

z/OS V1R13

DFSMS Open/Close/EOV: RAS and performance

Overview - New option for OCE ABENDs and reason codes

- Problem Statement / Need Addressed
 - OCE and the OPEN or CLOSE access method executors detect hundreds of error conditions that result in issuing the ABEND macro but the ABEND message contains numeric abend code and return codes with little English.
- Solution
 - Provide an installation option via DEVSUPxx to additionally include descriptive text appended to the associated ABEND message, eliminating the need to reference the message manuals to interpret the abend and return codes.
- Benefit / Value
 - Quicker problem determination.

Usage and invocation - New option for OCE ABENDs and reason codes

- Set a new DEVSUPxx PARMLIB keyword to request that the error description be appended to the determinant ABEND message:
OCE_ABEND_DESCRIP={YES | NO}
The default is NO.
- Not all OPEN/CLOSE/EOV messages are affected in this release.
- Example of the output:

```
IEC146I 513-08,IFG0196T,CRTAALL,CRTTSL1,SYSUT2,0920,,DATASET1 036
ERROR DESCRIPTION:
A LABEL VIOLATED THE PUBLISHED STANDARD FOR THAT LABEL, AND THE LABEL
VALIDATION EXIT ISSUED A RETURN CODE REQUESTING OPEN OR EOV TO REJECT
THE VOLUME.
END ERROR DESCRIPTION: IEC146I
```

Overview – RAS, reliability, availability and serviceability

- Problem Statement / Need Addressed
 - OPEN allows the DCB to be in key 9 storage if the caller is running in key 8 but OPEN does not allow the DCBE (DCB extension) to be in key 9 storage.
 - If the user makes a mistake in supplying the DCBE, OPEN ignores it with no diagnostics. For example the data set organization must be PS, DA or PO and the DCBE must not be in use by another DCB.
- Solutions
 - OPEN now allows the DCBE to be in key 9 storage.
 - OPEN will record a code in the SMF 14 or 15 record to show what was wrong with the DCBE.
- Benefit / Value
 - Simpler problem determination

Usage and invocation – RAS, reliability, availability and serviceability

- This byte in the type 8 extended information segment in the SMF type 14 or 15 record explains why the DCBE is invalid:

```
SMF14RAS      EQU *          EXTENDED INFO SEGMENT TYPE 8
SMF14DCBEEXCP EQU X'80'      DCB IS EXCP AND NO FOUNDATION
                                EXTENSION IS PRESENT
SMF14DCBEDSORG EQU X'40'      DSORG IS NOT PS, PO OR DA
SMF14DCBEFREE EQU X'20'      DCBE STORAGE IS NOT ADDRESSABLE
SMF14DCBEKEY  EQU X'10'      DCBE STORAGE IS NOT IN KEY OF CALLER
SMF14DCBEID   EQU X'08'      THE DCBEID IS NOT 'DCBE'
SMF14DCBEMIN  EQU X'04'      DCBE IS NOT AT LEAST THE MINIMUM
                                LENGTH REQUIRED (56 BYTES)
SMF14NODCBE   EQU X'02'      DCBEHIARC FLAGS SET BUT DCBDCBE IS
                                ZEROES
```

Overview – QSAM MULTSDN calculation with concatenation

- Problem Statement / Need Addressed

- The QSAM user can code MULTSDN in the DCBE macro to request OPEN to calculate an optimum BUFNO value for tape and specific types of DASD data sets. OPEN calculates it based on the first data set in the concatenation.
- When going to the next data set in the concatenation an out of storage condition can occur when the next data set's block size is much larger than in the previous data set. This is because the system does not recalculate a smaller number of buffers for the larger blocks.
- Solution
 - Dynamically recalculate the BUFNO value when switching to the next concatenated data set when QSAM with MULTSDN is specified.
- Benefit / Value.
 - The QSAM user that uses MULTSDN is less likely to run out of storage.

Usage and invocation – QSAM MULTSDN calculation with concatenation

- No change needed to how you code MULTSDN on the DCBE macro.
- The system will calculate a number of buffers that more closely matches the logic that is used during OPEN. You are less likely to run out of storage.
- Note that if the number of buffers for the first data set is small due to a large block size, then the current code gets the same number of buffers for the second data set even if it has a small block size that should deserve many buffers.
 - Therefore this case works but is likely to give poor performance.
 - This probable poor performance should be improved by this line item because of the BUFNO recalculation for each data set.

Overview – Free tape volumes at end of volume

- Problem Statement / Need Addressed
 - Long running programs that read or write multivolume tape data sets prevent other jobs from accessing any of the volumes until the job unallocates all of the volumes.
 - An old option allows authorized programs
- Solution
 - A new DD statement keyword, FREEVOL=EOV, allows any program to release tape volumes early before closing the data set.
 - No source code change or special authorization is needed.
- Benefit / Value

Usage and invocation – Free tape volumes at end of volume

- This function is honored only for input processing.
- EOV and CLOSE volume disposition processing will rewind and unload the volume.
- After a volume is released in this manner, OPEN or EOV will issue an abend if the task using the same JCL DD statement attempts to reprocess a previously dequeued volume serial number.
 - The new ABEND and return codes:
 - OPEN: IEC145I with ABEND 413-60
 - EOV: IEC026I with ABEND 637-C0

Overview – XTIO support for subsystem DCBs and ACBs

- Problem Statement / Need Addressed
 - OPEN/CLOSE/EOV XTIO support for BAM DCBs was added in z/OS V1R12 but did not include subsystem DCBs. Using an XTIO instead of a TIOT entry provides VSCR, virtual storage constraint relief.
- Solution
 - OPEN will accept subsystem DCBs with associated XTIOs if the subsystem supports it.
 - As with XTIO support for DASD and tape, the system programmer must set NON_VSAM_XTIO=YES in the DEVSUPxx member of PARMLIB and the user's DCBE macro must have LOC=ANY.
- Benefit / Value
 - 24-bit VSCR, meaning that several control blocks for that data set allocation will be above the 16 MB line.

Usage and invocation – XTIO support for subsystem DCBs and ACBs

- JES2 and JES3 set an indicator (SSALXTIO or SSAGXTIO) that results in the new DSABSSXT bit being on to mean that the subsystem supports these dynamic allocation options: XTIO, uncaptured UCBS and above-the-line DSABs
- Ask the provider of other subsystems whether they support this option by setting the SSALXTIO or SSAGXTIO bit in the IEFSSAL macro.

Interactions and dependencies

- Software Dependencies

- None
- Hardware Dependencies
 - None
- Exploiters
 - If you have a software subsystem (that is invoked by coding SUBSYS= on a DD statement), you can ask the provider whether it supports the XTIO, UCB nocapture and DSAB-above-the-line options of dynamic allocation.

Migration and coexistence considerations

- None unless you have a message automation product that will be affected by enabling the addition text in multiline messages from OPEN/CLOSE/EOV after setting on the new OCE_ABEND_DESCRIP=YES option in the DEVSUPxx member of PARMLIB.

Installation

- Nothing is needed.
- Optionally you can set these options in the DEVSUPxx member of PARMLIB:
 - OCE_ABEND_DESCRIP={YES | NO}
 - NON_VSAM_XTIO={YES | NO}

Session summary

- Some OPEN/CLOSE/EOV cryptic messages can contain English text.
- Diagnosis of user errors with a DCBE can be determined.
- QSAM does more efficient and reliable calculations for MULTSDN.
- Tape volumes can be freed early and reliably.
- You can get VSCR benefits with subsystem dynamic allocations.

Appendix - References

- *z/OS DFSMS Using Data Sets*, SC26-7410
- *z/OS DFSMS Macro Instructions for Data Sets*, SC26-7408
- *z/OS MVS JCL Reference*, SA22-7597
- *z/OS MVS System Messages, Vol 7 (IEB-IEE)*, SA22-7637
- *z/OS MVS Authorized Assembler Services Guide*, SA22-7608