IBM Education Assistance for z/OS V2R1

Item: Improved Accuracy for IWMEQTME
Element/Component: WLM/SRM

Material is current as of March 2013

Agenda

- Trademarks
- Presentation Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Migration & Coexistence Considerations
- Installation
- Presentation Summary
- Appendix

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.

Presentation Objectives

- With z/OS V2R1, WLM/SRM introduces a new parameter on the IWMEQTME service
- This presentation explains:
  - The purpose of the new parameter
  - How it is used

Overview – Problem Statement

- The IWMEQTME service returns the enclave processor times if the current dispatchable work unit invoking the service is associated with an enclave
- However, the values returned by IWMEQTME are not absolutely exact
- The enclave processor times returned by the service are the total accumulated times for the enclave up to the point when it was dispatched for the last time. All processor times since then are missing

Overview – Solution

- The IWMEQTME service is extended to call the dispatcher before the enclave processor times are fetched from the enclave control block
- This ensures that the enclave processor times are as accurate as possible
- Because calling the dispatcher is costly, the extension is controlled using an additional parameter
- The default is not to call the dispatcher, in order to avoid performance degradation for other exploiters
- Fundamental limitations to the accuracy of the IWMEQTME service:
  - The enclave processor times cannot be absolutely exact, because the IWMEQTME service itself is running in the enclave it is trying to observe. So it consumes CPU time for the enclave after having fetched the values from the enclave control block, which distorts the results slightly
  - For enclaves with multiple work units (tasks or SRBs) joined in parallel, the benefit is marginal, because for none of the other work units the current dispatch times are counted

Overview – Benefit/Value

- Callers get more accurate CPU values for the enclaves they use
- For example, for charge-back/capacity planning for zIIP processors based on accurately reported CPU usage

Usage & Invocation – New Syntax of IWMEQTME

```
>>__ ______ __IWMEQTME__ _________________ _____________________>
    |_name_|            |_CPUTIME=cputime_|
....
>__ ____________________________ __ _________________ __________>
   |_,ZAAPONCPTIME=zaaponcptime_|  |_,ZAAPNFACTOR=zaapnfactor_|

    _,CURRENT_DISP=NO__
>__|___________________|________________________________________>
   |_,CURRENT_DISP=YES_|

>__ __________________ __ __________________ ___________________>
   |_,RETCODE=retcode_|  |_,RSNCODE=rsncode_|

    _,PLISTVER=IMPLIED_VERSION_
>__|___________________________|________________________________>
   |_,PLISTVER=MAX_____________|
   |_,PLISTVER=0_______________|
   |_,PLISTVER=1_______________|
   |_,PLISTVER=2_______________|
....
```

Usage & Invocation – New Requirements for the Caller

- Dispatchable unit mode
  - Task if CURRENT_DISP=YES
  - Task or SRB otherwise
- Invoking IWMEQTME with parameter CURRENT_DISP=YES while being in SRB mode results in ABEND 05D-10
- Interrupt status
  - Enabled for I/O and external interrupts if CURRENT_DISP=YES
  - Otherwise enabled or disabled for I/O and external interrupts
- Invoking IWMEQTME with parameter CURRENT_DISP=YES while being disabled for I/O and external interrupts results in ABEND 05D-08
- The caller is responsible for error recovery when invoking the IWMEQTME service with the new parameter in wrong mode or interrupt status

Usage & Invocation – Documentation (I)

- CURRENT_DISP is an optional keyword input indicating whether to call the dispatcher before querying the enclave processor times. Enclave processor times are only accumulated when an enclave loses the processor. So the processor times are the total accumulated times for the enclave up to the point when it was dispatched for the last time. Because the enclave being queried by the service is currently active on a processor, the processor times are not accurate; all processor times since the last dispatch of the enclave are missing. To make the enclave processor times returned by the service more accurate, the dispatcher can optionally be called to update the accumulated processor times before they are returned by the service. This option provides benefit mostly for single-work-unit enclaves. For enclaves with multiple work units (tasks or SRBs) in parallel, the benefit will be marginal because the call to the dispatcher will only update the times for the current (calling) workunit.
- DEFAULT: NO

Usage & Invocation – Documentation (II)

- CURRENT_DISP=NO
  The enclave processor times are queried without calling the dispatcher before. Which means that processor times since the last dispatch of the enclave are missing, leading to slightly too small results. In return invocation of the service is less costly and has less prerequisites in the environment.

- CURRENT_DISP=YES
  The enclave processor times are queried after calling calling the dispatcher. Which means that all processor times up to invoking the service are returned. The price for getting more exact values is more cycles for invoking the service, and more restricted environments in which it can be invoked.

Usage & Invocation – Example

```
IWMEQTME
     CPUTIME=CPU_TIME,
     ZIIPTIME=ZIIP_TIME,
     ZAAPTIME=ZAAP_TIME,
     CURRENT_DISP=YES,
     RETCODE=RC,
     RSNCODE=RSN

*
* Storage areas
*
CPU_TIME DS    CL8            CPU time for the enclave
ZIIPTIME DS    CL8            ZIIP time for the enclave
ZAAPTIME DS    CL8            ZAAP time for the enclave
RC       DS    F              Return code
RSN      DS    F              Reason code
```

Interactions & Dependencies

- Software Dependencies
  - None
- Hardware Dependencies
  - None
- Exploiters
  - WebSphere Application Server for z/OS

Migration & Coexistence Considerations

- None – If the service is invoked on previous releases with the new parameter specified, the new parameter is ignored

Installation

- Support for this line item is packaged with, and installed as part of z/OS V2R1
- No rollbacks

Presentation Summary

- This session explained the purpose and use of the new parameter CURRENT_DISP on the EQTME service
- By specifying the new parameter, callers get more accurate CPU values for the enclaves they use, for example, for charge-back/capacity planning for zIIP processors based on accurately reported CPU usage
- Specifying the new parameter increases runtime of the service, so increased accuracy comes at increased cost
- Furthermore, the new parameter can only be specified in task mode and enabled interrupt status. The caller is responsible for error recovery when invoking the IWMEQTME service with the new parameter in wrong mode or interrupt status

Appendix

- Publications:
  - z/OS V2R1 MVS Programming: Workload Management Services (SC34-2663-00)
    - Chapter IWMEQTME - Querying Enclave CPU Time