g. High Velocities - over 80). WLM may also decide to skip some service classes for some length of time if the goal cannot be met.

- Ensure that WLM has good information available for decision making. To provide sufficient information for WLM:

    - Ensure there are enough completion's for Response Time based service classes.

    - Ensure there is enough ready work for Velocity Goals based service classes.

- Performance Index (PI)

    - PI is the comparison of how well the work in a service class period is performing against its goal.

    - Allows easy way to compare different service classes with different goal types.

  - PI of 1.0 means workload is exactly meeting the goal specified.
  - PI of greater than 1.0 means workload is missing its goal.
  - PI of less than 1.0 means workload is beating its goal.

- A periodic review of the WLM policy with performance data should be done at least once to twice a year to ensure goals are set so that the PI value is close to 1 when the work is running as expected.
- Use the RMF Post Processor reports to review the actual response times or velocities for the service classes defined.
- There are five importance levels, plus discretionary, which can be used for all user defined service classes. Importance levels tell WLM, and therefore you, which workloads can accept delay first.
  - Use all importance levels whenever possible. This will help to identify which workloads will be affected first and if importance one work is close to maximizing use of the system.
  - Importance levels in WLM need to be discussed and defined within and amongst business groups and management.
  - Review your WLM policy periodically to ensure service classes don't drift higher in importance.
  - Solve performance problems through diagnostics, not changing WLM.
- Use importance levels to ensure CPU Promotion does not end up higher than expected.
  - Enqueue promotion level is dynamically calculated.
    - If most of your work is at importance 1, enque promotion will happen at the importance 1 level.
- Started task classification rules:
  - Ensure that "system" work: such as VTAM, JES, RACF, TSS, etc.; are assigned to the SYSTEM and SYSSTC service classes.
  - Limit use of SYSSTC service class to highly critical and short running started tasks (Db2 IRLM, IMS IRLM, etc.).
    - Monitor the SYSSTC service class's CPU usage, if high and/or online systems are experiencing delay, adjust SYSSTC by moving some of the heavier CPU users to a different service class.
  - Let all z/OS system tasks that default to SYSTEM go to System. z/OS-provided address spaces, such as CONSOLE, GRS, DUMPSRV, and so on, should be allowed to default to the SYSTEM service class.
  - Use the SPM rules to ensure high importance tasks are in SYSTEM and SYSSTC.
  - For the remainder of the started tasks, create as few service classes as possible.

    - Db2 should be an importance 1, high velocity, and CPU critical set to yes.

- Server Management – using CICS and IMS transaction goals
    - WLM creates Internal Service Classes (ISC) for group of CICS/IMS regions running the same set of transaction service classes.
    - Minimize CICS and IMS transaction goals to simplify WLM management
    - Minimize the number of transaction service classes to reduce ISC creation. This creates more predictable results for WLM decisions.
    - Applies to all transactions which can run in a common set of regions.
        - Transactions isolated to their own region can have their own goal.

- Ensure that you have default classification rules for all subsystem types used.
    - If there are no classification rules for a subsystem type, the default is SYSOTHER with a discretionary goal. Aim to never have work run in SYSOTHER.
    - Ensure enclave transactions are properly classified in the WLM service definition to prevent them from being managed as discretionary work.
    - The most common error is not classifying Distributed Db2 Transactions (running under the DDF subsystem type) and some TCP/IP enclaves and having them end up managed as discretionary work.

- Group service classes by a workload type, type of work running.
    - Keep started task separate from batch jobs, DDF transactions separate from started tasks, etc.

9.0 Other Sources of Information

This section includes information about IBM Z Subject Matter Expert (SME) contacts and documentation.

9.1 Contacts

Document Author	Jack Billings	jack.billings@ibm.com
Document Author	Jovanna Hadley	jovanna.hadley@ibm.com
Document Author	Jim Thomas	jimbo@ibm.com
Parallel Sysplex & z/OS	Gene Sale	esale@us.ibm.com
CICS	Guy Shevik	gshevik@us.ibm.com
CICS	Leigh Compton	lcompton@us.ibm.com
Comm Server	Linda Harrison	lharriso@us.ibm.com
Comm Server	Chelsea Jean-Mary	chelsea.t.jean-mary@ibm.com
Db2	Mark Rader	mrader@us.ibm.com
IMS	Dennis Eichelberger	deichel@us.ibm.com
MQ	Carolyn Elkins	elkinsc@us.ibm.com
WLM	Brad Snyder	bradley.snyder@us.ibm.com

9.2 Additional Guides and Documentation

- IBM Knowledge Center
- IBM CF Sizer
- IBM Z with z/OS Resilience Best Practices Guide
- Getting Started with IBM Z Resiliency
- Parallel Sysplex Availability Checklist
- Scaling the Sysplex on IBM Z
- System Z Parallel Sysplex Best Practices
- Mission: AVAILABLE
- ABC's of z/OS System Programming: Volume 5
- Achieving the Highest Levels of Parallel Sysplex Availability

- Parallel Sysplex Application Considerations
- MQ SupportPac MP16