z/VM Resource Manager

Steve Wilkins, Sr. Software Engineer

Christine T. Casey, Sr. Software Engineer

z/VM Development

Endicott, NY

WAVV 2005	May 20-24, 2005

Colorado Springs,
Colorado

Disclaimer

The information contained in this document has not been submitted to any formal IBM test and is distributed on an "AS IS" basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

In this document, any references made to an IBM licensed program are not intended to state or imply that only IBM's licensed program may be used; any functionally equivalent program may be used instead.

Any performance data contained in this document was determined in a controlled environment and, therefore, the results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environments.

It is possible that this material may contain reference to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM products, programming or services in your country.

Trademarks

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

CICS*
DB2*
DB2 Connect
DB2 Universal Database
DFSMS/MVS*
DFSMS/VM*
e business(logo)*
Enterprise Storage Server*
ESCON*
FICON
GDDM*
HiperSockets
IBM*
IBM(logo)*
Language Environment*
MQSeries*
Multiprise*
MVS
NetRexx
OpenEdition*
OpenExtensions
OS/390*
Parallel Sysplex*
PR/SM
QMF
RACF*
RAMAC*
RISC
S/370
S/390*
S/390 Parallel Enterprise Server
VisualAge*
VisualGen*
VM/ESA*
VTAM*
VSE/ESA
WebSphere*
z/Architecture
z/OS*
zSeries*
z/VM*

* Registered trademarks of the IBM Corporation

The following are trademarks or registered trademarks of other companies.

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation.
Tivoli is a trademark of Tivoli Systems Inc.
Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries
UNIX is a registered trademark of The Open Group in the United States and other countries.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.

Notes:

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

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

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

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

IBM considers a product “Year 2000 ready” if the product, when used in accordance with its associated documentation, is capable of correctly processing, providing and/or receiving date data within and between the 20th and 21st centuries, provided that all products (for example, hardware, software and firmware) used with the product properly exchange accurate date data with it. Any statements concerning the Year 2000 readiness of any IBM products contained in this presentation are Year 2000 Readiness Disclosures, subject to the Year 2000 Information and Readiness Disclosure Act of 1998.

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

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Objectives

- Manage workloads to CPU and DASD I/O velocity goals
- Allow I/O priority queuing to be exploited on behalf of VM-based workloads
- Provide an infrastructure for more extensive workload management for future releases of z/VM
  - First released with z/VM 4.3.0
  - Enhanced with z/VM 4.4.0 and z/VM 5.1.0

Overview

- The Service Virtual Machine - VMRMSVM
  - The PROFILE EXEC for VMRMSVM begins operation of the server by calling the IRMSERV EXEC
    - May also be invoked from the command line
  - IRMSERV reads the customer-supplied definition file
    - Default is VMRM CONFIG A
    - Any other file name can be passed to the IRMSERV EXEC
- Uses VM monitor data
  - Obtains regular measurements (default 1 minute intervals) of virtual machine resource consumption

Overview (cont.)

- Based on definition of workloads, goals and priorities in the configuration file, the SVM...
  - Computes the achievement levels of interest for each workload
  - Selects one workload to adjust:
    - For each goal type of CPU and DASD
    - based on the customer-supplied importance value
  - Adjusts virtual machine tuning parameters to achieve defined goals

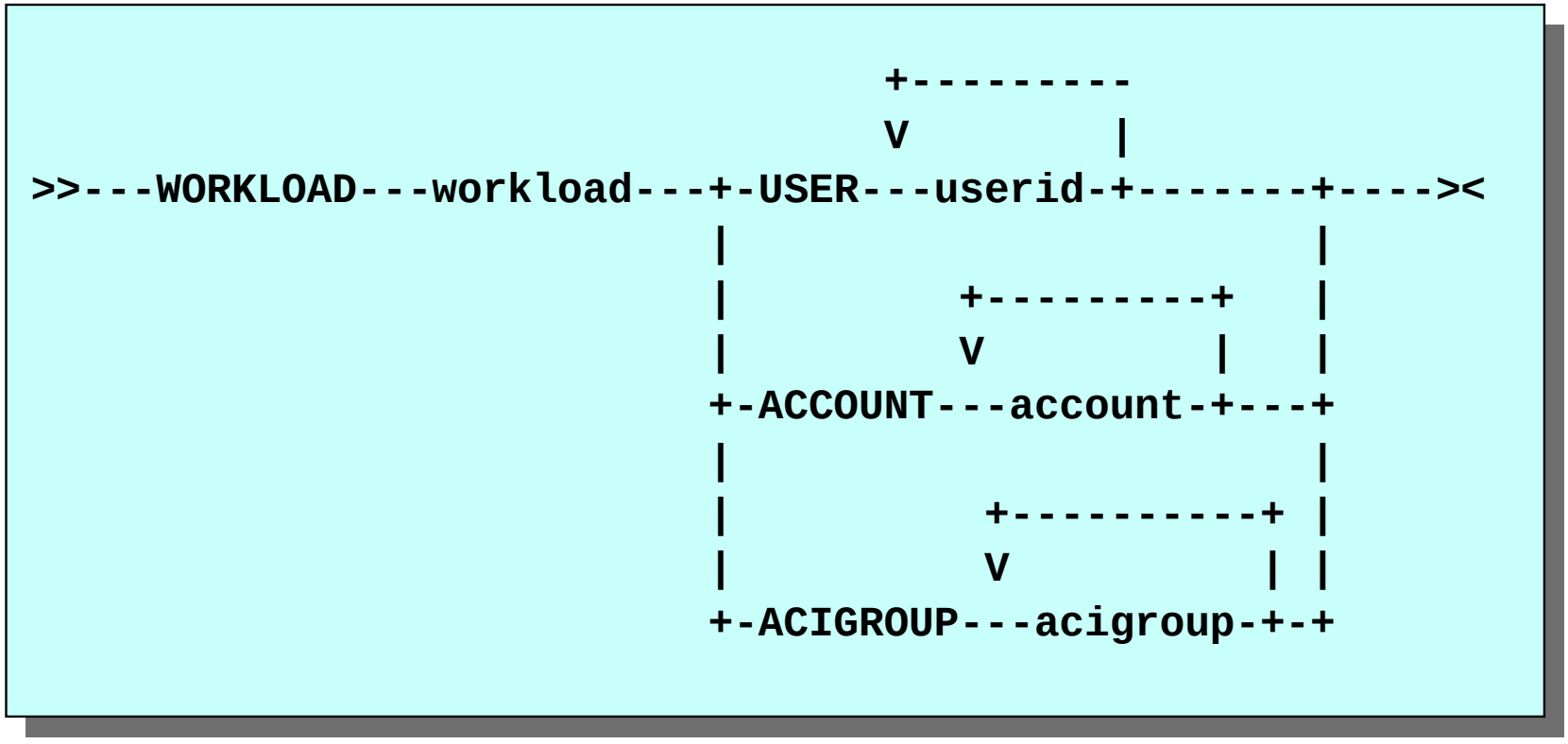
VMRM CONFIG File

- The VMRM CONFIG file supports 4 types of statements:
  - **WORKLOAD** - describes a workload by userid, account id, acigroup
  - **GOAL** - describes a DASD or CPU velocity goal
  - **MANAGE** - associates a workload with a goal and assigns an importance value
  - **ADMIN** - identifies a user to receive VMRM server messages and/or filename and directory for a new config file
- Syntax checking is performed on the configuration file
  - The server will not start if ANY errors found

WORKLOAD Statement

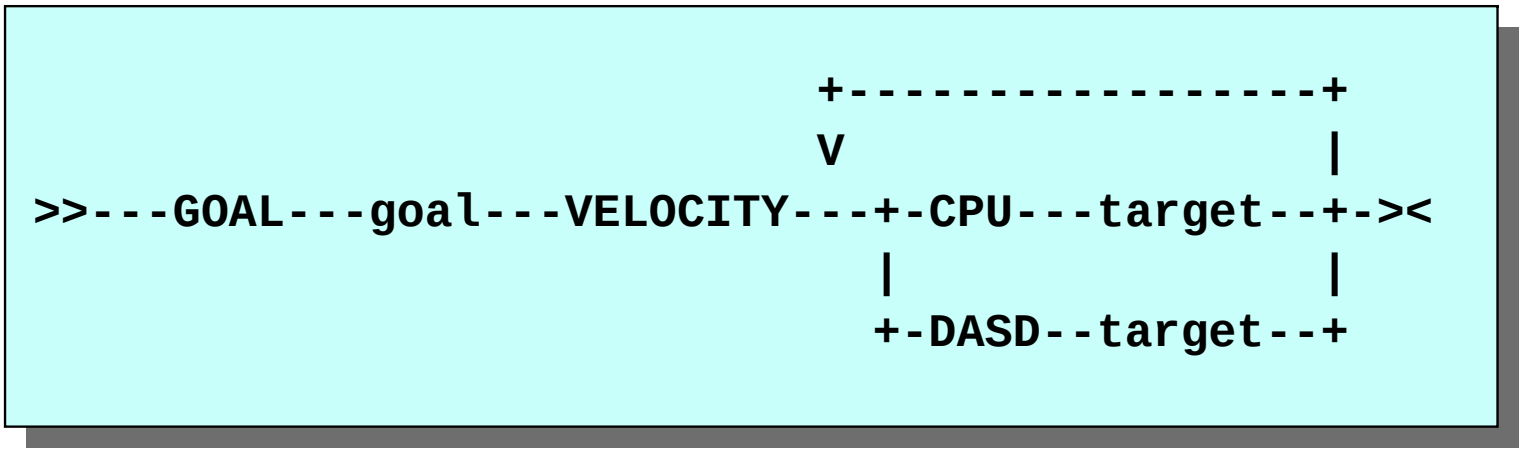
- A workload is comprised of one or more virtual machines identified by user ID, account ID, or ACI group name
  - Wildcarding allowed for user IDs:

WORKLOAD work1 USER Linux* chrisC jonR



GOAL Statement

- The GOAL statement specifies velocity goals for:
  - CPU - percentage of the time the user should receive CPU resources when it is ready
  - DASD - percentage of time that the user's DASD I/O requests are not outprioritized
  - Both CPU and DASD may be specified on one statement



MANAGE Statement

- Associates a workload with a goal
- Assigns an importance value to the relationship
  - Importance values can range from 1-10 (10 is most important)
- Only one manage statement is allowed for each workload

```
>>---MANAGE---workload---GOAL---goal---IMPORTANCE---value--->< 
```

ADMIN Statement

- Specifies a user ID on the same system where messages can be sent from the service virtual machine if necessary
  - Messages will also be logged to **VMRM LOG1 A**
- Specifies a filename, filetype and fully-qualified SFS directory name where a new configuration file resides
  - Can be put into production at a later time
- If multiple ADMIN statements exist, only the last will be used

```
>>--ADMIN---MSGUSER---userid---NEWCFG---fn---ft---dirid--->< 
```

ADMIN Statement: NEWCFG option

- Allows dynamic restart of the server with a new configuration file
- The VMRM SVM must be given READ access to the SFS directory and the configuration file(s) in that directory
  - Allows multiple config files to reside on an SFS directory
    - Can be placed into production after the server started
    - Server will detect when the file changes
    - Automatically restarts the server using the information in the new configuration file
  - Systems Management APIs can be used for update/query

Sample VMRM CONFIG File

```
*   This is a valid comment line  *
/*  So is this                    */
;   and this
ADMIN    MSGUSER  Chris
WORKLOAD work1    USER abcde,
                  a123 456
WORKLOAD work2    USER fghij*
WORKLOAD workabcd USER qrst
WORKLOAD work3    ACCOUNT 1234 5678
WORKLOAD work4    ACIGROUP  ABC
GOAL   goal1,     /* continuation allowed */
             VELOCITY CPU  10
GOAL   goal2 VELOCITY DASD 50
GOAL   goal3 VELOCITY CPU  80  DASD 20
MANAGE work1 GOAL goal1,
                          IMPORTANCE 10
MANAGE work2 GOAL goal1 IMPORTANCE  5
MANAGE work3 GOAL goal2 IMPORTANCE  2
MANAGE work4 GOAL goal3 IMPORTANCE 10
MANAGE workabcd   GOAL  goal2 IMPORTANCE  7
```

Configuration File APIs - 5.1.0

- Systems Management APIs for VMRM
  - VMRM_Configuration_Update
    - Updates a VMRM configuration file remotely from an RPC client using the NEWCFG support
  - VMRM_Configuration_Query
    - Query a VMRM configuration file remotely from an RPC client
  - VMRM_Measurement_Query
    - Query workload measurements from an RPC client - - returns workload goal and actual data

Verifying a Config File

- SYNCHECK option allowed on server invocation

  IRMSERV TEST CONFIG A (syncheck

  - Syntax checks a configuration file without starting the server
  - Allows Class G users to check a configuration file before it is put into use by the server
  - VMRM_Configuration_Update API always performs syncheck before updates go into production

VMRM Log File

- **VMRM LOG1 A** file used to log:
  - Messages sent to MSGUSER
  - Additional SVM events; measurement data
  - Debug messages
  - variable record format used (RECFM V)

- VMRM LOG1 A will be copied to VMRM LOG2 A
  - when it reaches 10,000 records.
  - VMRM LOG1 will then be erased and rewritten

Sample VMRM log file

```
2005-02-19 17:02:02 ServExe  MSG

MSG       IRMSER0022I VM Resource Manager Initialization started
PCfg      VMRM CONFIG A1 2/19/05 17:01:55
MSG       IRMSER0008W The ADMIN message user ID is not logged on..
InitEnv   Monitor sample started -- recording is pending
InitEnv   HCPMNR6224I Sample recording is pending because there...
InitEnv   MONITOR EVENT INACTIVE   BLOCK    4   PARTITION    0
InitEnv   MONITOR DCSS NAME - NO DCSS NAME DEFINED
InitEnv   CONFIGURATION SIZE        68 LIMIT      1  MINUTES
InitEnv   CONFIGURATION AREA IS FREE
InitEnv   USERS CONNECTED TO *MONITOR - NO USERS CONNECTED
InitEnv   ...
InitEnv   ...more data from Q Monitor...
InitEnv   ...
MSG       IRMSER0023I VM Resource Manager Initialization complete.
          Proceeding to connect to Monitor.
Exit      STARMON completed. RC=0
ExitSVM   Monitor sample stopped
MSG       IRMSER0012I VM Resource Manager shutdown in progress
```

Workload Selection

- Selection criteria
  - Workloads are selected first based on their importance value
  - If a workload was selected in the last interval either for improvement or degradation, it is skipped and an attempt is made to select another
  - If there are workloads of equal importance, the workload farthest from its goal is selected
  - Eligible users within a workload will have their SHARE or IOPRIORITY adjusted appropriately based on how far they are from the workload goal

Some Terminology

- Absolute vs. Relative
  - **Absolute** specifies a user is to receive a target minimum of nnn% of the scheduled system resources
  - Amount of resources available to relative share users = total resources available less the amount allocated to absolute share users
  - **Relative** portion that the user receives is nnnn / sum of all relative share users
  - VM Resource Manager will **not** adjust Absolute users
- Limithard vs. Limitsoft
  - **Limithard** specifies the user's share of CPU resource is limited (they do not receive more than maximum share of the CPU resource)
  - **Limitsoft** specifies that the user's share of CPU resource is limited, **but** the limit can be exceeded if the capacity is available

Adjustment Algorithms

- Individual users within the selected workload may be adjusted based on calculations from monitor data
- For CPU goals:
  - User must have a Relative SHARE setting
  - User does not have Limithard specified on their CPU SHARE setting
  - Sum of wait deltas and run deltas is > current sample size of 5
  - CPU actual = run delta / (run delta + wait delta) * 100
- For DASD goals:
  - User must have a Relative I/O Priority setting
  - Sum of I/O deltas and Outprioritized deltas is > current sample size of 5 for DASD
  - DASD actual = IO delta / (IO delta + outprior delta) * 100
- After above criteria is met, if user is not within 5% of workload goal, then they can be adjusted.

Adjustment Algorithms

- Determine how much to adjust each user
  - For CPU goals:
    relvalue = (Workload CPU goal / User actual) * User current share

    -- checking that value falls within 1-10,000 range

  - For DASD goals:
    relvalueLo = (Workload DASD goal / User actual) * User curr IO Lo

    relvalueHi = relvalueLo + (User curr Hi - User curr Lo)

    -- checking that values fall within 0-255 range

- Set Share and/or Set IOPriority command is issued on behalf of the user

I/O Priority Queuing

- Enables prioritization of virtual machine I/O
  - Guest's I/O priority queuing range may be set via
    - IOPRIORITY directory statement
    - SET IOPRIORITY command
  - To be queried via QUERY IOPRIORITY command
  - If I/O Priority Queuing is available and enabled (zSeries only)
    - I/O Priority Queuing low/high range is obtained from the hardware
    - Guest I/O Priority Queuing values are mapped to fall within that range
    - CP I/O uses highest value available
  - If not available or enabled, CP simulates a range of 0-255
  - For I/O priority-aware guests, the priority associated with the guest I/O requests will be enforced
  - For non I/O priority-aware guests, CP assigns a priority value

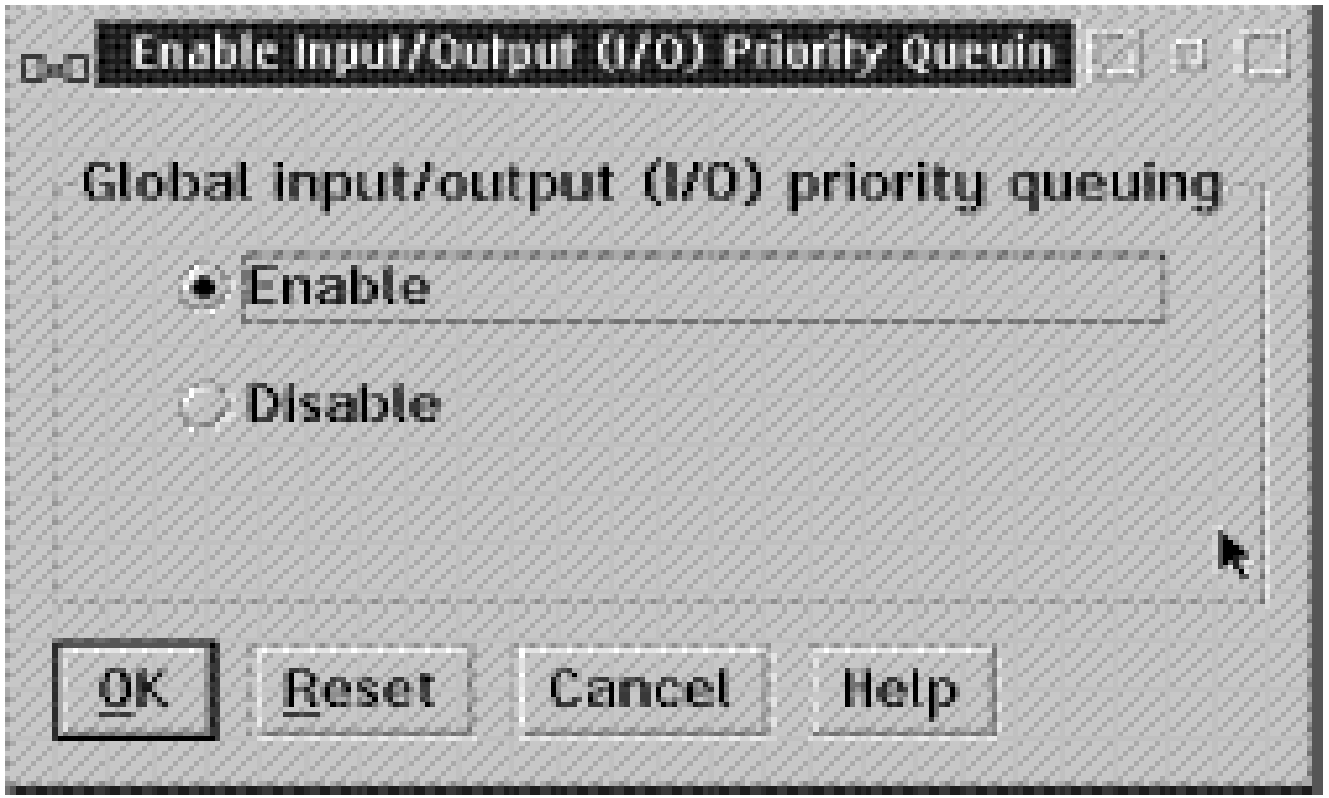
I/O Priority Queuing Mappings

- Mapping of requested range to "effective" range is based on whether hardware facility exists:

	Relative	Absolute
Hardware Not Enabled	0 - 255 on command maps to simulated effective range of 0 - 255	0 -255 on command maps to simulated effective range of 0 - 255
Hardware Enabled	0 - 255 on command maps proportionally to hardware range	User input maps directly to hardware range

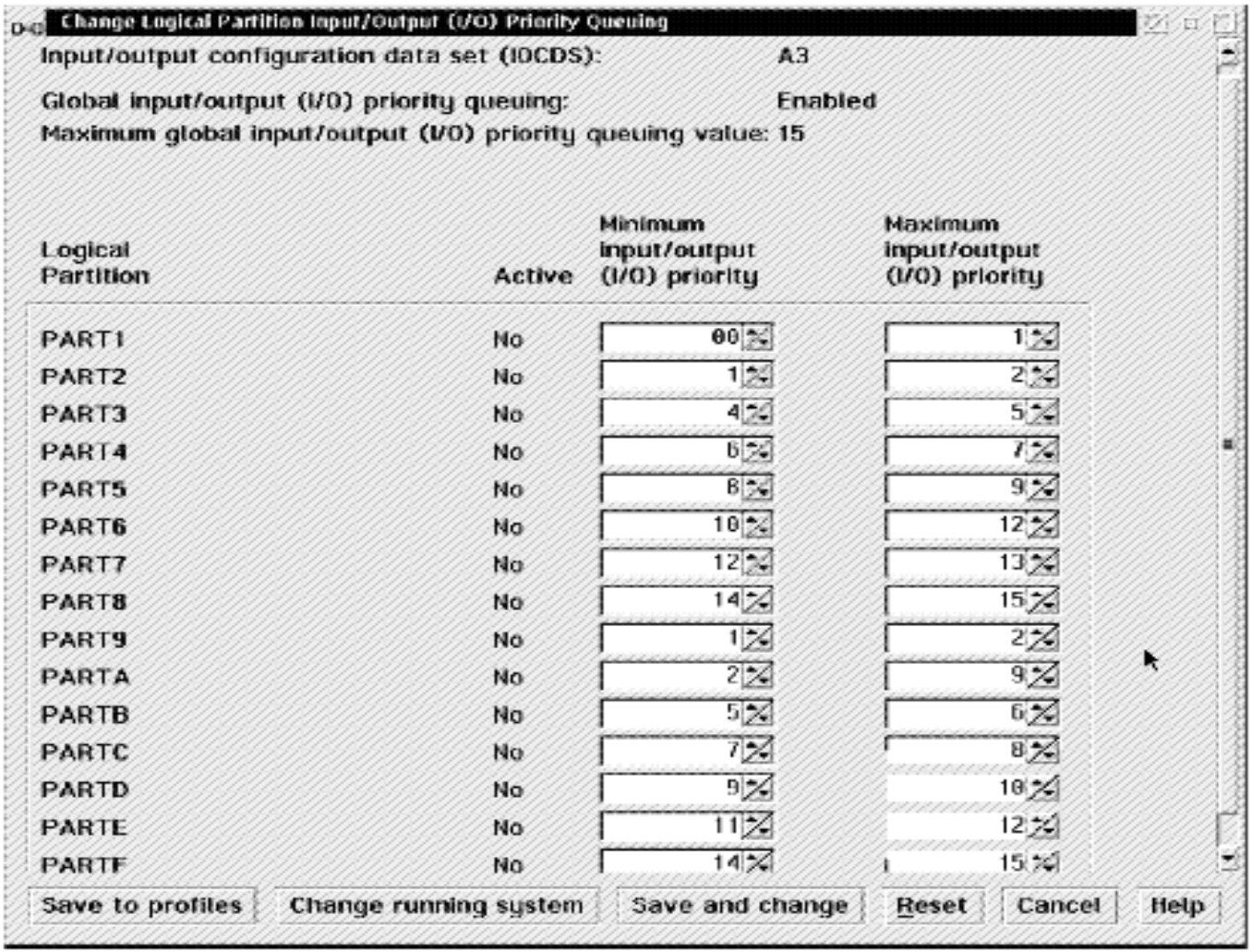
Enabling I/O Priority Queuing on zSeries Processors

- At the HMC use the "Enable I/O Priority Queuing task"
  - Available from the Central Processor Complex Operational Customization tasks list to either enable or disable I/O priority queuing for the entire CPC



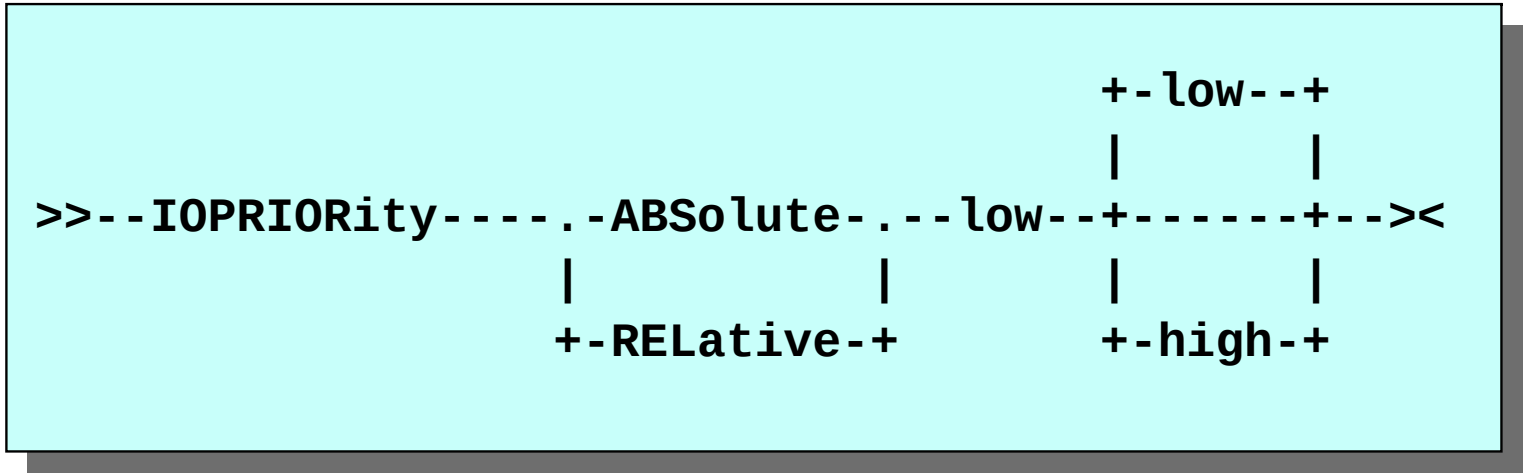
Setting Hardware I/O Priority Queuing Ranges

- Use the change LPAR I/O priority queuing task to set the minimum and maximum I/O priority queuing values



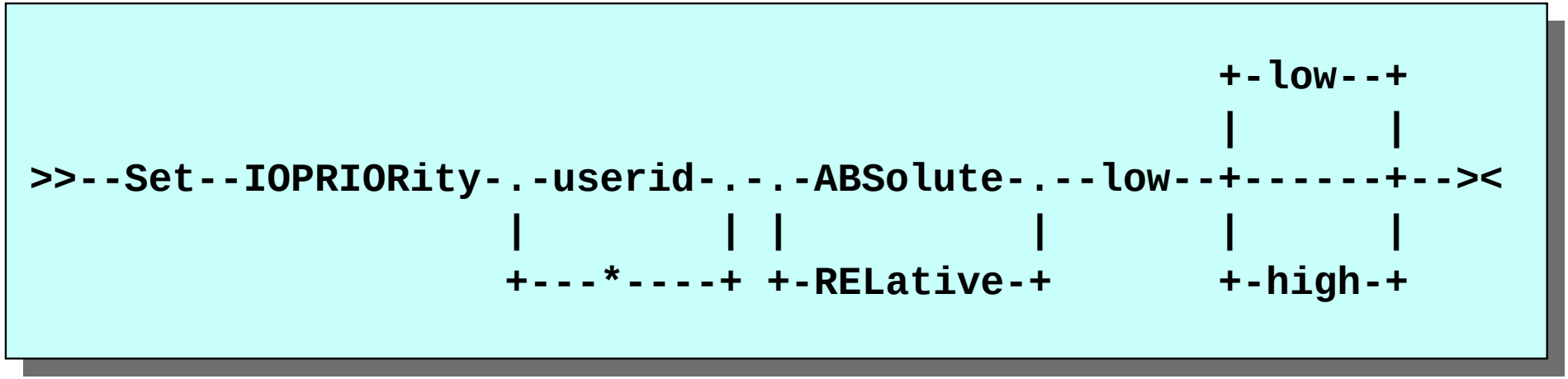
IOPRIORITY
Directory Statement

- Specifies the I/O priority range to be set when the user logs on
  - Low and high values must be decimal numbers from 0 to 255
  - If hardware priority queuing is available and enabled
    - Absolute priority ranges outside the range available to CP are clipped to fall within that range
    - Relative ranges are mapped to fall within the range available to CP
  - If IOPRIORITY is not specified in the directory, low and high are set to a relative value of 0



SET IOPRIORITY COMMAND

- A class A privileged user can adjust a guest's I/O Priority Queuing range using the CP SET IOPRIORITY command
  - Absolute: must fit in range available to CP (or will be clipped)
  - Relative: maps proportionally to the available range



QUERY IOPRIORITY COMMAND

- A class A or E user can display a guest's or the system I/O Priority Queuing range

```
>>--Query--IOPRIORity--.-userid-.--><
                        |         |
                        |---*----|
                        |         |
                        +-SYSTEM-+
```

  - **userid** requests the priority range of a given user ID
  - ***** requests the priority range of the user issuing the command
  - **SYSTEM** requests the priority range available to CP

Query IOPRIORITY Responses

- userid REQUESTED RANGE nnn mmm ABSOLUTE
  EFFECTIVE RANGE xxx yyy

- userid REQUESTED RANGE nnn mmm RELATIVE
  EFFECTIVE RANGE xxx yyy

where:

requested range indicates low and high ranges requested

effective range is the low and high range that CP will allow for this user

Examples of Absolute I/O Priority Queuing Ranges

- If the I/O priority queuing range available to CP is 50-75
  - Virtual machine requests for ranges from 0-49 will be assigned absolute value of 50
  - Virtual machine requests for ranges 50-75 will be accepted
  - Virtual machine requests for ranges 75-255 will be assigned an absolute value of 75

Examples of Relative I/O Priority Queuing Ranges

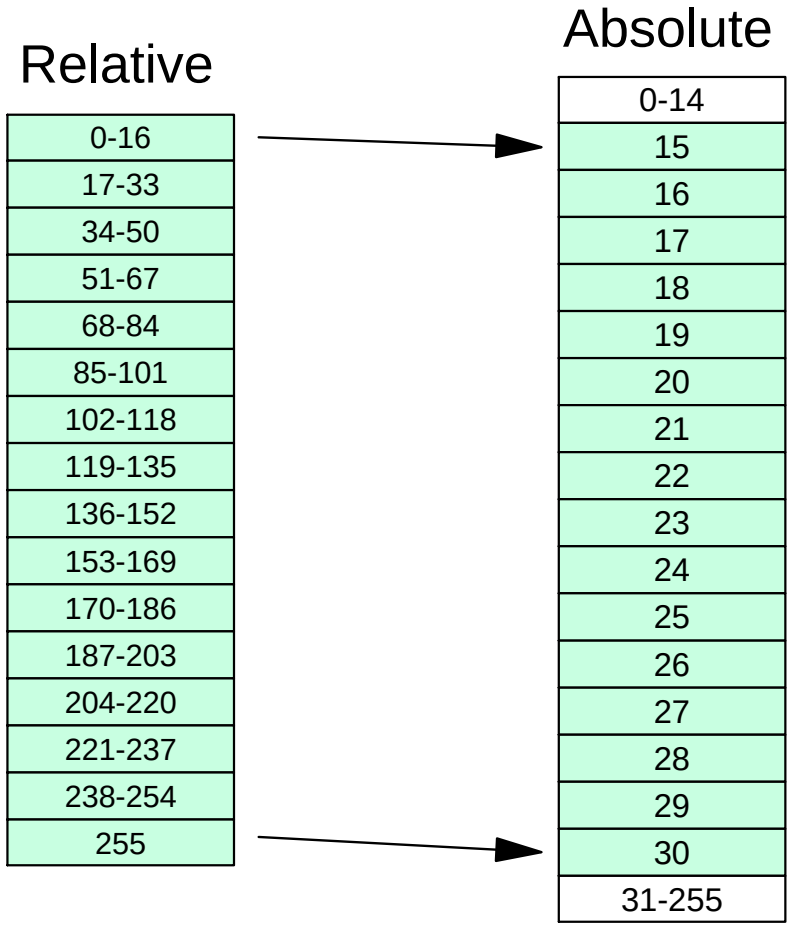
- The effective I/O priority queuing value is calculated from the requested value and the range available to CP

$$Eff\_Val = TRUNC\left(\frac{Rel\_Val * (CP\_Hi - CP\_Lo)}{255}\right) + CP\_Lo$$

- Where:
  - **Eff_Val** is the effective I/O priority
  - **Rel_Val** is the relative I/O priority
  - **CP_Hi** is the highest I/O priority value available to CP
  - **CP_Lo** is the lowest I/O priority value available to CP

Examples of Relative I/O Priority Queuing Ranges

- If the range of I/O priority values available to CP is 15-30 then relative priorities map to absolute priorities as follows:



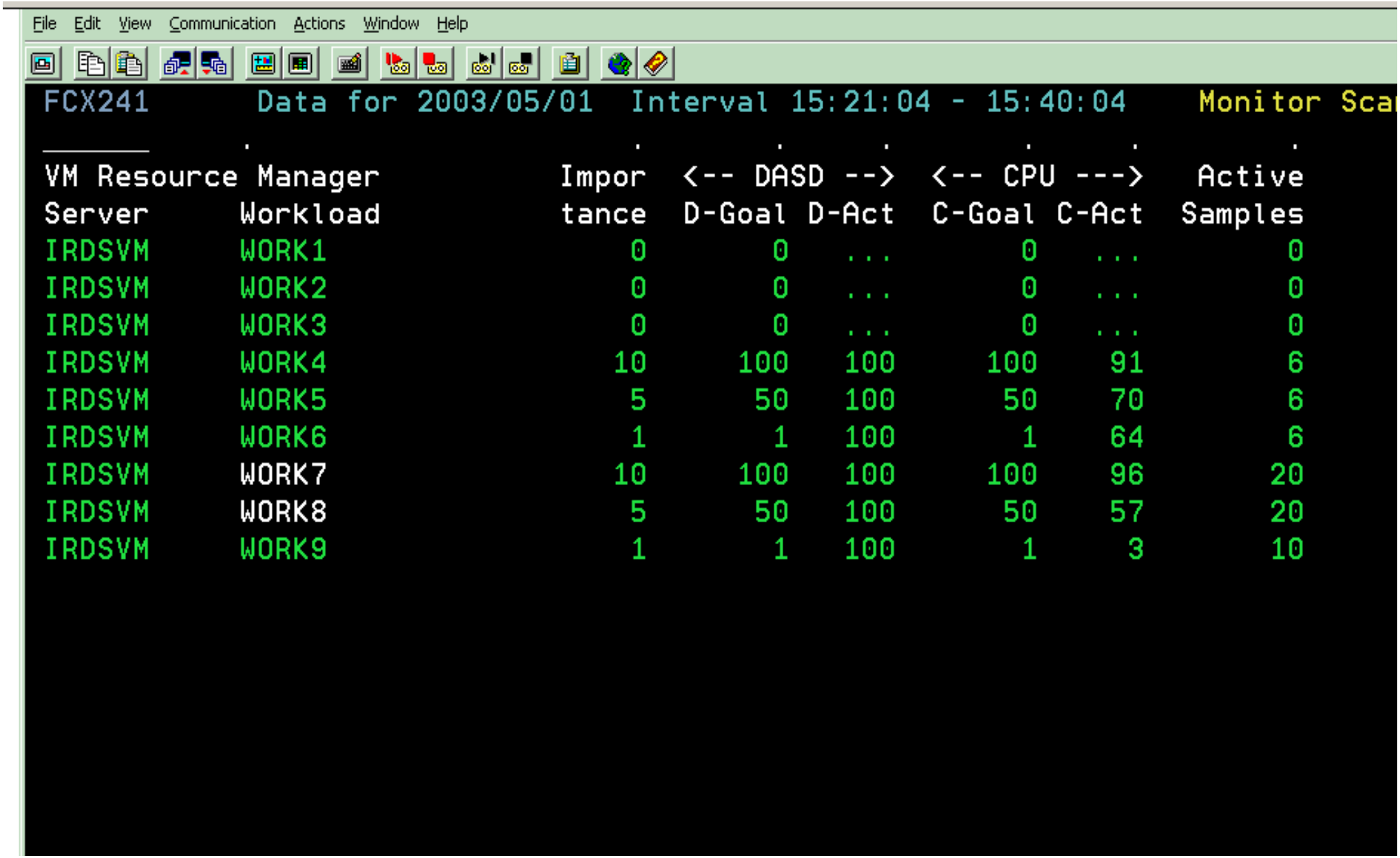
Monitor Data

- Monitor records updated
  - User Domain - User Activity Data - D4R3
    - Relative or absolute I/O priority
    - requested and effective priority range
    - Number of times DASD I/O requests have been outprioritized
  - System Domain - User Data - D0R8
    - I/O Priority Queuing Active flag
    - High & low values available to CP
- New Monitor record (in 4.3.0)
  - Scheduler Domain - I/O Priority Queuing Changes - D2R11
    - Event record when I/O priority queuing values change for a user
      - SET IOPRIORITY command
      - Range available to CP changes

Monitor Data

- VMRM Application Monitor Data (APPLDATA) defined in z/VM 4.4.0
  - Shows workloads, goals, and actual workload achievements
  - Performance Toolkit for VM is enhanced to interpret this data
    - detects when a new configuration file is put into production and refreshes data accordingly
  - Documented in the z/VM Performance publication - Appendix G

Performance Toolkit Screen with VMRM data



Future Enhancements

- Collaborative Memory Management
  - Prototype being developed that may be offered as a Beta on 5.1.0
  - A collaboration between VM and Linux to optimize memory management
  - System Admin identifies guests in the VMRM configuration file to be notified, treated with equal priority
  - VMRM tracks system memory utilization/demand and computes target "resident footprint" for each guest
  - VMRM sends SMSG to guests to adjust footprint
  - Guest device driver receives messages
    - uses existing guest logic to return the least valuable pages

Other Potential Enhancements?

- Network management...
- Customer requirements ... we welcome your feedback!
  - Other workload goals you wish to see managed ?

Contact Info: wilkinss@us.ibm.com or caseyct@us.ibm.com

Documentation: z/VM Performance, SC24-6109-00