IBM Education Assistance for z/OS V2R1

Item: IPLNIP Miscellany
Element/Component: BCP IPLNIP

Material is current as of March 2013

Agenda

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

Trademarks

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

Presentation Objectives

Describe IPL/NIP changes with respect to

- ARCHLVL 3
- MACHMIG statement of LOADxx

Overview

- Problems:
  - z/OS 2.1 introduces a new architecture level set.
  - When migrating to a new machine, it might be desired to have software avoid exploiting some new function until you're comfortable that the machine is working as needed
- Solution:
  - z/OS 2.1 requires z9 or later hardware
  - MACHMIG LOADxx statement
- Benefit / Value:
  - Better migration

Usage & Invocation (Architecture Levelset)

- z/OS 2.1 requires a machine that at a minimum has the features of a z9
- Wait state 07B-30 may result if you try to IPL on an older machine
- The SYSSTATE macro will support specification of ARCHLVL=3. If you know your program will require z/OS 2.1, you may specify that. If you use the SYSSTATE OSREL parameter to identify your minimum z/OS release, you may also specify ARCHLVL(OSREL) to indicate that the ARCHLVL value should be set according to the mininum archlvl for the release (i.e., ARCHLVL 3 if at least z/OS 2.1, ARCHLVL 2 if at least z/OS 1.6).
  - At this point, there might not be any macros that produce different code based on ARCHLVL=3 but over time there will be, as they can take advantage of the instruction set that corresponds

Usage & Invocation (MACHMIG)

- Migrating to a new machine is often done in stages, first trying to make sure that the machine does the “old stuff” properly and then moving forward to try the exploitation items. If software unconditionally exploits new items whenever it finds that they are available, the customer has no means of getting that software to “do it the old way” if there's some problem (hardware or software) with the “new way”.
- The MACHMIG LOADxx statement is intended to address that. It was introduced with OA38829 in z/OS 1.13 with zEC12 GA.

Usage & Invocation (MACHMIG) (cont)

- The operands supported for z/OS 2.1 are TX and EDAT2. You can specify one or more or the operands. For example, MACHMIG TX,EDAT2
- TX stands for transactional execution facility
- EDAT2 is the enhanced-DAT-facility-2 (i.e., Flash)
- Each operand indicates that z/OS is to “hide” those facilities from properly behaving software. For example, for TX, bits CVTTX and CVTTXC indicate (nominally) that the machine supports TX and the z/OS support is present. An application is not to use TX unless those bits are on. When MACHMIG TX is in effect, those bits are not turned on, so the application “finds” that the function is not available and thus does “something else”.

Interactions & Dependencies

- Software Dependencies
  - –None
- Hardware Dependencies
  - – z/OS 2.1 requires a z9 or newer machine
- Exploiters
  - – None

Migration & Coexistence Considerations

- You may consider using MACHMIG in your migration scenarios, whether running with z/OS 2.1 moving to a zEC12, or running on a zEC12 and moving to z/OS 2.1.

Appendix

- Publications:
  - MVS Initialization and Tuning Reference (LOADxx) SA23-1380-00
  - MVS Assembler Services Reference (SYSSTATE) SA23-1370-00