Your Changing z/OS Performance Management World: New Workloads, New Skills

Summer SHARE
August 2015
Session 17642

Agenda

- The new world of RMF monitoring
  - RMF XP and z/OSMF

- z Systems hardware functions
  - z13 Simultaneous Multithreading (SMT) RMF data
  - When your LPAR runs at 100%

- z/OS in the new world of cloud, mobile and analytics
  - z/OS Connect
  - IDAA and WLM
RMF Product Overview and RMF XP

RMF Postprocessor
Historical Reporting, Analysis and Planning

RMF Monitor II and III
Real-Time Reporting, Problem Determination

RMF Monitor I
RMF Data Gatherer

RMF Distributed Dataserver (DDS) & RMF XP (GPM4CIM)

RMF Sysplex Data Server and APIs

CIM Client APIs

RMF Data Portal for z/OS
RMF Spreadsheet Reporter

RMF Performance Data Portal

AIX, Windows & Linux CIM Provider

RMF XP Enhancements

► RMF XP is the solution for Cross Platform Performance Monitoring
► RMF XP supports the Operating Systems running on
  ► x Blades
  ► p Blades

► In addition RMF XP supports Linux on System z
  ► LPAR Mode
  ► VM Guest Mode
RMF XP – Component Overview

- RMF Monitor III
- RMF Distributed Data Server
- RMF Sysplex Data Server
- RMF Generic CIM Client
- GPM4CIM
- RMF Monitor III
- RMF Distributed Data Server
- RMF Generic CIM Client
- 2/OSMF Resource Monitoring

RMF XP Windows Support – Resource Model
RMF XP Windows Support – z/OSMF Resource Monitoring

Health Check for all Windows Systems in the Enterprise:
- Processor
- Memory
- Filesystem
- Network
RMF XP & SMF Records

RMF XP can be configured to write SMF records at interval end

<table>
<thead>
<tr>
<th>AIX on System p</th>
<th>ST</th>
<th>Linux on System x</th>
<th>ST</th>
<th>Linux on System z</th>
<th>ST</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX_ActiveMemoryExpansion</td>
<td>1</td>
<td>Linux_IPProtocolEndpoint</td>
<td>20</td>
<td>Linux_IPProtocolEndpoint</td>
<td>40</td>
</tr>
<tr>
<td>AIX_Processor</td>
<td>2</td>
<td>Linux_LocalFileSystem</td>
<td>21</td>
<td>Linux_LocalFileSystem</td>
<td>41</td>
</tr>
<tr>
<td>AIX_ComputerSystem</td>
<td>3</td>
<td>Linux_NetworkPort</td>
<td>22</td>
<td>Linux_NetworkPort</td>
<td>42</td>
</tr>
<tr>
<td>AIX_Disk</td>
<td>4</td>
<td>Linux_OperatingSystem</td>
<td>23</td>
<td>Linux_OperatingSystem</td>
<td>43</td>
</tr>
<tr>
<td>AIX_NetworkPort</td>
<td>5</td>
<td>Linux_Processor</td>
<td>24</td>
<td>Linux_Processor</td>
<td>44</td>
</tr>
<tr>
<td>AIX_FileSystem</td>
<td>6</td>
<td>Linux_UnixProcess</td>
<td>25</td>
<td>Linux_UnixProcess</td>
<td>45</td>
</tr>
<tr>
<td>AIX_Memory</td>
<td>7</td>
<td>Linux_Storage</td>
<td>26</td>
<td>Linux_Storage</td>
<td>46</td>
</tr>
<tr>
<td>AIX_OperatingSystem</td>
<td>8</td>
<td>Linux_KVM</td>
<td>30</td>
<td>Linux_zCEC</td>
<td>50</td>
</tr>
<tr>
<td>AIX_Process</td>
<td>9</td>
<td>Linux_Xen</td>
<td>31</td>
<td>Linux_zLPAR</td>
<td>51</td>
</tr>
<tr>
<td>AIX_SharedEthernetAdapter</td>
<td>10</td>
<td>Linux_zChannel</td>
<td></td>
<td>Linux_zChannel</td>
<td>52</td>
</tr>
<tr>
<td>AIX_ActiveMemorySharing</td>
<td>11</td>
<td>Linux_zCEKD</td>
<td></td>
<td>Linux_zCEKD</td>
<td>53</td>
</tr>
<tr>
<td>AIX_VirtualTargetDevice</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

One Subtype per Metric Category
“Simultaneous multithreading (SMT) permits multiple independent threads of execution to better utilize the resources provided by modern processor architectures.”

With z13, SMT allows up to two instructions per core to run simultaneously to get better overall throughput.

SMT is designed to make better use of processors.

On z/OS, SMT is available for zIIP processing:
- Two concurrent threads are available per core and can be turned on or off
- Capacity (throughput) usually increases
- Performance may in some cases be superior using single threading

*Note: Speed limit signs for illustration only
z13 - SMT Exploitation

- SMT Aware OS informs PR/SM that it intends to exploit SMT
  - PR/SM can dispatch any OS core to any physical core
  - OS controls the whole core – must follow rules
    - Maximize core throughput (Drive cores with high Thread Density [2])
    - Maximize core availability (Meet workload goals using fewest cores)
- SMT is transparent to applications
- LOADxx and IEAOPTxx parmlib options to enable SMT on z/OS:
  - LOADxx: PROCVIEW CORE|CPU
  - IEAOPTxx: MT_ZIIP_MODE={1 | 2}

z13 – z/OS SMT Metrics

- Capacity Factor (CF)
  - How much work core actually completes for a given workload mix at current utilization - relative to single thread
  - MT-1 Capacity Factor is 1.0 (100%)
  - MT-2 Capacity Factor is workload dependent
- Maximum Capacity Factor (mCF)
  - How much work a core can complete for a given workload mix at most
- Core Busy Time
  - Time any thread on the core is executing instructions when core is dispatched to physical core
- Average Thread Density
  - Average number of executing threads during Core Busy Time (Range: 1.0 - 2.0)
- Productivity
  - Core Busy Time Utilization (percentage of used capacity) for a given workload mix
  - Productivity represents capacity in use (CF) relative to capacity total (mCF) during Core Busy Time.
- Core Utilization
  - Capacity in use relative to capacity total over some time interval
  - Calculated as Core Busy Time x Productivity
**z13 – SMT: Postprocessor CPU Activity Report**

- PP CPU activity report displayed in "old" format when SMT is active
- PP CPU activity report provides new metrics when SMT is active
  - MT Productivity and Utilization of each logical core
  - MT Multi-Threading Analysis section displays MT Mode, MT Capacity Factors and average Thread Density
- One data line in PP CPU activity report represents one thread (CPU)
  - CPU NUM designates the logical core
- Some metrics like TIME % ONLINE and LPAR BUSY provided at core granularity only

---

**z/OS V2R1**

**SYSTEM ID CB8B**

**DATE 02/02/2015**

**INTERVAL 15.00.004**

---

**CPU ACTIVITY**

<table>
<thead>
<tr>
<th>NUM</th>
<th>TYPE</th>
<th>ONLINE</th>
<th>LPAR BUSY</th>
<th>PIL BUSY</th>
<th>PARKED</th>
<th>PROD</th>
<th>UTIL</th>
<th>SHARE %</th>
<th>RATE</th>
<th>% VIA TPI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>CP</td>
<td>100.00</td>
<td>68.07</td>
<td>67.94</td>
<td>0.00</td>
<td>100.0</td>
<td>68.07</td>
<td>100.0 %</td>
<td>370.1</td>
<td>13.90</td>
</tr>
<tr>
<td>1</td>
<td>CP</td>
<td>100.00</td>
<td>46.78</td>
<td>46.78</td>
<td>0.00</td>
<td>100.0</td>
<td>46.78</td>
<td>52.9</td>
<td>5.29</td>
<td>16.93</td>
</tr>
</tbody>
</table>

---

**TOTAL/AVERAGE**

<table>
<thead>
<tr>
<th></th>
<th>PROD</th>
<th>UTIL</th>
<th>SHARE %</th>
<th>RATE</th>
<th>% VIA TPI</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**MULTI-THREADING ANALYSIS**

<table>
<thead>
<tr>
<th>CPU TYPE</th>
<th>MODE</th>
<th>MAX CF</th>
<th>CF AVG</th>
<th>TD AVG</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP</td>
<td>1</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>IIP</td>
<td>2</td>
<td>1.485</td>
<td>1.279</td>
<td>1.576</td>
</tr>
</tbody>
</table>

---

**z13 – SMT: Postprocessor Workload Activity Report**

**WORKLOAD ACTIVITY**

<table>
<thead>
<tr>
<th>z/OS V2R1</th>
<th>SYSPLEX UTCPLXCB</th>
<th>DATE 02/02/2015</th>
<th>INTERVAL 15.00.004</th>
<th>MODE = GOAL</th>
</tr>
</thead>
</table>

**REPORT BY:** POLICY=BASEPOL

- **TRANSACTION**: TRAN-SHARE
  - AVG 790.12
  - MPP 790.12
  - ENDD 9173
  - #SWAPS 10.19
  - EXCDT 15866
  - REM ENC 0.00
  - MS ENC 0.00

---

**MT-1 equivalent service units and service times**

**MT-1 equivalent service units and service times**

**Percentage of maximum core capacity used**

- Pre SMT:
  - Service time: Logical processor CPU time
  - APPL%: Percentage of logical processor capacity

- SMT mode active:
  - Service time: MT-1 equivalent CPU time
  - APPL%: Percentage of maximum core capacity calculated as

  $\text{APPL\%} = \frac{\text{MT-1 equivalent CPU time}}{\text{Interval Length} \times \text{mCF}} \times 100$

For MT Mode = 1 $\Rightarrow$ mCF = 1
Running at or near 100% - Blocked Workload Support

- **Problem**
  - Work competes for resources, serialized by locks and latches
    - Low import work may hold a resource and high important work may have to wait for it

- **WLM Blocked Workload Support**
  - Recognizes blocked work
    - Work which doesn't show any progress for an elongated period of time
  - Allows this work to use a small amount of CPU periodically
    - With the hope to resolve existing (potential) resource contentions

**Blocked Workload Support: User Interface: IEAOPT**

<table>
<thead>
<tr>
<th>BLWLTRPCT</th>
<th>Percentage of the CPU capacity of the LPAR to be used for promotion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Specified in units of 0.1%</td>
</tr>
<tr>
<td></td>
<td>Default is 5 (=0.5%)</td>
</tr>
<tr>
<td></td>
<td>Maximum is 200 (=20%)</td>
</tr>
<tr>
<td></td>
<td>Would only be spent when enough units of work exist which need promotion</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BLWLINTHD</th>
<th>Specifies threshold time interval for which a blocked address space or enclave must wait before being considered for promotion.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Minimum is 5 seconds. Maximum is 65535 seconds.</td>
</tr>
<tr>
<td></td>
<td>Default is 20 seconds.</td>
</tr>
</tbody>
</table>
Blocked Workload Support: RMF

- Extensions of RMF Postprocessor CPU Activity and WLMGL reports with information about blocked workloads and the temporary promotion of their dispatching priority
- SMF record 70-1 (CPU activity) and SMF 72-3 (Workload activity)

Promoted transactions: RMF workload activity report
Promoted transactions RMF workload activity report

<table>
<thead>
<tr>
<th>SERVICE TIME</th>
<th>---</th>
<th>APPL %</th>
<th>--PROMOTED--</th>
<th>----</th>
<th>STORAGE ----</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>805.697</td>
<td>CP 92.24</td>
<td>BLK 1.489</td>
<td>AVG 1195.43</td>
<td></td>
</tr>
<tr>
<td>SRB</td>
<td>13.850</td>
<td>AAPC 0.00</td>
<td>ENQ 0.046</td>
<td>TOTAL 182122.4</td>
<td></td>
</tr>
<tr>
<td>RCT</td>
<td>9.995</td>
<td>IIPCP 0.00</td>
<td>CRM 5.593</td>
<td>SHARED 230.59</td>
<td></td>
</tr>
<tr>
<td>IIT</td>
<td>0.576</td>
<td>LCK 0.000</td>
<td>SUP 0.000</td>
<td>- PAGE-IN RATES-</td>
<td></td>
</tr>
<tr>
<td>HST</td>
<td>0.000</td>
<td>AAP 0.00</td>
<td>SUP 0.000</td>
<td>SINGLE 0.0</td>
<td></td>
</tr>
<tr>
<td>AAP</td>
<td>0.000</td>
<td>IIP 0.00</td>
<td>SUP 0.000</td>
<td>BLOCK 0.0</td>
<td></td>
</tr>
<tr>
<td>IIP</td>
<td>0.000</td>
<td>IIP 0.00</td>
<td>SUP 0.000</td>
<td>SHARED 0.0</td>
<td></td>
</tr>
</tbody>
</table>

RVED-----------------------------------------------

Promoted transactions RMF field definitions

CPU time in seconds that transactions in this group were running at a promoted dispatching priority, separated by the reason for the promotion:

**BLK** CPU time in seconds consumed while the dispatching priority of work with low importance was temporarily raised to help blocked workloads

**ENQ** CPU time in seconds consumed while the dispatching priority was temporarily raised by enqueue management because the work held a resource that other work needed.

**CRM** CPU time in seconds consumed while the dispatching priority was temporarily raised by chronic resource contention management because the work held a resource that other work needed.

**LCK** In HiperDispatch mode, the CPU time in seconds consumed while the dispatching priority was temporarily raised to shorten the lock hold time of a local suspend lock held by the work unit.

**SUP** CPU time in seconds consumed while the dispatching priority for a work unit was temporarily raised by the z/OS supervisor to a higher dispatching priority than assigned by WLM.
### System Address Space and Work Unit Analysis

<table>
<thead>
<tr>
<th>Queue Types</th>
<th>Min</th>
<th>Max</th>
<th>Avg</th>
<th>Number of</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN</td>
<td>550</td>
<td>1,008</td>
<td>594.8</td>
<td>65.5</td>
</tr>
<tr>
<td>IN READY</td>
<td>4</td>
<td>438</td>
<td>22.7</td>
<td>&lt;= N + 1</td>
</tr>
<tr>
<td>OUT READY</td>
<td>0</td>
<td>1</td>
<td>0.0</td>
<td>&lt;= N + 2</td>
</tr>
<tr>
<td>OUT WAIT</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>&lt;= N + 3</td>
</tr>
<tr>
<td>LOGICAL OUT RDY</td>
<td>0</td>
<td>628</td>
<td>10.3</td>
<td>&lt;= N + 4</td>
</tr>
<tr>
<td>LOGICAL OUT WAIT</td>
<td>178</td>
<td>634</td>
<td>588.4</td>
<td>&lt;= N + 5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address Space Types</th>
<th>Min</th>
<th>Max</th>
<th>Avg</th>
<th>Number of</th>
</tr>
</thead>
<tbody>
<tr>
<td>BATCH</td>
<td>281</td>
<td>284</td>
<td>282.0</td>
<td>&lt;= N + 6</td>
</tr>
<tr>
<td>STC</td>
<td>708</td>
<td>763</td>
<td>736.4</td>
<td>&lt;= N + 7</td>
</tr>
<tr>
<td>TSO</td>
<td>97</td>
<td>98</td>
<td>97.9</td>
<td>&lt;= N + 8</td>
</tr>
<tr>
<td>ASCH</td>
<td>0</td>
<td>1</td>
<td>0.0</td>
<td>&lt;= N + 9</td>
</tr>
<tr>
<td>OMVS</td>
<td>43</td>
<td>97</td>
<td>68.0</td>
<td>&lt;= N + 10</td>
</tr>
</tbody>
</table>

### CPU Types

<table>
<thead>
<tr>
<th>CPU</th>
<th>Min</th>
<th>Max</th>
<th>Avg</th>
<th>Number of</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP</td>
<td>444</td>
<td>888</td>
<td>555.5</td>
<td>&lt;= N + 11</td>
</tr>
<tr>
<td>AAP</td>
<td>22</td>
<td>33</td>
<td>28.8</td>
<td>&lt;= N + 12</td>
</tr>
<tr>
<td>IIP</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>&lt;= N + 13</td>
</tr>
</tbody>
</table>

---

**Work unit queue distribution: Mon I CPU report**

**Systems of Engagement and Systems of Record**

- **Focus on point of Interaction with the SOR**
  - Not just mobile phones ... any system or device ... including mainframe programs
  - This can be on System z, and in fact System z can make an excellent platform for SOE
  - This doesn't have to be System z, but a great deal of SOR data is on the platform
Mainframe as a Service

Another use-case for z/OS Connect is as a standard gateway into the z/OS LPAR to expose programs as a service:

z/OS Connect provides a way to do this with a single entry point (HA is possible) and common protocol (REST/JSON)

What is z/OS Connect?

1. z/OS Connect is a software function that runs in Liberty Profile for z/OS.
2. z/OS Connect is described and configured in the Liberty server.xml file.
3. z/OS Connect is designed to accept RESTful URIs with JSON data payloads.
4. One part of z/OS Connect is a servlet that runs in Liberty Profile z/OS.
5. A ‘Service Provider’ is software that provides the connectivity to the backend system.
6. z/OS Connect provides the ability to transform JSON to the layout required by backend.
7. ‘Interceptors’ are callout points where software can be invoked to do things such as SAF authorization and SMF activity recording.
8. Initially the backend systems supported will be CICS, IMS and Batch.
Context Within Overall Mobile Architecture

The message here is that z/OS Connect is a piece of the Mobile architecture, but in most cases will not be the only component:

Users of z/OS Connect would access through normal corporate firewall infrastructure.

IBM MobileFirst Platform to provide application management, security and operational governance for mobile applications.

z/OS Connect would be behind the secure firewall, and on LPARs along with backend systems.

Liberty Profile z/OS

Liberty Profile is IBM’s dynamic and composable server runtime. First shipped with Version 8.5, it is available on many platforms, including z/OS:

- Single JVM per server model
  As opposed to the multiple JVM model of traditional WAS z/OS (the CR/SR model)

- Simple configuration structure
  One XML file serves as the main configuration file

- Dynamic
  Changes to the configuration file or to the applications are detected and dynamically loaded

- Composable
  You tell Liberty Profile what features and functions you want and only that code is loaded

- On z/OS can run from UNIX shell or as a z/OS started task
  On z/OS we anticipate most will run as started task

Liberty Profile is the basis for z/OS Connect, so any discussion of z/OS Connect necessarily involves Liberty.
Audit (SMF) Interceptor

The audit interceptor writes SMF 120.11 records with the following information captured:

Liberty Profile z/OS

- System Name
- Sysplex Name
- Jobname
- Job Prefix
- Address Space Stoken

z/OS Connect

- System Name
- Sysplex Name
- Jobname
- Job Prefix
- Address Space Stoken

Server Identification Section

- Arrival Time
- Completion Time
- Target URI
- Input JSON Length
- Response JSON Length
- Method Name
- Service Name
- Userid

z/OS Connect

User Data Section

System z mobile pricing model

- Reduce z/OS peak MSUs attributable to mobile workloads -- up to 60%
- No Infrastructure Changes Required... (such as separate LPARs)

Current z/OS Peak Utilization

Mobile Workloads CPU

Other z/OS Workloads CPU

Adjusted z/OS Peak Utilization

Mobile

Other Workloads

Reduced cost of growth on Mobile workloads

- Customers must tag and track z/OS CPU seconds from mobile workloads.
- New MWRT tool replaces SCRT and will subtract mobile CPU seconds from peaks.
Mobile Workload Pricing helps alleviate spikes caused by increased mobile usage...

1. Measure LPAR peak with standard methodology
2. Measure CICS usage in new IBM reporting tool
3. Capture CICS transaction details and filter by transaction type, mobile or not
4. Subtract 60% of mobile usage: \[-60\% \times 200\]
5. Adjust LPAR peak with new reporting tool
6. Adjusted LPAR peak for month Pricing & billing BAU based on peak

Mobile Workload Pricing helps alleviate spikes caused by increased mobile usage...

Which tracking mechanism can be used for MWP?

1. Use an Individual LPAR for mobile-only workloads
2. Use Individual Regions for mobile-only workloads
3. Use Same Regions for mobile and non-mobile workloads

*Figures are for illustrative purposes only. Tracking processes and records will vary by customer.*

(c) 2014 IBM Corporation
The WLM Enclave

An "enclave" is a way to identify and manage individual pieces of work within the many parts of a running z/OS system.

Key points from this chart:
- An "enclave" is simply a way for WLM to understand priorities at a work unit level.
- WAS does this automatically... if you do no other configuration I'll still do this with default values.

Assigning a Service Class to the Enclave

Subsystem Type CB - WebSphere z/OS CN and TC Classifications

<table>
<thead>
<tr>
<th>Qualifier</th>
<th>Qualifier</th>
<th>Starting</th>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td># type</td>
<td>name</td>
<td>position</td>
<td>Class</td>
<td>Class</td>
</tr>
<tr>
<td>1 CN</td>
<td>DFDMSR*</td>
<td>1</td>
<td>CBCLASS</td>
<td>DFDMSR</td>
</tr>
<tr>
<td>1 CN</td>
<td>DFSR01*</td>
<td></td>
<td>CBCLASS</td>
<td>DFSR01</td>
</tr>
<tr>
<td>2 TC</td>
<td>DFTRAN1</td>
<td>2</td>
<td>DFTRAN1</td>
<td>DFSR01T</td>
</tr>
<tr>
<td>2 TC</td>
<td>DFTRAN2</td>
<td>3</td>
<td>DFTRAN2</td>
<td>DFSR01T</td>
</tr>
<tr>
<td>1 TC</td>
<td>DFTRAN3</td>
<td>4</td>
<td>DFTRAN3</td>
<td>DFTRAN3</td>
</tr>
</tbody>
</table>

Enclaves created in WAS CR are classified by rules in CB subsystem type:
1. CN of DFDMSR* matches the Deployment Manager. Work there goes to CBCLASS.
2. Work in DFSR01* cluster without a transaction classification gets CBCLASS as well.
3. Work in DFSR01* cluster with TC of DFTRAN1 or DFTRAN2 get service classes as shown.
4. Work that matches the TC of DFTRAN3 regardless of WAS CN gets service class DFTRAN3.
5. Anything that doesn't match any specific rules gets the default service class of CBDEFAULT.
The transaction class name file

The file supplies a set of criteria to match requests to transaction class names, which then match with rules in the CB subsystem type

The WAS for z/OS Liberty profile?

- The WAS for z/OS Liberty profile is Liberty with optional, independently enabled extensions that exploit z/OS facilities
  - Only enable exploitation of z/OS features you need
  - Only configure the z/OS functions you use
- Focus of v8.5 is basic integration and exploitation
Feature – z/OS Workload Manager

- Adds support to classify HTTP requests with z/OS WLM
  - Classification associates response time goals and importance to work run in WebSphere
  - z/OS workload manager will manage the resources available on the system in a way that ensures the most important work runs while attempting to meet response time goals
  - RMF reports provide information about completed transactions, response times, etc. by service class

```xml
<server description="z/OSworkloadManagement">
  <featureManager>
    <feature zoswlm="1.0"></feature>
  </featureManager>
  <wlmClassification transactionClass="WLTRADE" resource="/tradelite/" />
  <wlmClassification transactionClass="WLDBFL" />
</wlmClassification>
</server>
```

IBM DB2 Analytics Accelerator
WLM and IDAA Interaction

- DB2 detects WLM service class and importance level and sends it to the accelerator with each query.
- The accelerator maps the importance level to a Netezza priority and alters the session prior to the query execution, using the corresponding priority. Also, threads scheduled will have their priorities adjusted.

<table>
<thead>
<tr>
<th>WLM Importance Level</th>
<th>Netezza Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Version 3</td>
</tr>
<tr>
<td>System</td>
<td>Critical</td>
</tr>
<tr>
<td>Importance 1</td>
<td>Critical</td>
</tr>
<tr>
<td>Importance 2</td>
<td>High</td>
</tr>
<tr>
<td>Importance 3</td>
<td>Normal</td>
</tr>
<tr>
<td>Importance 4</td>
<td>Normal</td>
</tr>
<tr>
<td>Importance 5</td>
<td>Normal</td>
</tr>
<tr>
<td>Discretionary</td>
<td>Low</td>
</tr>
</tbody>
</table>

Summary

- The new world of RMF monitoring
  - RMF XP and z/OSMF
- z Systems hardware functions
  - z13 Simultaneous Multithreading (SMT) RMF data
  - When your LPAR runs at 100%
- z/OS in the new world of cloud, mobile and analytics
  - z/OS Connect
  - IDAA and WLM