

IBM Education Assistance for z/OS V2R2

Item: Deferred Class Performance

Element/Component: BCP Contents Supervisor (CSV)



Agenda

- Trademarks
- Presentation Objectives
- Overview
- Usage & Invocation
- Presentation Summary
- Appendix



Trademarks

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



Presentation Objectives

- Understand the support provided in OA43394 (z/OS 2.1 only) and OA45127 (z/OS 1.13 and z/OS 2.1) to enhance the performance of Cobol Version 5 modules



Overview (OA43394)

- Problem Statement / Need Addressed
 - Cobol compiler was changed to use more LE and deferred classes. Performance degradation of Cobol jobs resulted (processing for Cobol deferred classes is identical to that for any language's deferred classes)
- Solution
 - Dynamic area via CPOOL
 - Use user-key rather than MVCDK when source is not fetch-protected
 - Less local lock contention
- Benefit / Value
 - Improved performance for users of deferred classes



Usage & Invocation

- “Writeable static” is the typical use of a deferred class.
- When the module is loaded, a deferred class template is created. LE then asks to instantiate the deferred class
- Optimizations include use of CPOOL for storage, avoiding LOCAL lock in some situations to reduce contention
- Benefits any program object with deferred segment; not limited to Cobol



Overview (OA45127)

- Problem Statement / Need Addressed
 - Cobol Version 5 compiler was changed to use more LE and deferred classes. LLA does not cache a module with a deferred class. Performance degradation of Cobol V5 jobs resulted for customers having LLA manage the Cobol libraries since the PDSE caching of the PDSE does not compare well to the VLF caching of the PDS.
- Solution
 - Allow LLA to manage a program object with a single deferred segment
- Benefit / Value
 - Improved performance for users of deferred classes



Usage & Invocation

- When the COBOL V5 library is in the LNKLIST or has been identified to LLA via the CSVLLAxx parmlib member as a library to be managed
- LLA can utilize VLF to cache a Cobol 5 module that has a deferred segment
- Benefits any program object with deferred segment; not limited to Cobol



Presentation Summary

- Improved performance related to Cobol Version 5 modules



Appendix

- Publications:
 - Init & Tuning Reference (CSVLLAxx parmlib member)

