Intelligent Resource Director

WLM Lpar Cluster Management

Robert Vaupel
Schönaicherstr. 220
D-71032 Böblingen
Phone: +49(0)-7031-16-4239
email: vaupel@de.ibm.com
eServer WLM Development

Content

- Background
- LPAR Cluster definition
- Overview of functions
  - I/O Management
    - Dynamic Channel Path Management
    - Channel Subsystem Priority Queueing
  - LPAR CPU Management
    - LPAR Weight Management
    - LPAR Vary CPU Management

IBM @server. For the next generation of e-business.

© IBM Corporation 2001

Background: Goal Oriented Resource Management

- Management within an OS/390 image:
  - OS/390 is unrivaled in its ability to manage diverse, competing workloads in a single OS/390 image
  - With WLM Goal Mode resource management is based on a statement of desired results (goals) rather than low level resource based controls
- Management across OS/390 images
  - Primary workload balancing - Send work to the system with available resources
    - VTAM GR, DNS/WLM, WLM JES init mgmt,...
  - Once work lands on a system, it generally has to stay on that system

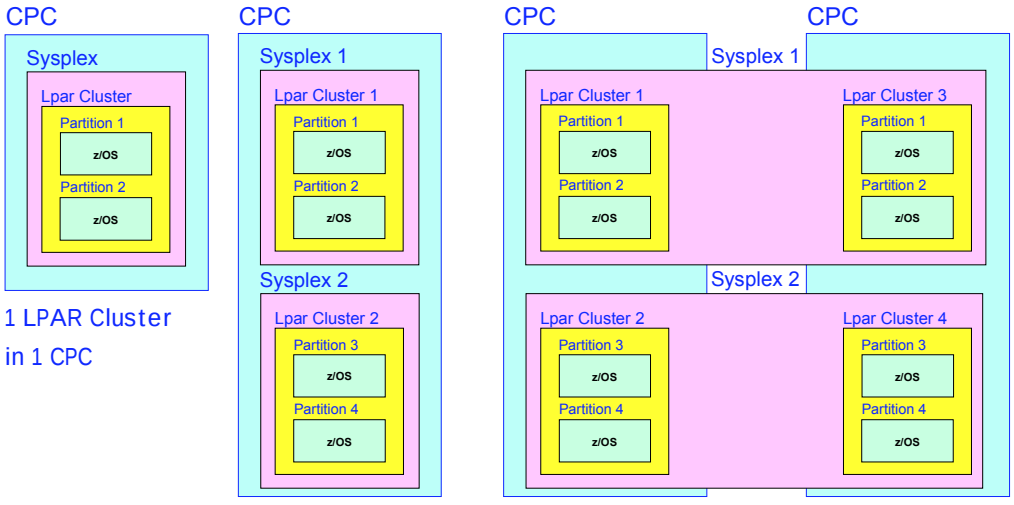
Background: Logical Partitioning

- Value of logical partitioning
  - Workload Isolation
  - Server Consolidation
  - Development, Test, and Maintenance
- Expect the value of logical partitioning to grow
  - Size of CECs growing rapidly
  - Continuing trend for server consolidation
  - Most major server platforms pursuing some form of partitioning: RS/6000, AS/400, SUN

LPAR Clusters

- Definition:
  - Set of z/OS images running in partitions of the same CEC that are part of the same sysplex
- Objective
  - Leverage platform strengths through integration
    - Workload Manager, Parallel Sysplex, LPAR
  - View partitions on CEC as single pool of computing resource
    - Move physical resource to logical workloads
    - Extend goal oriented resource management across logical partitions transparently to application subsystems
    - Initial resources managed: CPU and I/O

LPAR Clusters: Example



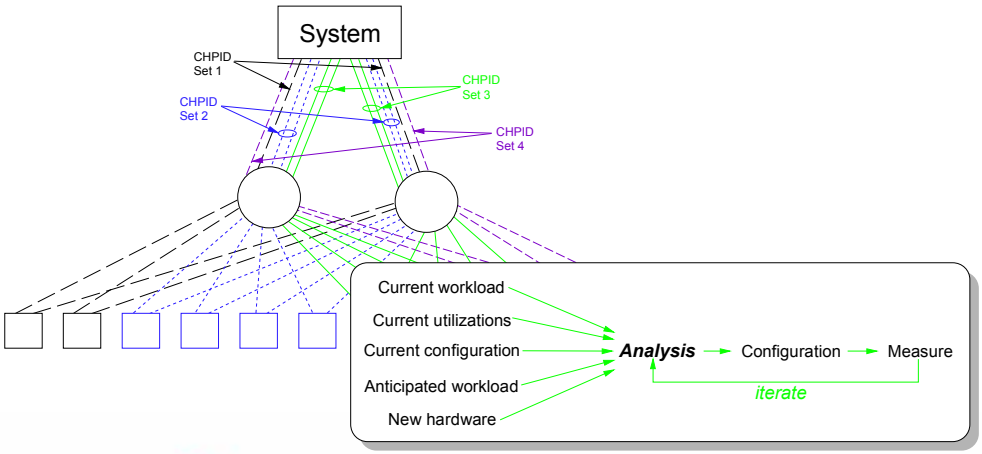
Dynamic Chpid Management

Dynamic Chpid Management: DCM

- Description
  - Allow the system to dynamically manage channel paths in response to changing workload demands
  - Moves channel capacity (bandwidth) to the DASD subsystem(s) experiencing need based on workload requirements
- Benefits
  - More efficient use of hardware resource
  - Reduces channel requirements
  - Simplifies I/O configuration planning and definition
  - Dynamically balances I/O connectivity based on workload demand

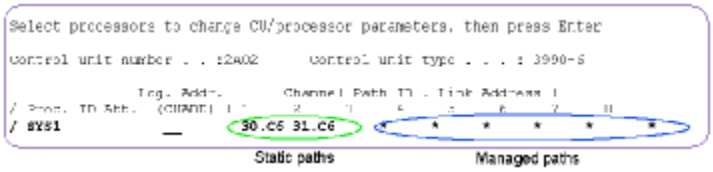
Without DCM

- statically associate subsets of channels with subsets of CUs
- requires analysis by customer



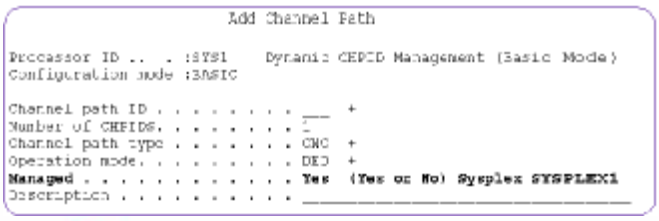
DCM Environment

- Managed Subsystem
  - Control unit(s) with all DASD devices in common
  - Must be switch connected
  - HCD control unit definition specifies the number of static/dynamic channel paths
    - Minimum of one static path required, two recommended
    - Configure enough static channels to support the normal workload
    - Limit of eight (8) total channels (static+dynamic)
    - Preferred path not allowed



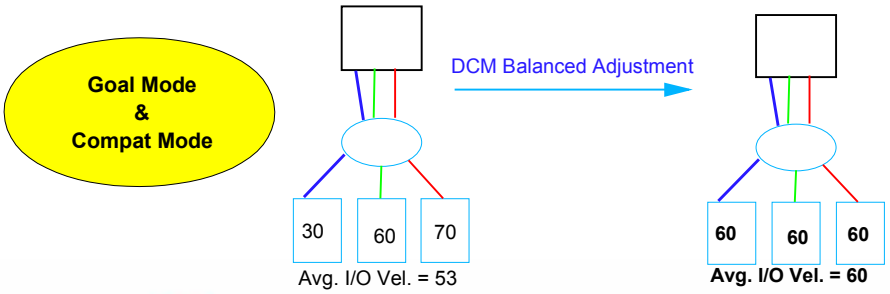
DCM Environment ...

- Static Channel
  - Channels assigned to a specific control unit using HCD/HCM
  - Cannot be dynamically moved
  - Each control unit within a managed subsystem must have at least one static channel path defined per image (two are recommended)
- Managed Channels
  - Defined through HCD as being capable of movement to managed subsystems
  - Associated with a specific LPAR cluster via sysplex name
  - Specify dynamic switch number (recommendation: also define switch ID and port)
  - Connected to a switch providing dynamic connections: ESCON, FICON Bridge
  - Shared via EMIF when in an LPAR environment



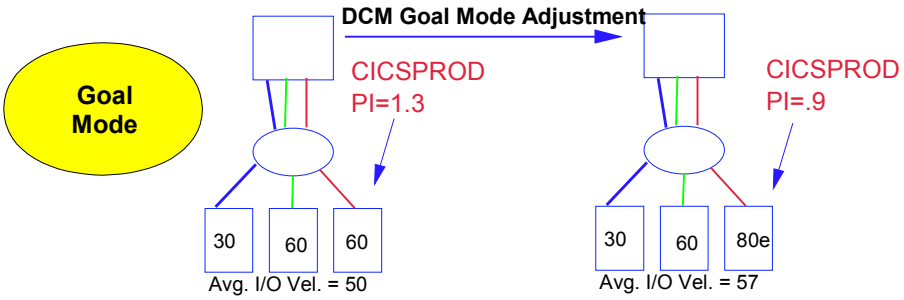
DCM Terminology

- Subsystem I/O Velocity
  - Metric used to determine the responsiveness of channels
  - Ratio of connect time to channel delay time
- DCM Balanced Mode
  - Channels are dynamically assigned to achieve the average I/O velocity across managed subsystems



DCM Terminology

- DCM Goal Mode
  - Channels are managed during times of contention
  - WLM can assign an explicit velocity to a subsystem to help a service class period to meet it's goals



DCM Hardware/Software Requirements

- *Warning!*
  - DCM is very dependent on self description data returned by DASD control units in order to understand the connectivity view of the hardware configuration. There has been a problem confirmed for some non-IBM DASD where the information was not consistent in its presentation.

Before using DCM with non-IBM vendor DASD, please contact your vendor for their specific DCM service requirements.

Channel Subsystem Priority Queueing

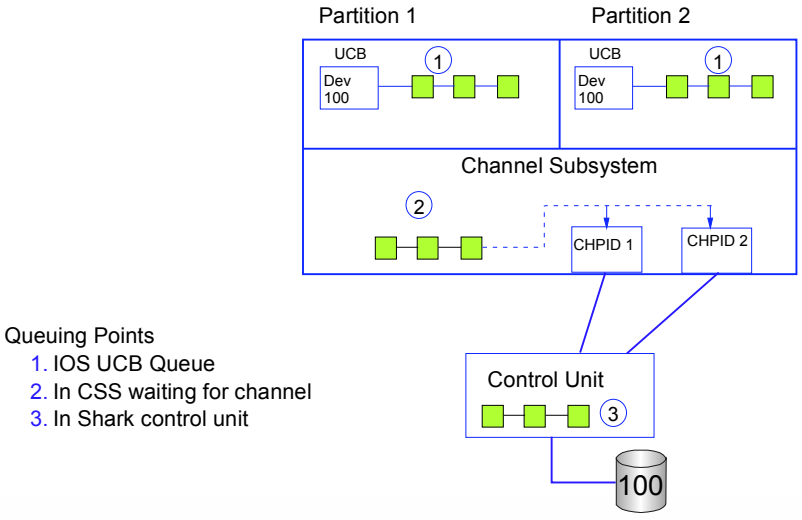
Channel Subsystem Priority Queueing

- New hardware function on z900 that allows OS to specify priority value to channel subsystem when I/O request started
  - When there is contention for channel resources, requests will be prioritized by this value
- z/OS in goal mode will set priority value based on goals of WLM policy
- Complementary to current goal mode priority management
  - IOS UCB queues
  - Shark control unit

Channel Subsystem Priority Queueing

- Function can be enabled/disabled on CEC
- In LPAR mode a range of priority values can specified for each partition
  - LPAR will ensure a partition only uses priority value within its range
    - Range can be used to give fixed priority to all requests from a partition running an operating system that does not set priority
    - Range can be used to prioritize one LPAR cluster against another on the same CEC
    - WLM management scope is LPAR Cluster
    - Same range should be set for partitions in an LPAR cluster
    - Allows WLM to manage based on goals

I/O Requests Queueing Points



Queuing Points
1. IOS UCB Queue
2. In CSS waiting for channel
3. In Shark control unit

WLM LPAR CPU Management

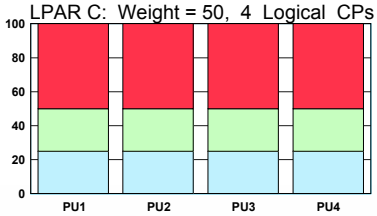
LPAR Weight Management
VARY Logical CPU Management

LPAR CPU Management

- Current Problems
  - LPAR cannot dynamically adjust CPU resource allocations across images based on workload demands to meet goals
  - LPAR distributes a partition weight evenly across the partition's online logical CPUs
  - Example: CPC with 4 PUs, 3 LPARs configured

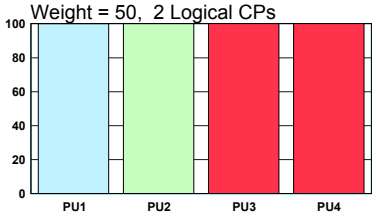
Example 1:

LPAR A: Weight = 25, 4 Logical CPs
LPAR B: Weight = 25, 4 Logical CPs
LPAR C: Weight = 50, 4 Logical CPs



Example 2:

Weight = 25, 1 Logical CP
Weight = 25, 1 Logical CP
Weight = 50, 2 Logical CPs



IBM @server. For the next generation of e-business.

LPAR CPU Management: Benefits

- LPAR Weight Management
  - Provides flexibility in managing CPU resources across Logical PARtitions in accordance with workload goals.
  - WLM can shift weight from one partition to another in the same LPAR cluster
- VARY Logical CPU Management
  - Optimize number of on-line logical CPUs based on partition's current weight and CPU consumption
  - Best to set the number of on-line CPUs so "logical CPU speed" is close to physical CPU speed. To achieve this WLM maintains the number of on-line logical CPUs to match the CPU consumption of the partition

LPAR Weight Management

- New Controls provided as part of LPAR definition
  - Enable/Disable CPU Mgmnt for partition
  - Initial weight
    - used when partition joins LPAR cluster
    - restored when partition leaves LPAR cluster
  - Min weight
    - WLM does not reduce below minimum
  - Max weight
    - WLM does not increase above maximum

LPAR Weight Management

- Definitions
  - if system is not fully utilized:
    - LPARs can have higher share of processor resources if no capping has been defined (same as today)
  - if system is highly utilized: all partitions use their weight
    - Example:
      - LPAR Cluster = PARTA + PARTC = Weight 200
      - PARTA = Weighting Range from 50 to 180
      - PARTC = Weighting Range from 20 to 150
      - Absolute share of processor capacity depends on the system utilization caused by PARTB and PARTD

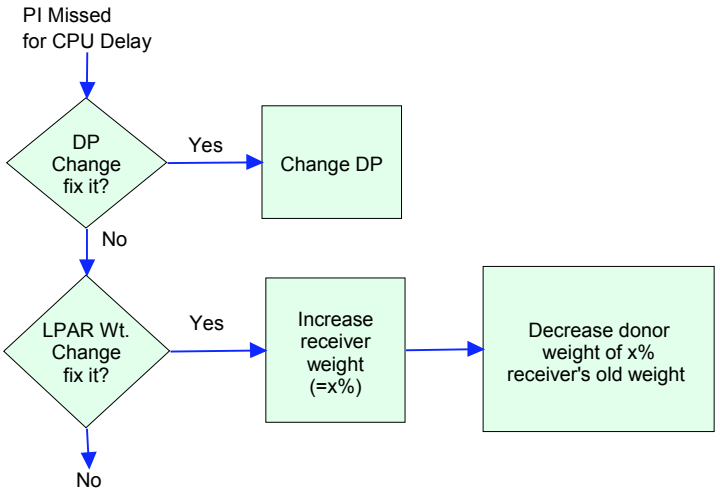
Logical Partition	Active	WLM Managed	Initial Processing Weight	Minimum Processing Weight	Maximum Processing Weight
PARTA	Yes	Yes	100	50	200
PARTB	Yes	No	100		
PARTC	Yes	Yes	100	20	150
PARTD	Yes	No	50		
PARTE	No	No	5		

LPAR Weight Management

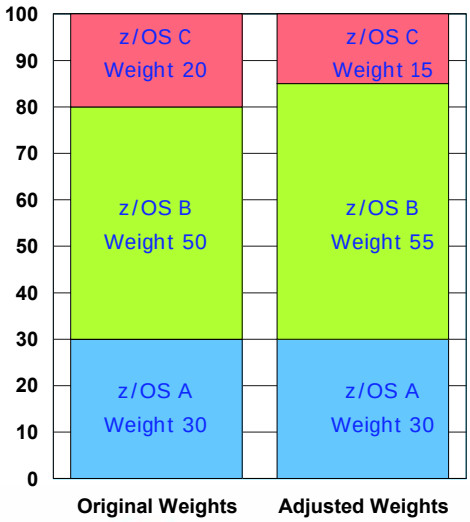
- Notes
  - Current partition weight becomes an initial value
  - WLM keeps the sum of the weights of an LPAR cluster constant.
    - This keeps the overall weight of the LPAR cluster constant compared to other partitions not in the cluster.
  - WLM does not manage the LPAR weights of the partitions that are capped or have dedicated CPUs
  - Partition reverts to its initial weight when
    - WLM is switched to COMPATIBILITY mode
    - the partition leaves the sysplex (system reset)
    - when WLM CPU Management is disabled
  - LPAR adjusts the weight of the partition when partition's minimum or maximum weight is changed
  - If WLM loses connectivity to its CF structure, the weights become static
    - They do not revert to initial weights
    - They can be changed via the HMC

LPAR Weight Management

- Algorithm



LPAR Weight Management: Example



PART C Donor
PART B Receiver
PART A

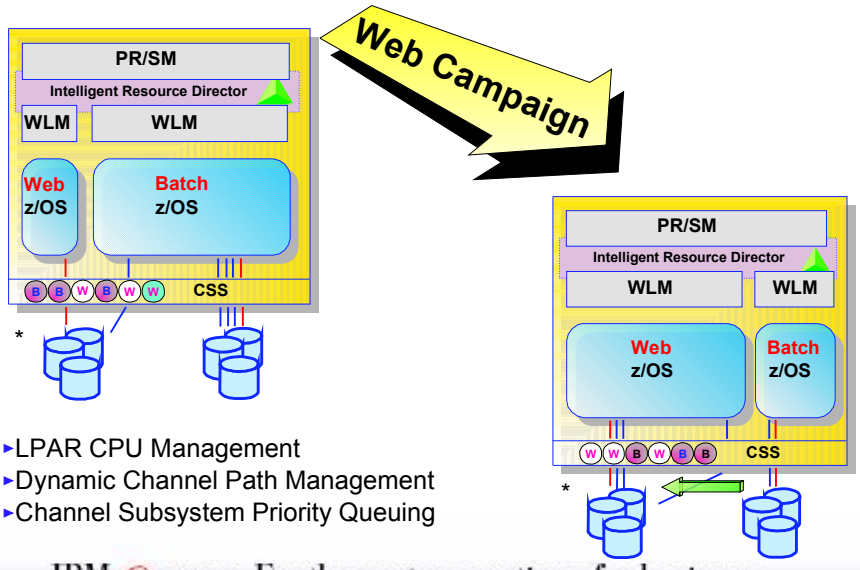
Assumption:
- CPC Utilization 100%
- LPAR Definitions
  - PART A, Min=10, Max=100
  - PART B, Min=10, Max=100
  - PART C, Min=10, Max=100
  - 3 z/OS systems run in the same sysplex
  - 3 LPars are all WLM managed

Select Donor/Receiver
- System B: Goals can't be achieved
- Workload running on system C meets its goals and taking away resources will not impact "important" work
- Choose system C as donor
- Increase weight of B
  - Example: by 10% of current weight of B
- Decrease weight of C
  - Example: by 10% of current weight of B

VARY Logical CPU Management

- Notes
  - Weight management must be enabled
  - Enabled or disabled in IEAOPTxx member of parmlib
    - VARYCPU=YES/NO
  - New construct - Equivalent Physical CPs
    - Minimal number of logical CPs required for a partition
  - WLM does not manage any CPU with special features
    - Crypto
    - Integrated Linux Facility (ILF)
  - WLM keeps the appropriate number of CPUs enabled for I/O interrupts
  - WLM and Operator Actions do not interfere ...
    - Operator cannot vary a CPU online that was taken offline by WLM
    - WLM will not vary a CPU online that was taken offline by the operator

Example: Value of IRD



- LPAR CPU Management
- Dynamic Channel Path Management
- Channel Subsystem Priority Queuing

IRD Summary

- gives you
  - dynamic goal-based resource allocation of I/O and CPU in a single CPC across all images in a sysplex
- based on proven IBM technologies
  - z/OS Workload Manager
  - LPAR management
  - Coupling Facility
- using a collection of new functions
  - Dynamic Channel Path Management
  - Channel Subsystem Priority Queuing
  - LPAR CPU Management

IRD Roadmap & Tools

- Implement Resource Sharing
  - Use z/OS Managed System Infrastructure (msys) for Setup to assist in generating resource sharing environment available in z/OS 1.1 (March, 2001)
  - Redbook: Parallel Sysplex Resource Sharing (SG24-5666)
  - Parallel Sysplex Configuration Assistant tool is available for non-z/OS users (ibm.com/s390/pso)
- Implement WLM Goal Mode
  - Tool: Goal Mode Migration Aid (ibm.com/s390/rmf)
  - Hot Topics #3 (ibm.com/s390/os390)
- Exploit IRD
  - Redbook: z/OS IRD (SG24-5952)
  - Hot Topics #4 (ibm.com/s390/os390)
  - Joan Kelley's SHARE presentation 2522
  - LPAR Clustering (IRD) in Action
  - Walt Caprice's presentations on DCM and LPAR CPU Mgmt
- Check for Required Service
  - DCM: non-IBM DASD may be require updates, check with vendors
  - IRD PSP bucket: Upgrade ID=2064DEVICE , Subset = IRD

IRD Extensions for z/OS 1.2

- Extends WLM's ability to manage non-z/OS and non-OS/390 partitions on the same CPC
  - Target to include Linux partitions in LPAR cluster
  - Initially for weight management only
- Annotations
  - IFLs' not supported
  - "Linux partition" is classified as one workload to a service class with Execution velocity goals
  - z/OS WLM decides whether Linux partitions need more or less weight to meet their goals