Discovering Mainframe Performance Opportunities: When the World Is Not Spinning Fast Enough

Norman Hollander
Senior Consultant, Mainframe West
CA Technology Services
norman.hollander@ca.com
310-957-3737

April 2005

Overview

- Mainframe Environments continue to grow and become more complex at a rapid pace
- z/Series Hardware and Associated Peripherals are challenging for most Customers
- Keeping Up with z/OS (z/VM, z/VSE) Technology is Time Consuming
- The number of IT professionals at most Customer Sites has NOT grown with the Environment
- Many Customer Sites have discovered that their previously well-behaved Systems are just not performing well

Overview

- Adding New System Images may cause the Entire Environment to NOT perform well
- Creating D/R, Development, Q/A, Test, and Other Sandbox Environments may cause the Entire Environment to NOT perform well
- Legacy Applications may no longer be performing well
- Even install CA Products may be suffering

Reality Check

"Users constantly call to complain about slow or poor system performance.

The phone never rings off the hook with calls of system performance that is too good or too fast."

-ndh

About Me

- With CA since January 2004
- In Data Processing since 1971
- Specialized in Operating Systems, Programming Languages, Hardware Configuration and Planning, System Programming, Capacity Planning, Performance and Tuning
- Worked in Communications Industry (AT&T, PacBell), Banking Industry (First Interstate, Home Savings of America), Power Industry (SCE), and Software & Consulting Industry (Candle, Independent)
- Since 1992, have been involved with S/390, Basic and Parallel Sysplexes, Coupling Facilities, Workload Manager
- Have presented sessions at IBM Expo, SHARE, CMG, CAWorld

About You

- Name
- How Long with CA
- Department
- Location
- Responsibilities
- What You Want to Get Out of this Class

Topics to be Covered

- The Elements of Performance
- Discovering Delays and Degradation
- Discovering the Operational Environment
- z/Series Technology
- Workload Manager
- Anatomy of a Performance Health Check
- Leaving no Stones Unturned
- Collecting Performance Data
- Generating Recommendations
- Revisiting the Process
- Q&A

Robbing Peter to Pay Paul- the Elements of Performance

- There are only 3 Elements of Performance
  - Processor
    - General OS Engine
    - Coupling Facility Engine
    - IFA or zAAP Engine
    - IFL or Linux Engine
  - Storage (Paging)
    - Real Storage
    - 64-bit Architecture
  - Input/Output
    - Peripheral Devices
    - Coupling Facilities
    - Communications Devices
    - Security Devices

Processor

- z/Series Operating Systems run on z/Series Hardware
- CMOS and Multiprise technology is no longer supported
- z/800, z/890
- z/900, z/990
- Current Fastest Engine is 450 MIPS
- Biggest Challenge in Faster, Fewer Engine Environments
  - May be an increase of MIPS
  - May cause Queuing Complications
  - May release resources for Latent Demand
  - Uni-Processor Environments should be avoided

Input/Output

"The BEST I/O is NO I/O!" -ndh

- Response Time to a Device is key
- ESCON and FICON Channels are typical
- Parallel and Byte Multiplexer Channels no longer available
- Highly Cached Devices (2-3 milliseconds)
- High-speed Communication Devices (OSA, Hipersockets)
- Bigger, Faster Tape Devices
- Faster, and Virtual, CF Links (15-20 microseconds)
- Faster ICSF Security Devices
- Larger than 32K Blocks of Data

Storage/Paging

"The BEST Page is NO Page!" -ndh

- Only Real Storage Available
  - 2 GB was the limit prior to z/OS
  - Large Storage Images now common for DB2 and Websphere Application Services
- No Expanded Storage
- 64-bit Architecture
  - Above the Line- 16MB
  - Above the Bar- 2GB
- Paging to DASD Devices
  - Must have sufficient Paging volumes for several 2GB simultaneous Dumps

Discovering Delays and Degradation

- The Real Challenge is discovering:
  - "What am I waiting for?"
  - "I have plenty of work to do."
  - "I should NOT be waiting."
- Back to 3 Elements
  - CPU, Storage, I/O
- What's Holding me Up?
  - Waiting for Device
  - Waiting for Volume
  - Waiting for Resource
  - Waiting for System Services
  - Waiting for JES Services
  - Waiting for Coupling Facility Process
  - Waiting for Communication Process
  - Waiting for Security Process
  - Waiting for Java Process
  - Waiting for Another Application
  - Waiting on Myself
  - Waiting on the UNKNOWN

What's Holding up the Application?

- Only 2 Areas to look into
  - Internally
    - Within the Application (faulty code?)
    - Within a System Service
  - Externally
    - Within the Same Image
    - In Other Systems

Can Optimization or Modernization benefit?

What's Holding Up the Operating System?

- Only 2 Areas to look into
  - Internally
    - Within a Server Subsystem
    - Device Allocation
    - HSM, SMS
    - Tape Library
    - XCF/XES
    - JES
    - Dump
    - WLM
    - Reserve, Enqueues
    - Catalogue
    - DBM, RRS
    - LLA, VLF, DLF
    - Within a System Service
    - Capacity
  - Externally
    - Within the Same Sysplex
    - Reserve, ENQ
    - WLM

What's Holding Up the I/O?

- Only 2 Areas to look into
  - Internally
    - Device Allocation
    - Dynamic Allocation
    - Reserve, ENQ
    - Catalogue
    - Cache Controller Issue
    - Cached Volume, DFW
    - IOSQ, PAV
    - Fragmented Volume
    - Fragmented Data Set
    - Within the Application (faulty code?)
    - Within a System Service
  - Externally
    - GPDS
    - XRC
    - Other Images doing the above

General Tuning Methodology

- Start from the Outside and Move In
  - ➢Want to Avoid making Application Changes Until the Rest of the System is Well Tuned
- Check Hardware
- Check Operating System
- Check Sub-Systems
- Check Peripherals
- Check System Services
- Check APARs and HIPER PTFs
- Check I/O Avoidance

Discovering the Operational Environment

- The Days of a Single Application on a Single Processor are long gone
- Need to determine all the components that make up the application
  - Hardware Components (general CPs, IFLs, CFs, zAAPs)
  - Operating System Components (SMS, HSM, ATLs)
  - Transaction Subsystems (CICS, CICS MRO, IMS)
  - DBM Subsystems (DB2, UDB, SAP, Datacom)
  - Message Transmission (Websphere MQ)
  - UNIX System Services (USS, TCP/IP)
  - Websphere Application Server

Hardware

- Processors (next several slides)
- ESCON, FICON Channels and Directors
- Cache Controllers
- DASD farms (Shark, Symetrics, HDS)
  - ➢Extended Copy, SRDF, Flash Copy, etc.)
- Tape Drives, Silos, ATLs, VTS,
- Unit Record Devices (Printers)
- TP (OSA, TCP/IP, SNA)
- Security Devices (ICSF, Crypto)

Central Processor Complexes

- Currently up to 24 Engines per Image
- z/800 and z/900 phasing out
- z/890 and z/990 will get Enhancements
- Number of Engines in each Image needs Consideration
- In-and-Ready Queue is a key Indicator of Stress
  - The In-and-Ready Queue should NOT exceed 2 times the number of assigned Logical Processors
- MIPS is NOT a good Indicator of Performance
  - It is a Marketing Tool to indicate relative Power of One Processor Model to Another

Coupling Facilities

- A means to Share Memory among many different Images
- May be Integrated, May be a Stand-alone footprint
- 3 Types of Structures
  - Lock
  - List
  - Cache
- Processor Utilization for a CF should NOT exceed 50%
- Good Cruising Altitude is around 30%
- Performance Indicators are the Service Time for Particular Structures
  - For z/990s, SYNC Operations should be in the 20-25 microsecond range, ASYNC Operations should be in the 75-100 microsecond range
  - DB2 Structure may vary greatly
  - System-managed Duplexed Structures should be carefully considered

Coupling Links

- Multiple Flavors Available
  - Fiber (slow)
  - Copper (fast)
  - Internal (fastest)
- Performance Indicators are:
  - Subchannel Delays (should be less than 10%)
  - Sharing Links in a Production Environment may cause Performance Challenges

Coupling Facility Impact

- Structures Used for:
  - Basic Sysplex Resource Sharing
  - Parallel Sysplex Data Sharing
- Who Are the Players?
  - XCF- JES, Consoles, VTAM Generic, Shared HFS, RMF, CCI
  - JES2 Checkpoint
  - Catalogue (ECS)
  - HSM
  - Security Packages (RACF, TSS, ACF/2)
  - CICS, IMS, DB2, Datacom, RRS
  - WLM, USS Enclaves
  - WLM, IRD
- Number of Coupling Facilities
  - Recoverability
  - Re-configuration
  - GDPS

Application Processors

- Integrated Coupling Facilities (ICF)
  - Provide Resource and Data Sharing Environment
  - Continuous Availability and Recovery a Key Consideration
- Integrated Facility for Linux (IFL)
  - Provides inexpensive Processor solution for Linux Workloads
  - May run as dedicated Processor, Virtual Imaging Facility (VIF, no longer used), or for z/VM
- Integrated Facility for Applications (IFA, zAAP)
  - Provides inexpensive Processor solution for JAVA Workloads on z/890 and z/990 Processors at z/OS 1.6 and above
- Any of these Processors may cause a Performance Bottleneck

Logical Partitions (LPARs)

- 2 Modes for Processor Complexes
  - Basic Mode, Single Image (SI) running on the CPC footprint
  - LPAR Mode, PR/SM firmware, Multiple Images (any OS) running on the CPC footprint
- z/890 and z/990 now only Support LPAR mode
- Processors may be Dedicated or Shared
- Processors may be Reserved for future inclusion in the Image
- Intelligent Resource Director (IRD) may effect the Shared Processor Configuration

Processor Weights

- A Specification Assigned to All Images with Shared Processors to Determine Access to the Processors
  - Needed when Physical Processor is at 100% Busy
  - The Higher the LPAR Weight the More Access to the Physical Processor it Receives
- Image Performance may be effected when Weights are Enforced
- Intelligent Resource Director (IRD) may effect the Weights of Managed LPARs

LPAR Capping

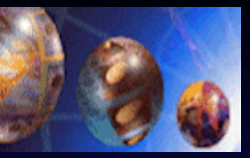
- Capping is Used to Enforce Weights when a Physical Processor is NOT at 100% Busy
  - Typically Used for Non-Production Images or Service Bureau Environments
  - Nullifies IRD
- Image Performance may be effected when Capping is Turned On

Wait Completion

- A Specification Set for an LPAR which Effects how PR/SM Management is performed
  - Generally LPARs are Interrupt Driven
  - Time Slice is Given Up When Engine Goes Into Wait
- Wait Completion is Set to YES for a Single Image on a LPAR mode CPC, or when there is a High-Importance Image running on the Footprint
  - Image Performance may be effected when Wait Completion is in Effect

Intelligent Resource Director (IRD)

- Previously Units of Work were Routed to Images with Available Resources
  - Overhead to Manage could be High
  - Licensing may cause Challenges
- IRD moves Resources to where the Workload is, based on WLM Requirements in a Sysplex Cluster
  - Weight Management
  - CPU Management
  - Channel Management
- Image Performance may be effected when IRD is Responding to WLM Decisions



Variable Workload License Charge (VWLC)

- A Licensing Scheme used to Reduce Software Costs
- Using 4-hour Rolling Periods tied to a Maximum MSU, Capping may be Imposed to Maintain the negotiated License Charge
- Image Performance may be effected when VWLC Capping is Enforced

Cross System Communication Facility (XCF)

- XCF is Used for Sending and Receiving Messages in a Resource Sharing Sysplex (CF not required)
- Transport Classes are used to Communicate Among ALL the Members of the Sysplex
  - When XCF messages are bigger than the defined Transport Class Buffer Length, XCF must find a buffer large enough to contain the “too big” message
  - When XCF message are smaller than the defined Transport Class Buffer Length, XCF uses the entire buffer for the short message
  - Excessive Transport Size uses more Virtual Storage. More overhead to transmit undersized messages in oversized buffers
  - When no suitable Transport Class is available, XCF dynamically adjusts the Buffer Length for the only Transport Class to accommodate the message, thus creating overhead. All XCF traffic among the systems essentially stops while this process occurs. As smaller messages occur after the “too big” message, XCF will re-adjust the Buffer Length down, thus creating more overhead. All XCF traffic among the systems will stop again, while this process completes. XCF Services allow authorized applications on 1 system to communicate with applications on the same system or on other systems
- The XCFAS Address Space may consume Large Amounts of CPU Resources when NOT properly Configured
  - Applications may Suffer Performance when they have to Wait on XCF
  - 3 Transport Classes (1K/8K/20K) have worked the Best in Most Environments

XCF Extended Services (XES)

- XES is an Extension of XCF using the Coupling Facilities
- XES runs as a Sub-Task in the XCFAS Address Space
- XES manages the movement To and From Structures in the Coupling Facility
- Software Managed and System Managed Structure Duplexing may cause an increase in CPU Utilization (18-22% has been observed)
- Additional Processor Resources should be added when using XES

Global Resource Serialization (GRS)

- Provides Integrity Protection for Global and Local Resources (ENQ and Reserves)
- May Use XCF for Messaging among Sysplex Members (Ring Mode)
- May Use Coupling Facility for Structure Messaging among Sysplex Members (Star Mode)
- May Use CTCs for Messaging among Sysplexes or Virtual Guests (Ring Mode)
- May Cause Performance Challenges if NOT properly configured
  - ➢ In Ring Mode, RESMIL should be 1 (not default of 10)
- Enqueues SYSIGGV2, SYSZTIOT, SYSZVOLS, and SYSDSN may cause delays

Multi-Image Manager Integrity (MIM MII)

- Provides Integrity Protection for Global (ENQ and Reserves)
- May Use XCF for Messaging among Sysplex Members
- May Use Coupling Facility for Structure Messaging among Sysplex Members
- May Use CTCs for Messaging among Sysplexes or Virtual Guests
- May Use Data Set on Shared DASD for Control File
- Even if MIM MII is active, GRS is still used for Local Resources
- MIM MII needs to have a High Dispatch Priority
- May Cause Performance Challenges if NOT properly configured
  - MODE=DEMAND may be preferred to MODE=GROUPS
  - Process Mode should be carefully specified
  - CYCLE and Interval Time should be carefully specified

Multi-Image Manager- Allocation (MIM MIA)

- Provides Tape Device Sharing among many z/OS, z/VM, and z/Linux Images
- May Use XCF for Messaging among Sysplex Members
- May Use Coupling Facility for Structure Messaging among Sysplex Members
- May Use CTCs for Messaging among Sysplexes or Virtual Guests
- May Use Data Set on Shared DASD for Control File
- MIM MIA needs to have a High Dispatch Priority
- May Cause Performance Challenges if NOT properly configured
  - MODE=DEMAND may be preferred to MODE=GROUPS
  - CYCLE and Interval Time should be carefully specified

Basic Sysplex

- Used for Managing Shared Resources
- Uses XCF for Communication
  - May Use Coupling Facility for Connectivity
  - May Use CTCs for Connectivity
  - Best Configuration uses both Communication Methods
    - XCF determines which Connectivity Method is achieving the Best Performance

Parallel Sysplex

- Used for Managing Shared Data Resources
- Requires a Coupling Facility and Links
- Uses Structures for Communication and Shared Data Memory
- XCF may use Structures in a Coupling Facility for Communications

Geographically Dispersed Parallel Sysplex (GDPS)

- Same as a Parallel Sysplex except spans Physical Sites using ESCON or FICON Connections
- May Experience Performance Challenges based on Distance between the Data Centers
- FICON Protocol Performance Much Better

Extended Remote Copy (XRC)

- XRC is used for Mirroring Data on IBM DASD and Cached Controllers
- Control Units at a Local Site have special Licensed Internal Code (LIC) to support XRC functions
- A System Data Mover (SDM) runs in an Image at a Remote Site that “reaches into the local Control Unit” to Copy Data to Remote DASD
- The Local Control Unit may go into “Slow Down” Mode if SDM cannot Keep Up
- Image Performance can be seriously Impacted by XRC

Data Sharing Structures

- Data Structures may be used to Share Application Data in a Parallel Sysplex Environment
- Consider Placement of these Structures for Optimal Performance in Multiple CFs
- Consider Correct Sizing (auto sizing) for the Structures
- CICS, IMS, DB2, and RRS are primary Users of Sharing Structures
  - ➢Use Lock, List, and Cache Structures

z/VM and Virtual Guests

- A Large Number of Guest Operating Systems may be Run in a z/VM Environment
- Sysplexes may be Emulated
- Coupling Facilities may be Emulated
- z/OS Guests may experience Performance challenges if NOT setup to run Under z/VM
- z/OS Guests better suited for Non-Production Environments

z/Linux

- Many Linux Operating Systems may be Setup to Run Under z/VM
- May be established to connect to z/OS Environments through Hipersockets
  - A OSA-like Communication Connection with NO Physical I/O (Storage-to-Storage)
- z/OS Images may experience Performance Challenges when connected to improperly tuned Virtual Guests

JOB Entry Subsystems

- Simply put, JES brings Work (Jobs) into the Environment, Controls Job Flow, Schedules Jobs for Execution, Creates Output, Prints/Punches Output, Transmits Data from one Location to Another
- By Segregation Functions, there is a higher degree of Management, Automation, and Multi-processing
- 2 Subsystems available
  - JES2 (HASP)
  - JES3 (ASP)
- Both Contain:
  - SPOOL
  - Checkpoint
  - Communication vehicles
  - Console Support
  - Device Support

JES2 vs. JES3

- In a single System (Monoplex), mostly identical
- JES2 provides Independent Control for JOB processing
  - Each JES2 Image controls its own Input, Scheduling and Output Processing
- JES3 provides Centralized Control for Job Processing through a single JES3 Global Processor
  - The Global provides Job Selection, Scheduling, Device Allocation for all JES3 Local Processors
  - JES3 provides increased control for Job Scheduling, Dealing Scheduling Capabilities, Device Allocation, and Job Flow

JES2 Multi-Access Spool (MASPlex)

- Multiple z/OS Images can Share the Same Spool
- Currently, up to 32
  - ➢ The Practicality needs to be heavily considered
- Multiple (secondary) JES2 Environments may exist within the Same Image
- JES2 and JES3 can exist in the same Sysplex or Image
- Many Sites have several MASPlexes within the same Sysplex
- Many sites use this process to Provide Security to a Complex Environment

Key JES2 MASDEF Specifications

Setting the Hold and Dormancy Times is site dependent. Most Times the Ranges vary and are acceptable. Often the Values for Each Member of the MASPlex are different. The following recommendations may be used for Cached DASD and Coupling Facilities:

- **HOLD TIME: should be between 20 and 50 (specified in hundredths of seconds). Less than 20 (.2 seconds) may cause excessive Overhead of Reading and Writing the Checkpoint, with Little Useful Time for Exclusive Control. Most Requests for the Checkpoint are clustered in Periods of less than 0.2 seconds.**
  - **Job Tracking/Scheduling and Output Retrieval/Archival Subsystems will benefit from Long Hold Times (up to a full second). Times over 1 second may lock out other Members, and be counter-productive (break point).**

Key JES2 MAS Specifications

- **MINIMUM DORMANCY TIME: Most requests for the Checkpoint can afford to wait for 3 or more Seconds without noticeable Degradation. This applies to heavy Batch, TSO, NJE, RJE, or Print Workloads.**
  - **Some Job Scheduling Subsystems and Output Retrieval Subsystems benefit from more Frequent Access if they are managing Large Queues of JES2 Work, such as submitting many Jobs, Status Commands, or PSO requests. These Subsystems tend to get behind if they don't have More Frequent Access and Longer Hold Times.**
- **MAXIMUM DORMANCY TIME: The default is 500 (5 Seconds) which is fine for most members. Anything less is appropriate for relatively JES2-Idle**

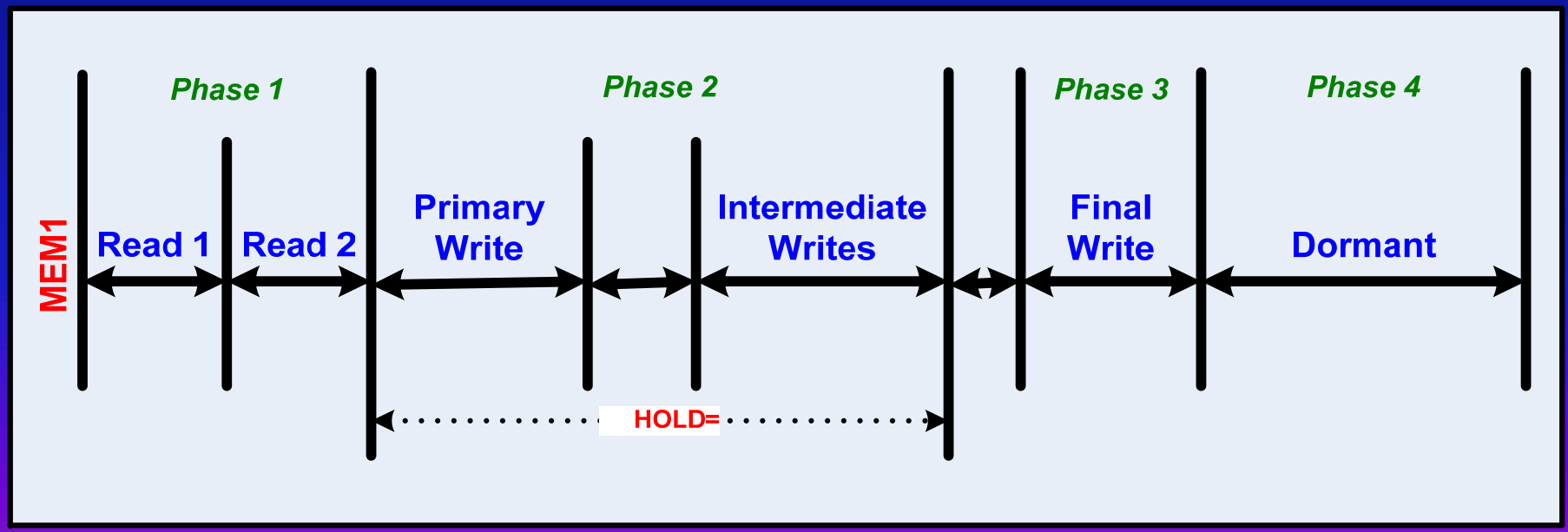
JES2 MAS Performance Tools

The Best Measurement Tool for the JES2 Checkpoint is the lack of Symptoms or JES2 Delays by the application.

- **SDSF/Sysview MAS Panel**
- **RMF Monitor III**
- **RMF CF Structure Activity Reports**
- **$D PERFDATA(QSUSE)**
- **$TRACE(17) Data**

JES2 Checkpoint Cycle

- HOLD: Time the MAS imaged has control of the Checkpoint (Reserved or Locked)
- MINDORM: Time to Wait for the Next HOLD
- MAXDORM: Time that Inactive Members look for their Work from Other Members



JES2 MAS Tuning Notes

- Avoid Cycle Times that Create “Hot Potato” effect
- General Recommendation for Multiple Member MASPlex

System Workload	2 Members	3 Members	4 Members	5 or More Members
Batch, NJE, RJE, TSO, Print	HOLD=50 DORM=(50,500)	HOLD=40 DORM=(80,500)	HOLD=30 DORM=(90,500)	HOLD=20 DORM=(100,500)
Heavy SSI Usage	HOLD=80 DORM=(20,500)	HOLD=80 DORM=(20,500)	HOLD=80 DORM=(20,500)	HOLD=80 DORM=(20,500)
Little JES2 Activity	HOLD=30 DORM=(80,500)	HOLD=20 DORM=(100,500)	HOLD=20 DORM=(100,500)	HOLD=20 DORM=(100,500)

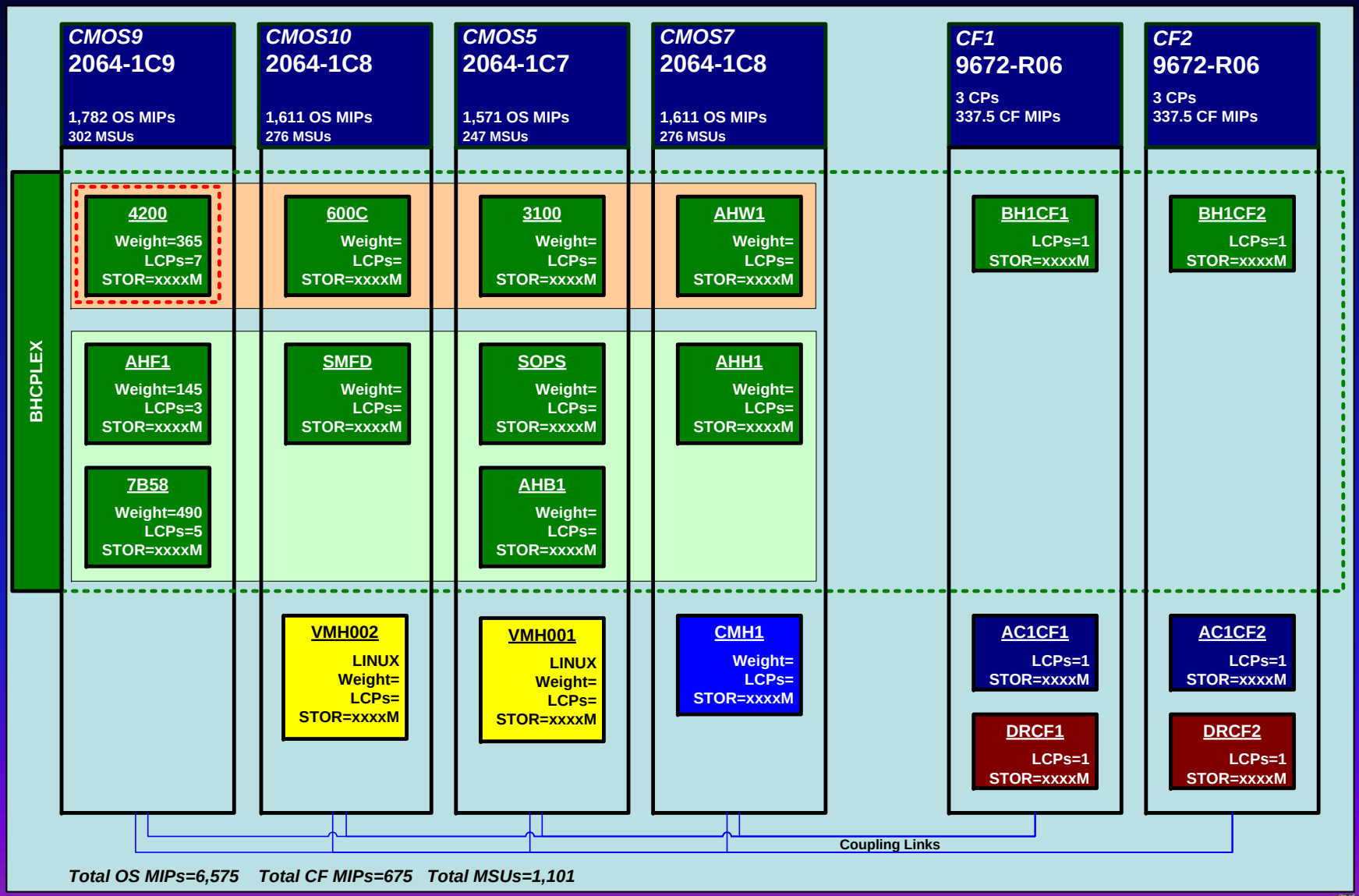
JES2 MAS Tuning Notes

- MASDEF Parms may be different for each MAS member
- Checkpoint Configuration may be different
- For Complex MASPlexes, a specification of DORMANCY=(100,300) has worked well
- For Complex MASPlexes, a specification of HOLD=50 has worked well
- SPOOLDEF definitions, such as TRKCELL, BLKSIZE should be considered
- ALL JES2 Parms should be evaluated for currency

JES3

- Completely different management than JES2
- Far less JES3 sites than JES2
  - Typically very large and complex Environments
- CTCs, XCF, and Coupling Facilities must be properly tuned
- Some sites run JES3 in JES2 mode for the Scheduling Capabilities
- NO MAS considerations
- Spool and Checkpoint may have Tuning Opportunities

Topology Example



System Environmental Topologies

- In order to Understand the Capacity Available to a Particular System Image, we must determine:
  - What Focal System(s) are We Examining
  - What CPCs are Involved
  - What LPARs are Involved
  - What Coupling Facilities are Involved
  - What LPARs on the Coupling Facility are Involved
  - What non-z/OS LPARs are Involved
  - What Kind of Images are Involved

Production Systems

- We will consider any Sysplex, z/OS Images, Coupling Facility LPARs, non-z/OS LPARs to be considered Production, if:
  - The Work is considered Mission-Critical (Bread and Butter Applications)
  - The Work supports the Mission-Critical Environment (Batch for the Onlines, D/R)
  - The Work supports Hardware for the Mission-Critical Environment (Printers, Check Sorters)

Non-Production Systems

- Anything Other than Product Systems
  - Q/A
  - Development
  - Test/Sandbox

Pecking Order

- Production is Number 1
  - Be Sure Environment is Setup this way
- Non-Production may vary but should NOT exceed Production
  - Test/Sandbox may be Number 2
  - Q/A maybe Number 3
  - Development may be Number 4

Throughout the entire Analysis, We want to be sure that every Pecking Order matches the Business Importance.

Subsystem Topologies

- In most Complex Environments, there are many components that make up the Subsystems
  - CICS may have one Application Region
  - CICS may have many Application Regions with Support Regions
  - IMS may have many Application Regions (MPRs) with many Support Regions
  - DB2 may have one Application Regions (DBM) with many Support Regions
  - Websphere/MQ may have one Application Regions (CHIN) with other Support Regions

Transaction Subsystems

- CICS Transaction Server can run with as little as 1 Application Owning Region (AOR)
  - Typically there will be many regions aligned by Application or Availability
  - CICSPlex/SM will add additional Functionality and more Regions
  - Regions may include:
    - CPSM
    - CPSM Coordinating Address Space
    - CICS TS Gateway (from WAS)
    - Terminal Owning Region(s)
    - File Owning Region(s) or SMSVSAM
    - Temp Storage Regions
    - Shared Data in Coupling Facility
    - Queue Owing Regions
    - Resource Owning Regions
    - Printer Owning Regions
    - Application Owning Regions

Transaction Subsystems

- IMS Transaction Manager can run in 2 Modes
  - IMS DB/DC (with Terminal Support)
  - IMS DB/CTL (Data Base portion only)
  - Typically there will be many regions aligned by Application or Availability
  - Regions may include:
    - Lock Manager (IRLM)
    - Control Region (CNTL)
    - Message Processing Regions (MPRs, could be hundreds)
    - DLISAS

Database Management Subsystems

- DB2 Data Base Manager can run in 2 Modes (Dependent, Independent Remote Support)
  - DB2 Calls may come from Batch, TSO, CICS, IMS
  - Remote DB2 Calls may come from Servers or Desktops (Gateways or MS Office Access or Excel)
  - Typically there will be many regions aligned by Application or Availability
  - Regions may include:
    - Lock Manager (IRLM)
    - Master Region (MSTR)
    - Data Base Manager (DBM1)
    - Stored Procedure Address Spaces (SPAS)
    - Workload Manager Application Server Environments (WLMx)
    - Remote Distributed Gateway (DIST)

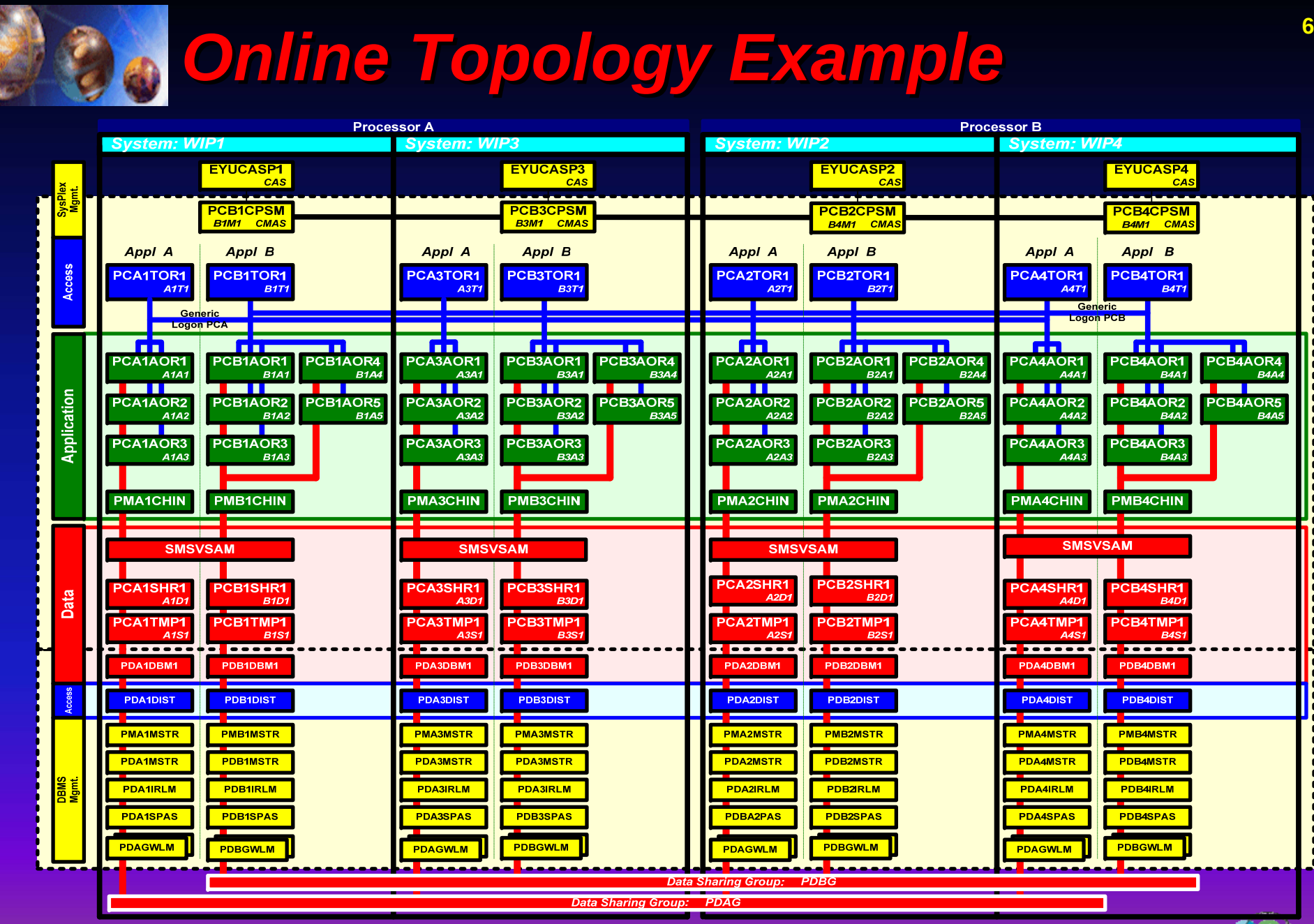
Websphere MQ Subsystems

- MQ Series can run with Mainframe-to-Mainframe Connections or Mainframe-to-Distributed Connections
- Many Subsystems are run Based on Application or Availability
  - MQ Requests Calls may come from Batch, TSO, CICS, IMS
  - Regions include:
    - Channel Manager (CHIN)
    - Master Region (MSTR)

Applications Topologies

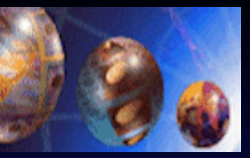
- Determining How Applications Interact with Each Other can be Challenging
- It is essential that the correct Pecking Order among All the Different Components and Subsystems is maintained
- Many Applications may be supported by a Single Set of Subsystems
- Or, more typically, Several Sets of Subsystems, which may Have their own Pecking Order
  - High-Priority
  - Medium Priority
  - Low Priority

Online Topology Example



UNIX System Services (USS)

- USS is no longer an Option in the Operating Environment
- Many Critical Subsystems Run under USS
  - TCP/IP
  - FTP
  - Websphere Application Server
- All Work inside of USS is NOT Interactive by nature
- The correct Pecking Order for USS work is Essential
- The Operating System Components Need to Run Like those of z/OS
- Initiating Forked and Spawned Work Needs to Run Like those of JES2
- Long Running Daemons may need to run like High Priority Transaction Systems or like Other High-Priority Support Processes



Websphere Application Server Subsystems

- Many Sites are now Running Production Websphere Application Servers
- Some or All of the Components may be z/OS based
- Some of the Components may be run Distributed and Connect into z/OS based processed
- Many Web types of Transactions may Connect to Legacy Applications (CICS, IMS, DB2)
- A HTTP Server may be running on z/OS
- Security Pieces (like LDAP) will be used, which runs under USS
- The correct Pecking Order for IWEB, CB, USS work is Essential

z/Technologies

- Current and Future z/Technologies are evolving at a Rapid Pace
- Typically New Hardware Functionality Appears Every 18-24 Months
- New Operating System Software Appears Every 12 Months
- There are now 3 z/Operating Systems
  - z/OS
  - z/VM
  - z/VSE
- Linux for the Mainframe Environment is now a Player
- Virtualization Engines Will Begin to Appear in September
- Virtualization Engines for Storage Devices are Also Appearing

z/Series Processors

- z/890 is the New Mainframe on the Block
  - Up to 4 OS Processors
  - Speeds Range from 365 to 212 to 171 to 108 to 90 to 45 to 27 MIPS
  - Processor is the Same Speed; Microcode Slows Down the MIPS
  - ICFs, IFLs and zAAPs Available
- z/990 is Currently the Larges and Fastest
  - Currently Up to 24 Processors (soon 32)
  - Each Engine runs at 450 MIPS (less the MP effect)
  - ICFs, IFLs and zAAPs Available

z/Series Processors

- z/990 is Currently the Larges and Fastest
  - Next Version of z/990 will run at about 700 MIPS
  - Future Version of z/990 will run at about 1,200 MIPS with up to 255 CPs in a single Footprint
- These z/Series Processors will Support the 64-Bit Architecture
  - Allowed Us to Break the 2Gigabyte Real and Virtual Storage Limit
  - We Can Now Get into the Exobyte Range (a HUGE Number)

z/OS 1.6 and Beyond

- z/OS 1.6 is a Pivotal z/OS Release
- It is a Architectural Level Set (ALS) Release
  - z/OS 1.6 only Runs on z/Series Hardware
  - Does Not Run on 9672 CMOS or Multiprise Processors
- z/OS 1.7 Has Major Enhancements in Several Key Components
  - TCP/IP
  - JES2
  - SMS
  - HSM
- z/OS 1.7 Introduces the Virtualization Engine, which Includes Enterprise Workload Manager (EWLM)

Workload Manager

- What is Workload Manager?
  - Workload Manager is part of the System Resource Manager (SRM)
  - 2 Mode were Available from ESA 5.2.1- Compatibility and Goal
  - Compatibility has not been support since z/OS 1.3
  - The Facility for Categorizing Work by Business Importance and Assign Priorities (the Pecking Order)

What's Bad About Compatibility Mode?

- Too Complex, Labor Intensive, Time Consuming
- Incorrect Specifications would Cause Unexpected SRM Controls
- Managed at a Task Level
- Maximizes System Efficiencies based on SRM Specifications, not necessarily What is Required
- Tuned for Peak Periods
- Tuned for a Superset of all Environments (or must maintain many individual definitions)
- No Recognition of Subsystems and Enclaves
- Less Flexible as Systems becomes More or Less Busy

What's Good About Goal Mode?

- Sysplex Managed and Workload Balanced among all the Members of the Sysplex
- Goal Oriented for Strategic Work
- Business Unit related
- Subsystems and Enclaves are Manageable
- Resource Group Management
- More Flexible as Systems become More or Less Busy
- Dynamic I/O Intensive Workload Management (native Shark)
- Intelligent Resource Director

Basic Concepts and Management

- The Service Policy contains all the Components for Managing the Workloads
- Workloads are Defined for Reporting Purposes
- Service Definition Coefficients are Defined to Establish How Service Units are Accumulated
- Service Classes are Defined to specify an Importance, a Goal, and possible a Duration
- Subsystem Qualifications are Defined to Assign Work to a Service Class
- Report Classes may be Defined
- Other Functions may be Defined that Exploit WLM
  - Application Environments
  - Scheduling Environments

Effective Batch Workload Management

- JES2 has 36 available (A-Z, 0-9)
- JES3 has 255 available (A-Z, 0-9, assigned Class Names)
- Less is More!
- WLM does better with fewer Service Classes with similar work in it
- Segregate Jobs into JobClasses based on Resource Requirements
- It may be Time for New Standards or a JobClass Overhaul
  - Are all 36 JobClasses in Use?
  - Are they different on each System? Sysplex?
  - Are there single-purpose Legacy JobClasses for a single group of Users?
  - Are there many similar JobClasses with “one-off” characteristics?
  - Are there Jobs that are better suited as Started Tasks?
- Define a set of JobClasses for JES-managed Initiators
- Define a set of JobClasses for WLM-managed Initiators
- Define a set of JobClasses for Operator Intervention
- You will probably need JobClasses based on Production vs. Ad-Hoc vs. Systems Support vs. Development

Suggested JobClass Categories

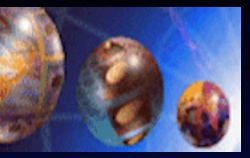
More Resources, Less Importance, Less Priority, Less Aggressive Goal

Non-Restricted JobClasses

Utility Jobs <30 sec, No Tapes *<=8MB Storage* *<=25,000 EXCPs*	Non-Production Jobs <=2 min, No Tapes *<=16MB Storage* *<=100,000 EXCPs*	Non-Production Jobs <=5 min, No Tapes *<=32MB Storage* *<=250,000 EXCPs*	Non-Production Jobs >5 min, No Tapes *>32MB Storage* *>250,000 EXCPs*
	Non-Production Jobs <=2 min, 1 Tape *<=32MB Storage* *<=100,000 EXCPs*	Non-Production Jobs <=5 min, 2 Tapes *<=32MB Storage* *<=250,000 EXCPs*	Non-Production Jobs >5 min, >2 Tapes *>32MB Storage* *>250,000 EXCPs*

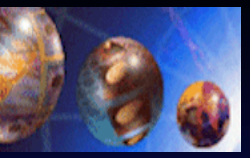
WLM-Managed Initiators

- JES2/JES3 Initiators for WLM-Managed Job Classes are dynamically controlled by WLM
- WLM adjusts the number of WLM-managed Initiators based on:
  - Jobs queued in the WLM-managed Job Class
  - Performance Goals and Importance of a Job
  - How well Goals are being met (PI)
  - Available Capacity on the System



What's Bad About WLM-Managed Initiators?

- WLM does NOT "Workload Balance" among the members of the MAS or JESPLEX (although JES3 does have its own Workload Balancing mechanisms)
- WLM does NOT know about the number of Tape Mount Requirements or the Type of Tape Mount for all Jobs
- Queuing Delays while waiting for an Initiator affects Response Time Calculations and Performance Index (PI)
- All Jobs within a defined Service Class should be managed by the same type of Initiator (MODE=JES or MODE=WLM)
- z/OS 1.6 and Higher is getting Better at "Attempting" a Balance



Effective Interactive Workload Management (TSO, USS)

- Interactive Workloads tend to have a large Number of Users Doing Many Different Things
- The Work tends to be very dissimilar
- Production vs. non-Production Work
- TSO and USS are similar in the Interactive Part, USS is different in the Long-running Daemons and System-like Work
- Several Periods of Work are often Defined to allow Quick Work to get Through Faster
  - For 3 Periods, 90% of the Work should complete in the 1st Period, 5% in the 2nd Period, and 5% in the 3rd Period (I prefer 2 Periods)
- Several Service Classes may be Defined for General Users vs. System Programmers vs. Operations

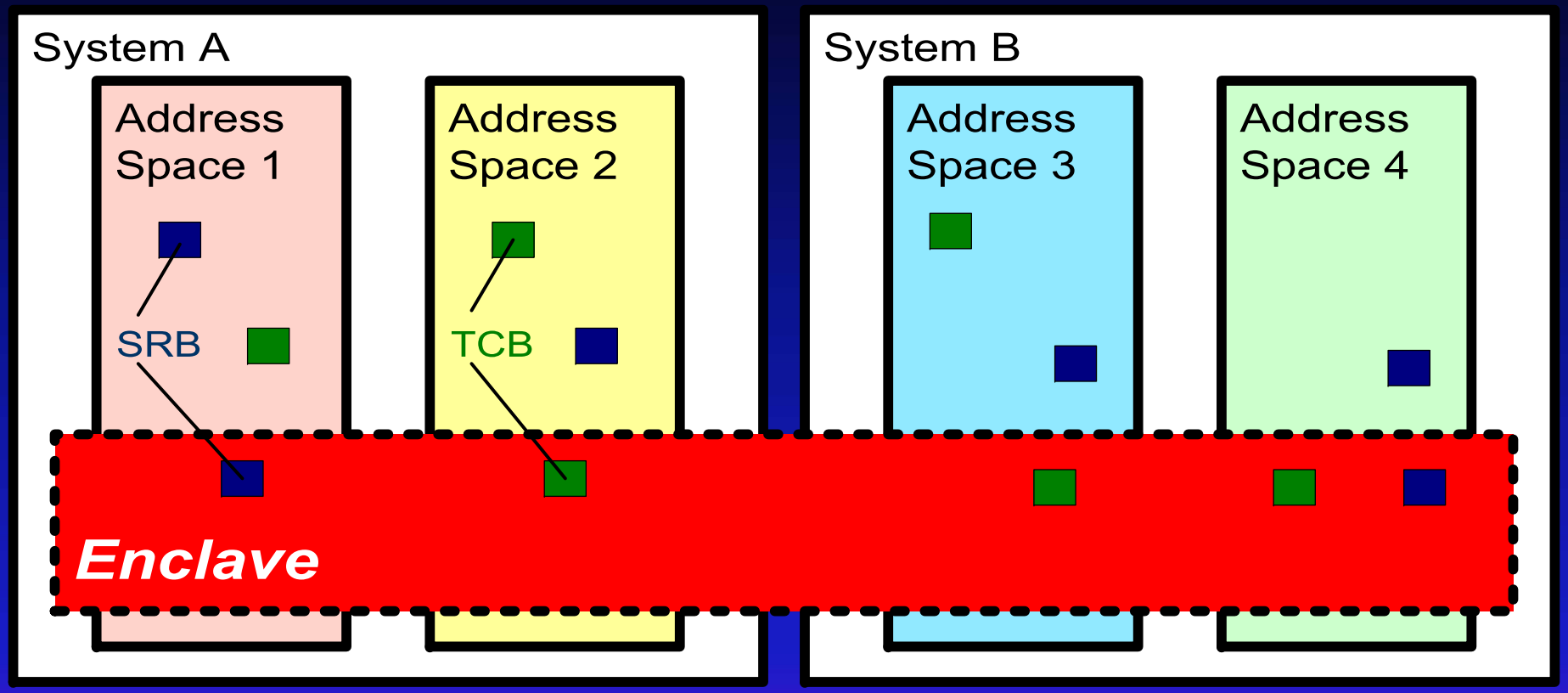
Effective Online Transaction Workload Management

- Transaction Workloads may be Managed by Region Goals or by Transactions Goals
- Many Levels of Applications may co-exist in the Same System
- Many Levels of Transactions may co-exist in the Same Region
- Transaction Regions have the largest Number of Users Doing Many Different Things
- The Work tends to be very dissimilar
- Production vs. non-Production Work
- Percentile Goals are Often Defined for Transactions
  - 95% < 2 seconds
- Transaction Workloads are Complex to Define and Manage because of all the Possible Moving Parts (CICS- MRO, Management Regions, etc.; IMS- Control Regions, MPRs, etc.; Connectivity to DB2)
- Several Service Classes may be Defined for Unique Levels of Applications

Effective Database Manager Workload Management

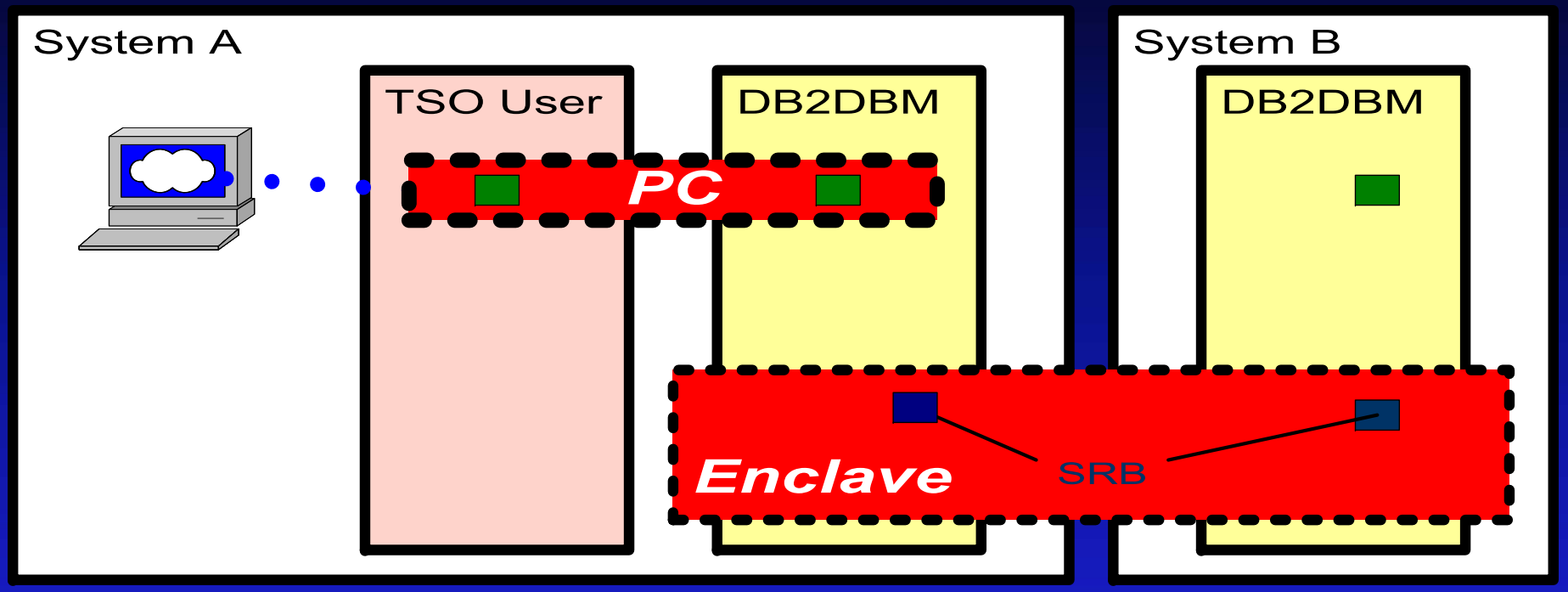
- DB2 Workloads that Originate within the Same System)Dependent) are relatively Easy to Manage- NO DEFINITION IS NECESSARY
  - DB2 Transactions/Threads (Dependent Enclaves) Inherit the Attributes of the Calling Environment (CICS/IMS, Batch, TSO)
- DB2 Components Need to be Properly Defined (IRLM, MSTR, DIST, WLM)
- Non-DB2 Regions Must be Defined According to the Vendor Recommendations
  - All are Different
  - Datacom, IDMS, SAP, Oracle, Oracle Gateways
- Many DB2 Subsystems will Exist in Large Complex Environments
  - Should be Configured based on Availability Needs, or Special Connections to Other Components
- Several Service Classes may be Defined for Unique Levels of Applications

Anatomy of the Enclave



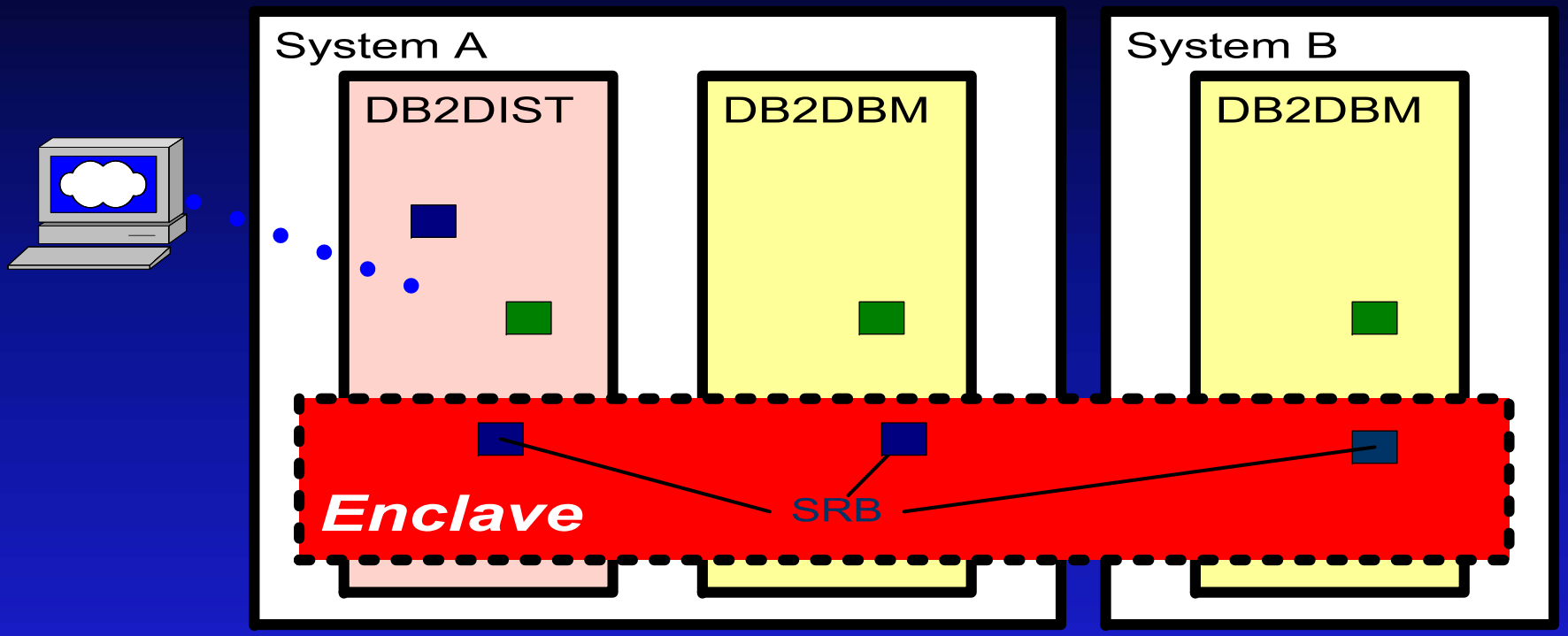
- 2 Types of Enclaves
  - Dependent- a continuation of a transaction already executing
  - Independent- a new standalone transaction

Anatomy of the Dependent Enclave



- TSO User issues a Database Call (PC) to DB2DBM
- DB2DBM creates the Dependent Enclave (to execute a SQL Query)
- TSO User Classification (and related Service Attributes) continues with the Enclave

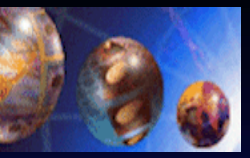
Anatomy of the Independent Enclave



- DB2DIST recognizes DDF Request (Network) from Outside the System
- Creates the Independent Enclave
- Schedules Enclave SRB targeted to DB2DBM
- Needs to be Classified under DDF Subsystem (otherwise assigned to SYSOTHER)

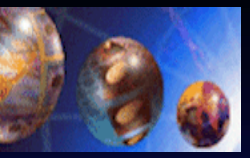
Effective Started Task Workload Management

- Started Tasks (STCs) fall in Several Categories of Workloads and Importance
  - Operating System
  - System Support
  - Subsystem Support
  - Network Management
  - Operations Management
  - Performance Management
  - Data Management
- High Dispatch Priority System Tasks assigned to SYSTEM Service Class
- Privileged Dispatch Priority System Tasks assigned to SYSSTC Service Class
- Subsystem Classification only needed if assigning to Report Class
- Subsystem Parameter (SPM) may be used as a failsafe from inadvertently assigning System Tasks in other parts of Subsystem Classifications Rules



Understanding Enclaves and Application Environments

- What is an Enclave?
  - Simply put, an Enclave is a Construct that represents a Business Unit of Work
  - It is an Anchor used for Resource Accumulation by a Transaction, no matter where it executes
  - It appears as the End-user's perception of a Transaction
  - It can traverse many Server Address Spaces (even across Members of a Sysplex), and is managed Independently
  - It is classified by the creating Subsystem, and can be assigned to a Service Class and/or a Report Class

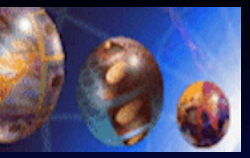


Understanding Enclaves and Application Environments

- What is an Application Environment?
  - Simply put, a Collection of Server Regions that are Started and Stopped by WLM
  - The Number of Servers can be Managed by the Definition of the APPLENV
  - Each APPLENV may Service any Number of Service Classes
  - Each Server will Serve only 1 Service Class for Each APPLENV
  - APPLENVs are Currently Employed by DB2, and Websphere Application Servers
  - There May Be a Large Number of APPLENV Servers Running at Any Given Moment

EWLM

- Enterprise Workload Manager
- Extends Mainframe Concepts to Other Parts of e-Business
- Improves Utilization of IT Resources
- Manages Business Process Service Levels
- Part of IBM's Virtualization Services
- Discovers Business Transaction Topologies
- Connects with Other Autonomic and Self-Healing Solutions
- Provides Business-Oriented Reporting



Anatomy of a Performance Health Check

- Physical Processor Configuration
- Logical Processor Configuration
- Processor Utilization
- Channel Utilization
- Paging
- Storage
- Coupling Facilities
- XCF
- Running Tasks
- In and Ready Queue
- Workloads
- Top Resource Consumers
- Recommendations

Leaving No Stones Unturned

- The Real Challenge is Analyzing All the Possible Moving Parts to See if They Are Running as Expected
  - Pecking Order Correct?
  - All Support Components Optimized?
  - All Hardware Optimized
  - I/O Avoidance In Place?
  - Any Obvious PTFs NOT “Applied”?
  - What’s Left Over is the Obvious Opportunity

Overhauling PARMLIB Specifications

- PARMLIB Members Define the Many Definitions and "Knobs" for Running z/OS
  - Since OS/390, a PARMLIB Concatenation has been Available
  - Every Site will have a SYS1.PARMLIB
  - My z/Approach to Setting Up the Operating Environment has Several PARMLIBs
    - Site Level
    - Sysplex Level
    - System Level
    - IBM's Original 2 PARMLIBs
  - Most Sites Have Not Updated PARMLIBs as They moved from Earlier OS Versions
  - Many Specifications have Changed or are Obsolete

Overhauling PARMLIB Specifications

- IEASYSxx Defines Which Other Members to Include
- COUPLExx Defines the Sysplex and XCF specifications
- IEAOPTxx Defines Many of the Performance Options
- GRSCNFxx and GRSRNLxx Define GRS Options
- CSVLLAxx and COFVLFxx Define LLA and VLF Options
- CONSOLxx and CNGRPxx Define Console Devices
- DIAGxx Define Common Storage Tracing
- ERBRMFxx Define RMF Options
- SMFPRMxx Define SMF Options

- Also Need to Look in Other Independent PARMLIBs
  - JES2, JES3
  - Multi-Image Manager

Getting to the Data

- It Would Be Nice if All We Had to do Was Just "Get" or "Put" Data
- Need to Understand What The Data is
  - Sequential Data
  - VSAM Data (Linear)
  - Media Manager Data (DB2)
- Need to Understand All the Parts In Between Your "Get" and "Put" and the Actual Data

Current Technologies

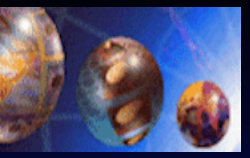
- It Would Be Nice if All We Had to do Was Just “Get” or “Put” Data
- Need to Understand What The Data is
  - Sequential Data
  - VSAM Data (Linear)
  - Media Manager Data (DB2)
- Need to Understand All the Parts In Between Your “Get” and “Put” and the Actual Data
  - ESCON or FICON Channels
  - Directors or Switches
  - Cache Control Units, DASD
  - Control Units, Tape, TP, UR

Current Technologies

- Most Devices “Emulate” 3380s or 3390s
- No Real Ones Still Used
- Sharks Use SCSI Drives with Large Cache in Front to Spread the Data Out to Unused Memory Segments
- Symetrics Use SCSI devices in a Drawer with Large Cache in Front
- Caching Needs to be Active at the Subsystem Level
- Caching Needs to be Active at the Volume Level
- Deferred Fast Write (DFW) needs to be Active
- Sufficient Channels are Needed for Performance and Availability
- Data Mirroring Overhead Must be Considered
- GDPS Overhead Must be Considered

Understanding and Optimizing Catalogues

- Today's Dataset Catalogues are Integrated Catalogue Facility (ICF)
- There is 1 Master Catalogue with Many User Catalogues Connected
- In Modern Data Centers, the Master Catalogue is Shared Among All the Members of a Sysplex
- User Catalogues are typically Setup the Same Way
- By Sharing, there is Coordination between the Catalogue Address Spaces (CAS) on all Images
- XCF is used for this Coordination
  - Optimizing XCF is now Even More Important
- There will be Enqueues on Various Catalogue Components
  - SYSIGGV2
  - SYSZVVDS
- Catalogue Placement is crucial
- Placement of High Level Qualifiers (HLQs) is crucial
- Catalogue Management can become a Bottleneck (Backups)



Understanding and Optimizing VSAM Data

- VSAM Components are still Important
  - Segregate Index from Data
  - Spread Data across many Volumes
- Buffering is the MOST Important Performance Tool for VSAM Data
  - Index and Data have different Requirements
- Re-organizing the Data is VERY Important
  - Eliminate the Challenge of CI and CA Splits
- CA-Hyperbuff and IBM dfSMS managed Buffering can help Eliminate Buffering Challenges

Understanding and Optimizing Sequential Data

- Flat Files Benefit from Defragging and Buffering
- PDS and PDSEs can still be Challenging
  - Data Management must still me Routinely executed
    - Compressing PDS
    - Defragging PDSEs
  - Buffering is Still Important
    - Default for PDS is 2 Buffers
    - Improved Performance is Typical when BUFNO=20 or 32 is specified
    - Buffering for PDSEs is much more Complex
    - SMS ACS Rules Need to be Defined
    - A Hyperspace for the SMSPDSE Address Spaces needs to be adjusted from the Default

Understanding I/O Avoidance

- Remember: “The Best I/O is No I/O”
- Data in Storage Techniques is beneficial
  - Linear Data Sets
  - Hiperspaces and Dataspaces
  - VIO for Sequential Data
  - LLA and VLF for Libraries
  - CA-PMO and CA-PDSMAN, or IBM’s dfSMS PDSE
- Buffering to Process More Blocks of Data with 1 I/O
- DFW to Reduce Hardware Overhead
- Using Appropriate Access Methods to Reduce Hardware Overhead
  - BSAM and BDAM can cause Performance Challenges

Collecting Performance Data

- z/OS, Sub-Systems, and Applications generate Performance Data, Traces, and Logs
- Be Sure the Appropriate Data is Being Collected and Archived
- Data Should Be Obtained from an Agreed Upon Peek Period (or Periods)
- Data Should Be Collected from All Related Systems for the Same Periods
- Data Intervals Should Be the Same

Resource Measurement Facility (RMF)

- 3 Different Components are Available
  - Monitor I- Primary Data Collection
  - Monitor II- Online, Real Time Monitor
  - Monitor III- Historical Collection
- Some Sites May Use BMC's CMF Product (about 95% compatible)
- Monitor I Stores its Data into the SMF Files (Type 70-79)
- Post Processing Reports and Data Extraction Provides several Canned Reports And Data Fields That Can be Used In Excel Spreadsheets

System Management Facility (SMF)

- Most Sites Collect Vast Amounts of SMF Data
- All Major Components, Subsystems, Transaction Regions, and Database Regions Produce SMF Data
- Complex Archival Procedures are Typical
  - Certain Data is kept Short Term
  - Certain Data is kept Long Term
  - Certain Data is kept Forever
- Accounting and Monitor Data is Most Useful
- SMF Data is Input into CA-JARS, CA-NEUMICS, RMF Post Processor, MXG

MIM 189 Records

- Multi-Image Manager MII and MIA Can Product SMF Records (typically 189)
- For MIA, Contains Mount Times, and Device Usage Data
- For MII, Contention Data is Generated over Each Interval
- Canned Reports Available for Standard Reports

GRS Enqueue Trace

- GRS Events May Be Generated Into a Trace Format By Starting the Trace Started Task
- Among Other Items, the Time a Reserve or Enqueue is Started; then Ended.
  - This allows Examination to See If The ENQ/DEQ Process Timings are Reasonable
- There is a Filter Available to Reduce the Amount of Data Collected
  - The Filter Exit is Needed to Collect Data About SYSZTIOT
- Run This Trace Sparingly, There is Some Overhead to Look At Every ENQ/DEQ Event
- To Get a Complete Picture, Run the Started Task On All Systems in the Sysplex that is Being Examined

Generating Recommendations

- Every Health Check or Assessment Needs a Written Document or Presentation Documenting:
  - What Was Analyzed
  - How the Data Was Collected
  - When the Data Was Collected
  - Difficulties Obtaining the Data
  - Findings
  - Corrective Actions
  - Future Processes

Revisiting the Process

- Every Health Check or Assessment needs to be revisited:
  - Periodically at Regular Intervals
  - When Major Events in the Environment Occur
    - Hardware Upgrades
    - Software Upgrades
    - New Major Applications
    - Application Upgrades
    - New Ancillary Processes are Included (Distributed Processors, zAAPs, etc.)
  - When Performance Challenges are Perceived (preferably before)

Summary

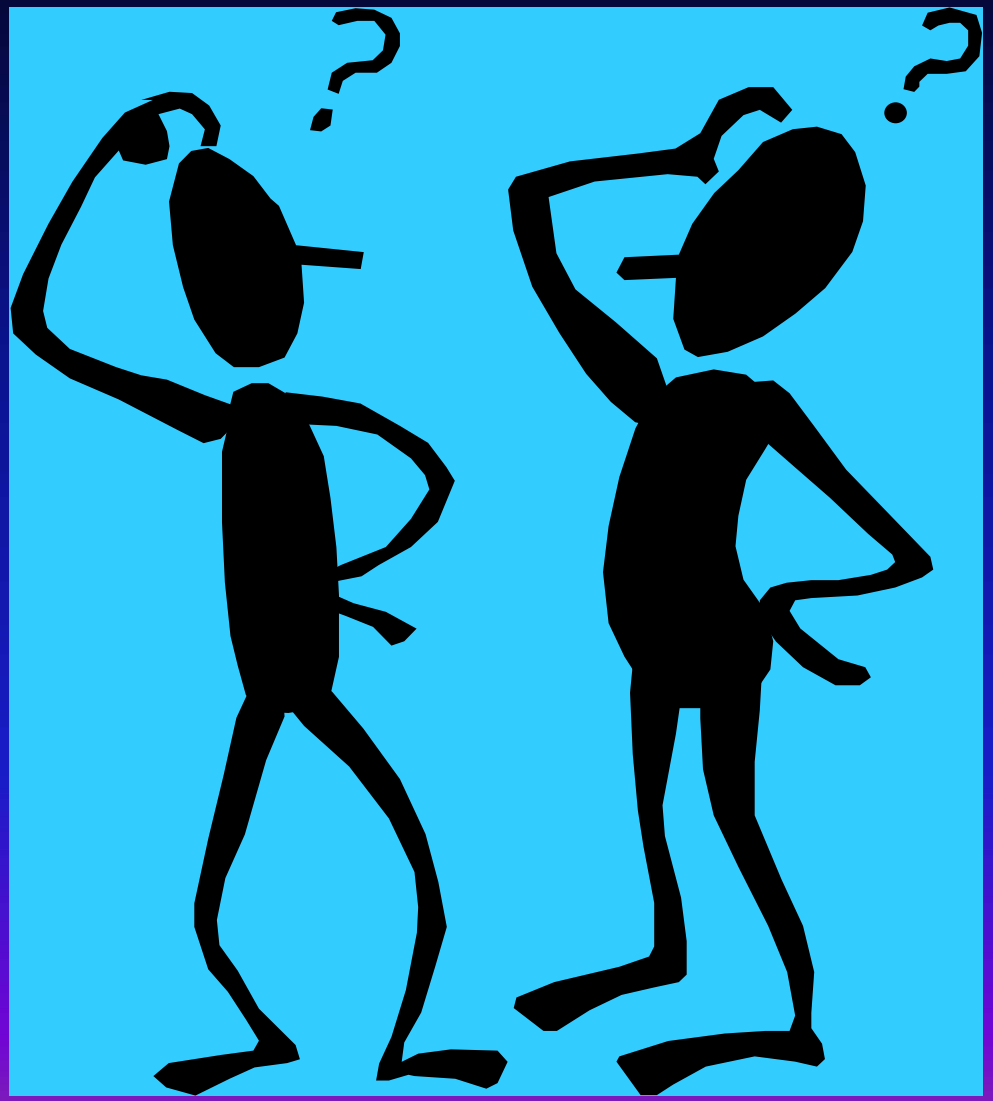
- We've Looked At a Lot of Different Areas That Can Impact Performance
- It is NOT a Simple Matter with a Simple Solution
- The Solutions are
  - ➢Low Cost, Low Risk, High Impact

    All the way to
  - ➢High Cost, High Risk, Low Impact
- We've Looked at Mostly Environmental Issues Surrounding Specific Applications
- The Specific Applications Do Have Plenty of Opportunities

Conclusion

- And Now a Word from Our Sponsor
  - Discovering Mainframe Performance Opportunities is a Complex Process
  - When You Find the Right Opportunity, It Can Be Very Rewarding
  - The Process Leads to Enhanced Relationships with Our Customers
  - A Successful Process Can Lead to New Solutions Opportunities (either on the Mainframe or Other)
- I am Available to Help With these Opportunities

Questions?



THANK YOU