IBM Education Assistance for z/OS V2R2

Item: Pause Multiple Elements
Element/Component: BCP Supervisor

Material current as of May 2015

Agenda

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

Trademarks

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

Presentation Objectives

- Understand the new IEAVPME2/IEA4PME2 Pause Multiple Elements service

Overview

- Problem Statement / Need Addressed
  - Wait/Post can run into performance issues due to local lock contention. Pause/Release was specifically designed to avoid that issue and can be used instead of Wait/Post in many situations. However, there has been no Pause/Release solution to replace Wait with a list of ECBs
- Solution
  - The new IEAVPME2 'Pause Multiple Elements' service allows a user to specify a list of PETs to Pause upon, along with how many of those PETs must be Released for the Pause to be satisfied
  - IEA4PME2 is the 64-bit version of IEAVPME2
- Benefit / Value
  - Allows applications which need to wait upon more than one event to use Pause/Release, avoiding local lock contention
  - Applications which use Pause multiple will also gain the improved security/Integrity characteristics of Pause/Release (over Wait/Post)

Usage & Invocation – IEAVPME2

- **IEAVPME2** is similar to the existing IEAVPSE2 'Pause single' service, with the same authorization requirements, etc. Unique parameters:
  - **pause_element_token_list** specifies a list of PETs to Pause upon
  - **updated_pause_element_token_list** specifies where updated versions of the PETs are to be placed after the Pause multiple has been Released
  - **release_code_list** specifies where release codes for the PETs are to be placed after a successful Pause multiple and Release. When the overall return code is 5, 9, D, 21, 35, 3D, or 41, individual return codes (not release codes) for each PET are returned here. These are the same return codes used by Pause single (eg: 4, 8, …) for PET problems with the low-order bit added to signify that they have been placed into the release_code_list
  - **number_of_PETs_in_each_list**
  - **number_of_PEs_to_release** specifies the minimum number of PEs that must be Released to satisfy the Pause
  - **return_code** contains either zero for a successful Pause/Release or the highest return code for the list of PETs if a problem was found
  - **workarea** points to a 216-byte area where we save the caller's 64-bit registers and ARs. We do not require R13 to point to a savearea

Usage & Invocation – Example of using IEAVPME2

In this example, 4 PETs were previously obtained and placed contiguously into MyPets. When any three have been Released, control will be returned to the caller and UPetList will contain the updated PETs. For the PETs which were released, RlsCList will contain x'01' in the high-order byte and the release code in the three low-order bytes. For the PET which was not released, RlsCList will contain zero.

```
         CALL IEAVPME2,(RC,MyPets,UPetList,RlsCList,                   *
               NumPets,NumToRel,IEA_LINKAGE_SVC,WorkArea)
...
kNumPets EQU 4                Number of PETs (total)
MyPets   DS (kNumPets*4)F     PETs to Pause upon
UPetList DS (kNumPets*4)F     Updated PETs after Pause
RlsCList DS (kNumPets)F       Release code list
NumPets  DC A(kNumPets)       Number of PETs to Pause upon
NumToRel DC F'3'              Number of PETs required to be Released
RC       DS 1F                Return code
WorkArea DS 27D               Work Area for Pause service
```

Usage & Invocation – IEA4PME2

- **IEA4PME2** is the 64-bit version of **IEAVPME2**, with the same parameters and return codes
- Addresses in the parameter list are 64-bits long

Migration & Coexistence Considerations

- No Migration considerations, no Coexistence considerations

- Pause/Release services are invoked via callable services stubs IEACSS or IEA4CSS. These stubs may be linked into an executable load module or Loaded and called using the names of the various Pause/Release services as aliases
  - The new versions of these stubs check for CVTPAUS5 (equated to CVTZOS_V2R2) to ensure that Pause multiple is actually available on the system they are running on. If not, they return the existing IEA_UNSUPPORTED_MVS_RELEASE return code
  - If someone tries to invoke IEAVPME2 or IEA4PME2 using the old versions of these stubs, the name of the service will not be recognized and either the linkedit or the call will fail

Presentation Summary

- IEAVPME2 and IEA4PME2 are the new Pause Multiple Elements service
- These new services are similar to the Pause single service, except that they allow a user to Pause on multiple PETs

Appendix

- Publications:
  - Assembler Services Reference (SA227607)
  - Assembler Services Guide (SA227608)
  - Authorized Assembler Services Reference (SA227610)
  - Authorized Assembler Services Guide (SA227608)
  - Diagnosis Reference (GA227588)