DFSORT User Guide

IBM Integrated Accelerator for Z Sort (PH03207)

on

zOS V2R3 and V2R4

PTFs UI90067 and UI90068

GA Version 1.0

September, 2020

DFSORT Development

IBM Systems
Tucson, Arizona

DFSORT WEB SITE
For papers, online books, news, tips, examples and more, visit the DFSORT home page
at URL: http://www.ibm.com/storage/dfsort

Overview

This paper is the documentation for DFSORT z/OS V2R3 PTF UI90067 and z/OS V2R4 PTF UI90068, which will be made available in SEPTEMBER 25, 2020.

The IBM z15 provides a new on chip sort accelerator known as the IBM Integrated Accelerator for Z Sort. This paper highlights and explains how to take advantage of the z15 Z Sort capabilities using DFSORT and DFSORT's robust ICETOOL utility. Details on restrictions and messages are also provided.

Contents

User Guide for DFSORT APAR PH03207

Introduction

DFSORT is IBM's high-performance sort, merge, copy, analysis and reporting product. DFSORT is an optional feature of z/OS. Sorting data is a significant part of IBM Z workloads including batch workloads, database query processing, and utility processing. The amount of data stored and processed on Z continues to grow at a high rate driving an ever-increasing sort workload. The IBM z15 provides an Integrated Accelerator for Z Sort, a new CPU co-processor, and also provides the new SORTL instruction. Use of the SORTL instruction allows DFSORT to use a hardware accelerated approach to sorting. DFSORT and DFSORT's ICETOOL are enhanced to exploit this hardware to reduce the CPU costs and improve the elapsed time for eligible workloads.

Users of the Db2 Utilities for z/OS Suite with the application of APAR PH28183 will also be able to take advantage of this new function during REORG TABLESPACE if applied in conjunction with DFSORT APAR PH03207.

A new version of the IBM Z Batch Network Analyzer (zBNA) V2.2 provides support for the Integrated Accelerator for Z Sort. zBNA has a new application, DFSORT Z Sort, which uniquely identifies DFSORT eligible candidates and estimates the z15 benefits, using data from the current environment and does not require a z15. The new information integrates with the traditional batch information already available in zBNA.

This no-charge tool is available for download on IBM Techdocs.
www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS5132

This paper provides the documentation you need to start using the features and messages associated with z/OS DFSORT APAR PH03207 which corresponds to V2R3 PTF UI90067 or V2R4 PTF UI90068. The information in this paper will be included in the z/OS DFSORT books in the next refresh.

For information on DFSORT please refer to z/OS DFSORT Application Programming Guide (SC23-6878) that provides an overview on DFSORT and ICETOOL features, and the framework of existing DFSORT features upon which these new features are built. You should refer to z/OS DFSORT Messages, Codes and Diagnosis Guide (SC23-6879) for general information on DFSORT messages.

Please refer to the documentation provided by Db2 APAR PH28183 for details for the REORG DFSORT improvement new function as part of the Db2 Utilities for z/OS Suite.

You can access all DFSORT books online by clicking the PUBLICATIONS
link on the DFSORT home page at URL:
http://www.ibm.com/storage/dfsort

Prerequisites for exploiting IBM Integrated Accelerator for Z Sort

Hardware Prerequisites

In order to take advantage of the IBM Integrated Accelerator for Z Sort, a z15 or later
processor is required. DFSORT was enhanced to use the latest I/O technologies
available. When using the IBM Integrated Accelerator for Z Sort and if DFSORT is not
able to contain a SORT workload entirely in memory, then it may require work datasets
that support zHPF (z High Performance FICON) and must be enabled. Please see the
section on Memory Requirements to ensure your system is configured with the proper
amount of memory for your sort workloads.

Software Prerequisites

The DFSORT support will be available on zOS V2R3, V2R4 and later releases. In order
to use the new function, you must install the PTFs for this support shipped in APAR
PH03207.

You can run a DFSORT job to determine if the PTF is applied on your system by
referencing the ICE201I message.

    ICE201I C RECORD TYPE ...

The C indicates you have z/OS DFSORT V2R3 PTF UI90067 or z/OS DFSORT V2R4
PTF UI90068 and can use the new function described in this paper. If you do
not see ICE201I C, ask your System Programmer to install z/OS DFSORT V2R3 PTF
UI90067 or z/OS DFSORT V2R4 PTF UI90068 so you can use the new function.

Operational changes that require user action

The following are operational changes that may require user action for existing DFSORT/ICETOOL applications that use certain functions as specified.

DFSORT exploitation of IBM Integrated Accelerator for Z Sort is shipped as disabled by default. Enabling the new function via DFSORT installation option allows you to quickly use the new function without having to modify JCL and Programs. You can enable it for individual jobs as well. You should consider first enabling Z Sort for individual jobs. This allows you to evaluate Z Sort in your overall batch framework. These options can be activated using one of the following methods in:

- ICEPRMxx members of concatenated PARMLIB, or
- an ICEMAC macro option or
- OPTION control statement

ICEPRMxx

When you activate ICEPRMxx members with a START ICEOPT command, the changes to the defaults in ICEPRMxx are merged with and override the specified or IBM-supplied ICEMAC defaults. The specific value used for each installation default in each environment is determined from the following hierarchy:

1. The last value specified in an activated ICEPRMxx member if any
2. The value specified by the ICEMAC macro if any
3. The IBM-supplied default.

Example of change to JCL (ICEAM1) installation environment options:
       JCL
        ZSORT=YES

Example of change to INV (ICEAM2) installation environment options:
       INV
        ZSORT=YES

Or you can reference the "Figure 18. An ICEPRM01 Member in PARMLIB" in the DFSORT Installation and Customization publication.

ICEMAC

To enable using ICEMAC, specify

JCL,ZSORT=YES or
INV,ZSORT=YES

To disable using ICEMAC, specify

JCL,ZSORT=NO or
INV,ZSORT=NO

JCL OPTION control statement

The OPTION control statement may be specified in either SYSIN or DFSPARM DD statements.

To enable you can specify OPTION ZSORT in the SORT Control statement.

To disable you can specify OPTION NOZSORT in the SORT Control statement.

If DFSORT determines a job does not meet the criteria/requirements for using IBM Integrated Accelerator for Z Sort, then it will ***NOT*** be eligible to use Z Sort DFSORT. If the job is not eligible, then it will instead use the existing framework in DFSORT. One of the primary requirements for IBM Integrated Accelerator for Z Sort is providing enough virtual, real and auxiliary storage. Sort jobs which run in memory constrained environments, where the amount of memory available to be used by DFSORT jobs is restricted, may not achieve optimal performance results or may not be able to exploit the IBM Integrated Accelerator for Z Sort (see Memory Requirements).

Required storage for running DFSORT resident

The size of LPA will be increased if you run DFSORT resident. The following table shows approximate kilobytes (KB) of virtual memory required in the LPA (SORTLPA library) after this PTF is applied.

Release	Below	Above
V2R3	160K	1005K
V2R4	160K	1012K

Supported functions using DFSORT Z Sort

DFSORT will be leveraging the IBM Integrated Accelerator for Z Sort which will help improve CPU and elapsed time of DFSORT sort applications that are eligible. Invocations of DFSORT through batch or using the ICETOOL program will exploit this new capability. DFSORT's most efficient sort technique BLOCKSET is enhanced to exploit IBM Integrated Accelerator. The older and less efficient Peerage/Vale and Conventional techniques do not support the exploitation of IBM Integrated Accelerator. DFSORT by default uses Blockset for a sort application unless something prevents it from doing so. If possible and appropriate, update your jobs to always use Blockset technique. The ICE800I message provides reasons why Blockset could not be used.

Memory Requirements

DFSORT requires the use of 64-bit memory objects (above –the-bar storage) in order to exploit the Z Sort Accelerator for sort workloads. Because the Integrated Accelerator for Z Sort is part of the CPU, and memory latency is much lower than disk latency, sorting in memory is more efficient than sorting with both memory and disk workspace. By allowing DFSORT Z Sort to process the input completely in memory, it can achieve the best results in both elapsed time and CPU time. In order to process completely in memory, DFSORT may require a memory object size to be ~2x the input size (actual size may vary depending on record format, record length (LRECL), and key length). For simplicity MOSIZE=MAX is recommended, which allows DFSORT to obtain the optimal size of memory object. At a minimum, DFSORT memory object size must be at least 75% of the input file size in order to be eligible for Z Sort. When a sort workload is not able to be processed completely in memory, DFSORT will use disk workspace in combination with memory object sorting.

When Z Sort requires the use of disk workspace (sort works), it is required that System z High Performance FICON (zHPF) is enabled on the system and the disks, used for workspace, support it. This is needed in order to provide the benefit of reducing elapsed and CPU time. When using Z Sort with disk workspace, DFSORT requires enough disk workspace to be allocated and available. The amount of disk workspace required is at least the size of the input file. When DFSORT dynamically allocates the sort workspace, DFSORT ensures enough space is allocated. For jobs that define the sort disk workspace using SORTWKxx DD statements, you must specify enough primary and secondary space in the JCL space parameters.

If DFSORT is not able to obtain enough 64-bit storage for Z Sort, or enough disk space when using sort works, DFSORT will transition to traditional sort processing. DFSORT will issue message ICE267I indicating whether the ZSORT accelerator was used with a

reason code to explain the condition the sort job met. Please refer to the section for New Messages for an explanation on the ICE267I message.

For below-the-bar memory requirements, DFSORT must be able to obtain enough memory for its various control blocks that reside both below and above the 16MB line. Specifying a SIZE/MAINSIZE=MAX is recommended in order to achieve the best performance.   For sorts under 100Gb, the shipped default DSA value of 128 will be enough. For sorts larger than 100GB, consider increasing DSA to 256. Note that this is only a limit. DFSORT's dynamic storage adjustment calculates the optimal virtual storage allocation for each sort. So, increasing the DSA limit will only impact very large sorts that DFSORT determines would benefit from increased virtual storage.

Please note, memory object size and limits can be affected by the following:
- REGION and MEMLIMIT parameters on the JOB or EXEC JCL statement:
  - If MEMLIMIT is not specified and REGION=0M is specified, MEMLIMIT=NOLIMIT is assumed by the system.
  - If REGION is set equal to a value other than 0M and MEMLIMIT is not specified, the default MEMLIMIT value defined in the SMFPRMxx member of PARMLIB is used. It is recommended that your jobs that request ZSORT specify a REGION=0M.
- An IEFUSI Exit can limit the total amount of virtual storage above the bar available to any address space
- MOSIZE or EXPOLD/EXPRES/EXPMAX DFSORT Job/Installation options can limit the size of the memory objects obtained by DFSORT. It is recommended to use MOSIZE=MAX for Z Sort, and to not further govern with EXPOLD/EXPRES/EXPMAX limits.
- Physical resources: Ensure the system is configured with adequate real and auxiliary storage to back any increased virtual storage demands.

Performance Improvements

The IBM Integrated Accelerator for Z Sort has been implemented on each core and it provides new architected instructions for sorting data to speed up sorting operations. It supports multiple active sorts in parallel and it is designed to:

- Optimize elapsed time and CPU time
- Shorten the batch window (primarily targeting existing batch type sort workloads)
- Improve select database functions, such as reorganization to help reorganize data more frequently as sorted data is faster to access in interactive use

The following describes performance benchmarks that were captured during internal testing.

When migrating from z14 to z15 and exploiting the Integrated Accelerator for Z Sort for your DFSORT in-memory sort jobs you may be able to reduce elapsed time by up to 30% and CPU time by up to 40% for datasets with record lengths up to 500 bytes.

When migrating from z14 to z15 and exploiting the Integrated Accelerator for Z Sort for your DFSORT in-memory sort jobs can reduce elapsed time by up to 20% and CPU time by up to 40% for datasets with record lengths greater than 500 bytes.

DFSORT In-Memory Sort
Average Elapsed Time and CPU Benefit

z15 DFSORT
with Integrated Accelerator for z Sort
vs.
z14 DFSORT
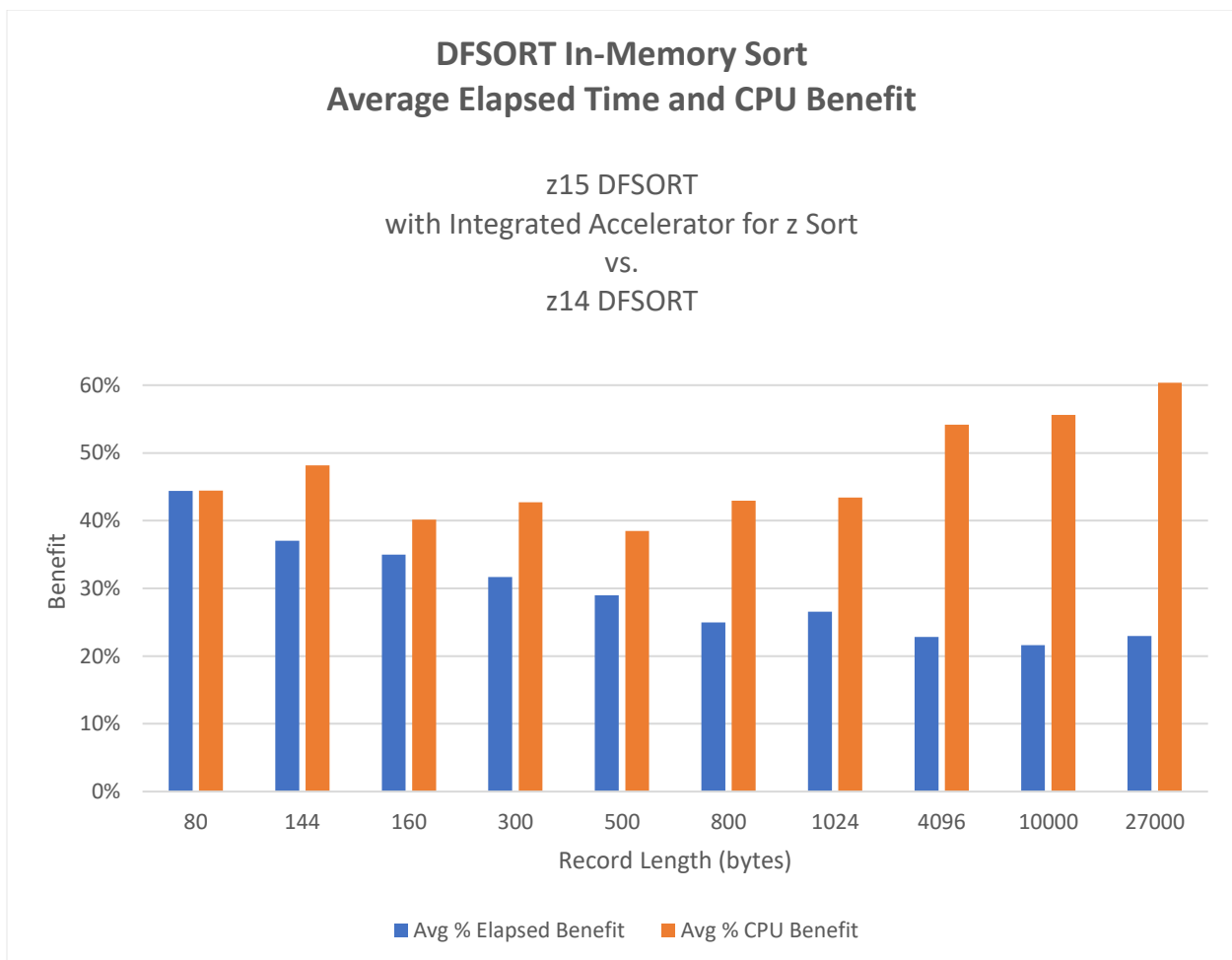
Updating your DFSORT in-memory sort jobs to exploit the Integrated Accelerator for Z Sort on z15 can reduce elapsed time by up to 25% and CPU time by up to 30% for datasets with record lengths up to 500 bytes.

Updating your DFSORT in-memory sort jobs to exploit the Integrated Accelerator for Z Sort on z15 can reduce elapsed time by up to 15% and CPU time by up to 30% for data sets with record lengths greater than 500 bytes.
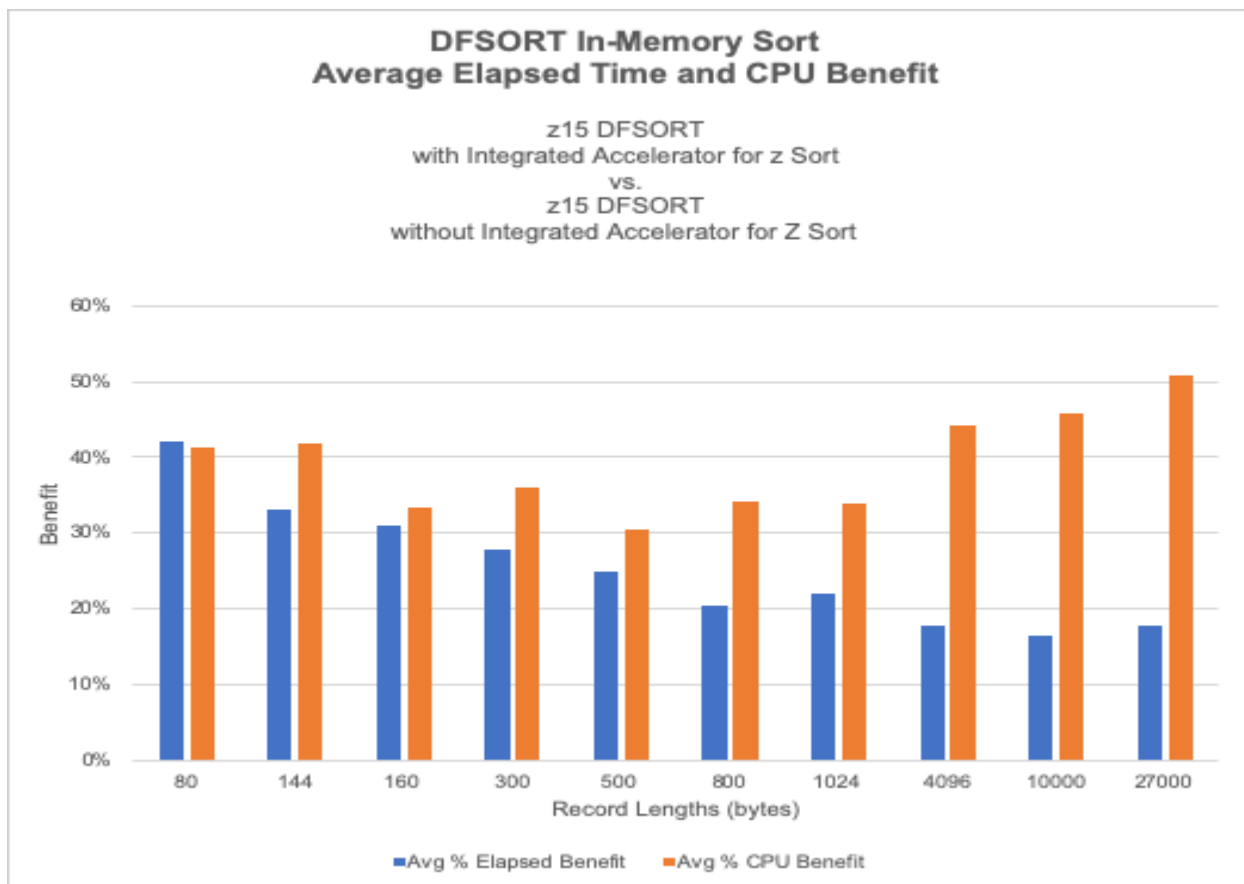
Sort Application Restrictions

DFSORT will issue message ICE267I indicating whether Z Sort was used or not used. If Z Sort was not used, a reason code will be issued to explain the non-usage condition. Please refer to the section "New messages" for an explanation of the ICE267I message.

Below is a list of restrictions that will disable DFSORT's use of the IBM Integrated Accelerator for Z Sort and will revert to DFSORT traditional sorting technique:

SORTL Facility is not enabled on the processor
ZSORT is not enabled for the DFSORT job
COPY Operation specified
INREC specified
JOINKEYS specified
MERGE FIELDS specified
MODS(EXITS) statement specified
OUTREC specified
OUTFIL specified
SUM FIELDS specified
DFSORT invoked from program
Memory Objects cannot be created
Insufficient Memory Object Storage available (required more than currently available)
Unsupported sort fields specified (examples Unicode, Locale, ALTSEQ …)
Unknown File Size or FSZ=0
SIZE/FILSZ=Uxxxxxx is specified
Input/Output is a VSAM Cluster
Sort Control field positions is beyond 4092 and VLSHRT is specified
Usage of EXCP was requested
Insufficient storage (example above the line or below the line)
Key greater than 4088 or greater than 4080 if EQUALS is specified
For variable records, the record length (LRECL) must be greater than 24
ZHPF is unavailable for a sort that could not be performed entirely in memory
Insufficient amount of sort workspace

Please note that since DFSORT is dependent on the usage of the BSAM Access Method for reading from input and writing to output when using the Integrated Accelerator for Z Sort, a BSAM supported blocksize must be specified for the SORTIN and SORTOUT DD statements.

SMF Updates

DFSORT has enhanced the SMF type-16 record by adding new fields that will allow the capture of statistical information when ZSORT is used and with SMF=FULL or SMF=SHORT is in effect.

The changes to the SMF type-16 record consist of adding a new self-describing section (new triplet) that is pointed to under the existing ICESMFH (header) DSECT. For SMF analysis programs to report on the new ZSORT statistics, a reserved field ICERES1B has been changed to ICEHDRLN that contains the length of the header/self-defining section and can be used to determine if the new triplet is present. Evaluate your applications using DFSORT's ICESMF macro to determine if modifications are required.

```
ICEOFDSN DS    H       NUMBER OF OUTFIL DATA SET SECTIONS (MAX OF 16)
*
ICEHDRLN DS    H       Length of Header/self-defining section  @PH03207C
ICESPGN  DS    H        PERFORMANCE GROUP NUMBER
ICEUSER  DS    CL8     USER ID OF JOB OF SESSION IN EXECUTION
ICEGROUP DS    CL8     GROUP ID OF JOB OF SESSION IN EXECUTION
ICERES1D DS    XL8     RESERVED - DO NOT USE
*                                                              @PH03207A
ICEZSRT  DS    F       Offset to ZSORT statistics section      @PH03207A
ICEZSRTL DS    H       ZSORT statistics section length         @PH03207A
ICEZSRTN DS    H       Number of ZSORT statistics sections     @PH03207A
*                        Zero if no ZSORT statistics section   @PH03207A
*
ICEHEND  EQU   *,*-ICESMFH   END OF SMF RECORD SECTION
*
```

The ICESMFD DSECT was changed to add fields at the end of the data portion of the DSECT. These fields will be present if SMF=SHORT or SMF=FULL is in effect.

```
ICETCBT  DS    F       Total TCB time in hundredths of a sec  @PH03207A
ICEFLBY5 DS    XL1     Flag byte 5                            @PH03207A
ICEZSRTF EQU   X'80'   Bit 0 - ZSORT used                     @PH03207A
*                      Bits 1-7 RESERVED                      @PH03207A
*                                                             @PH03207A
ICEZSRNU DS    XL1     ZSORT non usage code                   @PH03207A
*                       X'00' if ZSORT used                   @PH03207A
ICERES6D DS    XL2     UNUSED                                 @PH03207A
*                                                             @PH03207A
ICEDEND  EQU   *,*-ICESMFD   END OF SMF RECORD DATA SECTION
```

The ICESMFZS DSECT is the new DSECT added. This section will only be present if zSORT is used and SMF=FULL is in effect.   There may be some cases where this section is present but ZSORT was disabled by DFSORT. The ICEZSINF flag bit can be used to verify if the statistics are valid.

```
ICESMFZS DSECT                                                 @PH03207A
***************************************************************@PH03207A
*   ZSORT STATISTICS SECTION                                   @PH03207A
***************************************************************@PH03207A
ICEZSFLG DS    X       ZSORT Flags                             @PH03207A
ICEZSINF EQU   X'80'   BIT 0 - ZSORT STATISTICS VALID          @PH03207A
*                      BITS 1-7 UNUSED                         @PH03207A
ICEZSRSV DS    XL3     RESERVED                                @PH03207A
ICEZSPH1 DS    F       ZSORT PH1 ELAPSED IN HUNDRETHS OF SEC   @PH03207A
ICEZSPH3 DS    F       ZSORT PH3 ELAPSED IN HUNDRETHS OF SEC   @PH03207A
ICEZSTC1 DS    F       ZSORT PH1 TCB TIME IN HUNDRETHS OF SEC  @PH03207A
ICEZSTC3 DS    F       ZSORT PH3 TCB TIME IN HUNDRETHS OF SEC  @PH03207A
ICEZSRS1 DS    XL24    RESERVED                                @PH03207A
ICEZSDIA DS    XL400   ZSORT DIAGNOSTIC INFORMATION AREA       @PH03207A
ICEZSDIV DS    XL1     ZSORT DIAGNOSTIC VERSION                @PH03207A
ICEZSRS2 DS    XL3     RESERVED                                @PH03207A
*                                                              @PH03207A
ICEZSEND EQU   *,*-ICESMFZS END OF ZSORT STATISTICS SECTION    @PH03207A
         SPACE 2                                               @PH03207A
```

New messages

This section shows messages that have been added for PTFs UI90067 and UI90068. Refer to z/OS DFSORT Messages, Codes and Diagnosis Guide (SC23-6879) for general information on DFSORT messages.

ICE267I

ICE267A ZSORT ACCELERATOR PATH {USED | NOT USED}   RSN=*nnn*

EXPLANATION: For this run, USED | NOT USED indicates if the new zSORT accelerator path is selected or not. If zSORT accelerator path is selected, then RSN will be 0 unless an error was encountered. If ZSORT accelerator path is NOT selected, then *nnn* indicates the reason codes and DFSORT will revert to using a traditional sorting technique unless another terminating error was encountered.

The following are reason codes that explain why the IBM Integrated Accelerator for Z Sort was not used.

193 - The Job is NOT running on a z15 and higher processor

      The job in question is running on a machine which does not have the IBM Integrated Accelerator for Z Sort available

194 - ZSORT Path is not activated

      DFSORT exploitation of IBM Integrated Accelerator for Z Sort is shipped as disabled by default. It can be enabled by either with ICEPRMxx members of concatenated PARMLIB, with the ICEMAC macro, or with the ZSORT option command statement.

195 - COPY Operation specified

      The job is requesting a COPY operation and does not support the IBM Integrated Accelerator for Z Sort

196 - MERGE FIELDS specified

      The job is requesting a MERGE operation and does not support the IBM Integrated Accelerator for Z Sort

197 - INREC is specified and is not supported with the use of the IBM Integrated Accelerator for Z Sort

198 - OUTREC is specified and is not supported with the use of the IBM Integrated Accelerator for Z Sort

199 - OUTFIL is specified and is not supported with the use of the IBM Integrated Accelerator for Z Sort

200 - DFSORT was invoked via a program and is not supported with the use of the IBM Integrated Accelerator for Z Sort

201 - Obtaining the required amount of Memory Object storage failed and is required for using the IBM Integrated Accelerator for Z Sort.

Memory object size and limits can be affected by the following:
- REGION and MEMLIMIT parameters on the JOB or EXEC JCL statement:
  - If MEMLIMIT is not specified and REGION=0M is specified, MEMLIMIT=NOLIMIT is assumed by the system.
  - If REGION is set equal to a value other than 0M and MEMLIMIT is not specified, the default MEMLIMIT value defined in the SMFPRMxx member of PARMLIB is used. It is recommended that your jobs that request ZSORT specify a REGION=0M.
- An IEFUSI Exit can limit the total amount of virtual storage above the bar available to any address space
- MOSIZE or EXPOLD/EXPRES/EXPMAX DFSORT Job/Installation options can limit the size of the memory objects obtained by DFSORT. It is recommended to use MOSIZE=MAX for Z Sort, and to not further govern with EXPOLD/EXPRES/EXPMAX limits.
- Physical resources: Ensure the system is configured with adequate real and auxiliary storage to back any increased virtual storage demands.

209 - Minimum requirements are NOT met to exploit the Z Sort Accelerator

The amount of memory sorting the specific input exceeded the amount of memory DFSORT was able to obtain

210 - DFSORT was unable to obtain sufficient Memory Object storage or MOSIZE=0 was specified. In order to use the IBM Integrated Accelerator for Z Sort DFSORT must be able to obtain a enough memory object storage

211 - Unsupported sort fields (UTFxx Formats, LOCALE, ALTSEQ.)

IBM Integrated Accelerator for Z Sort is aimed at improving performance. Using formats that require conversion before sorting may degrade the performance. Using such formats will disable the use of the IBM Integrated Accelerator for Z Sort

212 - SUM FIELDS is specified and is not supported with the use of the IBM Integrated Accelerator for Z Sort

213 - JOINKEYS Maintask, Subtask1 or Subtask2 execution is specified and is not supported with the use of the IBM Integrated Accelerator for Z Sort

214 - MODS Statement is specified (User Exits are in use) and is not supported with the use of the IBM Integrated Accelerator for Z Sort

215 - Unknown File Size or FSZ=0 and is not supported with the use of the IBM Integrated Accelerator for Z Sort

   If DFSORT does not have an accurate file size to be sorted, then IBM Integrated Accelerator for Z is not used

216 - When sorting variable data, the input LRECL must be greater than 24 bytes.

217 - The input data set is a VSAM data set is not supported with the use of the IBM Integrated Accelerator for Z Sort

226 - Usage of DEBUG Parms such as $EXCPI$,$EXCPO$,$EXCPF$ and is not supported with the use of the IBM Integrated Accelerator for Z Sort

   The exploitation of IBM Integrated Accelerator for Z Sort requires the use of BSAM. Forcing DFSORT to use EXCP will disable the use of IBM Integrated Accelerator for Z Sort.

227 - DFSORT was unable to obtain sufficient amount of below-the-bar storage

   Specifying SIZE/MAINSIZE=MAX may allow the job to obtain enough below-the-bar storage that will allow the use of the IBM Integrated Accelerator for Z Sort.

228 - The SORT Control Field key position is beyond 4092 bytes for variable records and VLSHRT was specified.

229 - The total length of the key is greater than 4080 or 4088

   DFSORT's exploitation of IBM Integrated Accelerator for Z Sort has the following limits on the total length of the control fields

   With EQUALS in effect the total length of control fields is 4080

   With NOEQUALS in effect the total length of control fields is 4088

Please Note: DFSORT no longer restricts sorting up to 4,294,967,295 records with EQUALS when using IBM Integrated Accelerator for Z Sort

230 - A condition exists in key processing that prevents DFSORT using the IBM Integrated Accelerator for Z Sort

231 - File Size is 32MB or smaller and is not supported with the use of the IBM Integrated Accelerator for Z Sort

232 - zHPF is either disabled on the system or the disk sortwork data sets reside on non zHPF capable DASD volumes.

Please ensure zHPF is available on the system and ensure sortwork data sets are created on devices that support zHPF. The necessary amount of memory object was not available that would have allowed the sort job to be processed entirely in memory.

233 - SIZE/FILSZ=Uxxxxxx was requested and prevents DFSORT from using the IBM Integrated Accelerator for Z Sort

240 - DFSORT was unable to obtain 75% of the input as memory object storage.

Memory object size and limits can be affected by the following:
- o REGION and MEMLIMIT parameters on the JOB or EXEC JCL statement:
  - ▪ If MEMLIMIT is not specified and REGION=0M is specified, MEMLIMIT=NOLIMIT is assumed by the system.
  - ▪ If REGION is set equal to a value other than 0M and MEMLIMIT is not specified, the default MEMLIMIT value defined in the SMFPRMxx member of PARMLIB is used. It is recommended that your jobs that request ZSORT specify a REGION=0M.
- o An IEFUSI Exit can limit the total amount of virtual storage above the bar available to any address space
- o MOSIZE or EXPOLD/EXPRES/EXPMAX DFSORT Job/Installation options can limit the size of the memory objects obtained by DFSORT. It is recommended to use MOSIZE=MAX for Z Sort, and to not further govern with EXPOLD/EXPRES/EXPMAX limits.
- o Physical resources: Ensure the system is configured with adequate real and auxiliary storage to back any increased virtual storage demands.

241 - DFSORT was unable to obtain enough amount of disk sortwork space

For JCL defined sortworks using SORTWKxx DD statements, the amount of space specified in the SPACE parameters was not enough. Increasing the space that amounts to the input size or using dynamically allocated sort works

data sets are options to overcome this reason.

SYSTEM ACTION: None.

PROGRAMMER RESPONSE: Please refer to the reason code description. If your job returns a code that is not described, you can bypass this by disabling ZSORT for the job as a temporary circumvention and report it to IBM.

ICE268I

ICE268I B ZSORT STEPNAME *stepname* PROCESSING PHASE *n* BEGINS AT *local_timestamp*

EXPLANATION: Informational to determine processing time in the various phase of DFSORT processing.

SYSTEM ACTION: None.

PROGRAMMER RESPONSE: None.

ICE269A

ICE269A ZSORT SORT PROCESSING FAILURE: *diagnostic information*

EXPLANATION: Critical. An error was encountered during ZSORT Accelerator processing.

SYSTEM ACTION: The program terminates.

PROGRAMMER RESPONSE: Contact IBM. You can bypass this error by disabling ZSORT for the job as a temporary circumvention.

ICE399I

ICE399I MEMORY OBJECT USED IN ZSORT PATH = nM BYTES

EXPLANATION: n is the number of megabytes of memory object storage DFSORT used during this sort. If n is zero, DFSORT did not use ZSORT for this sort operation.

SYSTEM ACTION: None.

PROGRAMMER RESPONSE: None.